LEARNING PATHS

Application Security Role-Based Curriculum

SECURITY INNOVATION

.NET Developer (40 Courses, 25 Hours, 25 CPE Credits)

The .NET Developer Learning Path is designed to provide a solid foundation of .NET security features for building secure web applications, sophisticated desktop applications, or modern mobile applications.

.Net Developer learning path includes concepts such as Code Access Security (CAS), .NET cryptographic technologies and secure coding best practices that will enable learners to build secure applications in .NET. More advanced courses offer platform and language specific secure coding best practices, including ASP.ET, AJAX, C# and Windows.

- AWA 101 Fundamentals of Application Security
- AWA 102 Secure Software Concepts
- COD 101 Fundamentals of Secure Development
- COD 153 Fundamentals of Secure Ajax Code
- COD 216 Leveraging .Net Framework Code Access Security (Cas)
- COD 217 Mitigating .Net Security Threats
- COD 251 Creating Secure Ajax Code Asp - Net Foundations
- COD 255 Creating Secure Code Web API Foundations
- COD 261 Threats to Scripts
- COD 264 Protecting Sensitive Data While Scripting
- COD 311 Creating Secure Asp.Net MVC Applications
- COD 321 Protecting C# From Integer Overflows and Canonicalization Issues
- COD 322 Protecting C# From SQL And XML Injection
- COD 323 Protecting Data in C#
- DES 101 Fundamentals of Secure Architecture Fundamentals of Secure Architecture
- DES 202 Cryptographic Suite Services: Encoding, Encrypting and Hashing
- DES 203 Cryptographic Components Randomness, Algorithms, And Key Management
- DES 204 The Role of Cryptography in Application Development
- DES 205 Message Integrity Cryptographic Functions
- DES 212 Architecture Risk Analysis and Remediation
- DES 222 Applying OWASP 2017: Mitigating Injection
- DES 223 Applying OWASP 2017: Mitigating Broken Authentication
- DES 224 Applying OWASP 2017: Mitigating Sensitive Data Exposure
- DES 225 Applying OWASP 2017: Mitigating XML External Entities
- DES 226 Applying OWASP 2017: Mitigating Broken Access Control
- DES 227 Applying OWASP 2017: Mitigating Security Misconfiguration
- DES 228 Applying OWASP 2017: Mitigating Cross Site Scripting
- DES 229 Applying OWASP 2017: Mitigating Insecure Deserialization
- DES 230 Applying OWASP 2017: Mitigating Use of Components with Known Vulnerabilities

- DES 231 Applying OWASP 2017: Mitigating Insufficient Logging and Monitoring
- DES 311 Creating Secure Application Architecture
- ENG 191 Introduction to The Microsoft SDL
- ENG 192 Implementing the Agile MS SDL
- ENG 193 Implementing the MS SDL Optimization Model
- ENG 194 Implementing MS SDL Line of Business
- ENG 195 Implementing the MS SDL Threat Modeling Tool
- ENG 205 Fundamentals of Threat Modeling
- ENG 211 How to Create Application Security Design Requirements
- ENG 311 Attack Surface Analysis and Reduction
- ENG 312 How to Perform A Security Code Review

Android Developer (38 Courses, 19 Hours, 19 CPE Credits)

The Android Developer Learning Path is designed to provide a solid foundation of security features necessary to develop applications for devices powered by the Android operating system.

Android Developer learning path provides secure coding best practices for designing and building android applications. Developers will understand how to use mobile application development best practices, identify common android application risks, create a mobile application threat model, and apply Android platform-specific knowledge.

- AWA 101 Fundamentals of Application Security
- AWA 102 Secure Software Concepts
- COD 110 Fundamentals of Secure Mobile Development
- COD 229 Insecure IoT Mobile Interface
- COD 234 Mobile Threats and Mitigations
- COD 235 Defending Mobile Data with Cryptography
- COD 236 Mobile App Authentication and Authorization
- COD 237 Defending Mobile App Code
- COD 318 Creating Secure Android Code in Java
- DES 202 Cryptographic Suite Services: Encoding, Encrypting and Hashing
- DES 203 Cryptographic Components Randomness, Algorithms, And Key Management Cryptographic Components Randomness, Algorithms, And Key Management
- DES 204 The Role of Cryptography in Application Development
- DES 205 Message Integrity Cryptographic Functions
- DES 212 Architecture Risk Analysis and Remediation
- DES 260 Fundamentals of IoT Architecture and Design
- ENG 112 Essential Access Control for Mobile Devices
- ENG 191 Introduction to The Microsoft SDL

- ENG 193 Implementing the MS SDL Optimization Model
- ENG 194 Implementing MS SDL Line of Business
- ENG 195 Implementing the MS SDL Threat Modeling Tool
- ENG 205 Fundamentals of Threat Modeling
- ENG 211 How to Create Application Security Design Requirements
- ENG 311 Attack Surface Analysis and Reduction
- ENG 312 How to Perform A Security Code Review
- TST 252 Testing for OS Command Injection
- TST 257 Testing for Use of Hard-Coded Credentials
- TST 259 Testing for Unrestricted Upload of File with Dangerous Type
- TST 260 Testing for Reliance on Untrusted Inputs in A Security Decision
- TST 261 Testing for Execution with Unnecessary Privileges
- TST 264 Testing for Download of Code Without Integrity Check
- TST 266 Testing for Inclusion of Functionality from Untrusted Control Sphere
- TST 267 Testing for Incorrect Permission Assignment for Critical Resource
- TST 268 Testing for Use of a Potentially Dangerous Function
- TST 270 Testing for Incorrect Calculation of Buffer Size
- TST 271 Testing for Improper Restriction of Excessive Authentication Attempts
- TST 272 Testing for Open Redirect

Back-End Web Developer (23 Courses, 18 Hours, 18 CPE Credits)

The Back-End Web Developer Learning Path is designed to provide a solid foundation of security features needed to write web services and API's used by front-end developers and mobile application developers.

Back-End Web Developer provides secure coding best practices in all phases of the development life cycle across cutting-edge technologies like Node.js, Angular.js, and MySQL with special attention to managing the interchange of data between the server and users.

- AWA 101 Fundamentals of Application Security
- AWA 102 Secure Software Concepts
- COD 101 Fundamentals of Secure Development
- COD 153 Fundamentals of Secure Ajax Code
- COD 241 Creating Secure Oracle Database Applications
- COD 251 Creating Secure Ajax Code - Asp Net
- COD 252 Creating Secure Ajax Code - Java Foundations
- COD 253 Creating Secure Aws Cloud Applications
- COD 254 Creating Secure Azure Applications
- COD 255 Creating Secure Code Web API Foundations
- DES 101 Fundamentals of Secure Architecture
- DES 202 Cryptographic Suite Services: Encoding, Encrypting and Hashing

- DES 203 Cryptographic Components Randomness, Algorithms, And Key Management Cryptographic Components Randomness, Algorithms, And Key Management
- DES 204 The Role of Cryptography in Application Development
- DES 205 Message Integrity Cryptographic Functions
- DES 212 Architecture Risk Analysis and Remediation
- DES 224 Applying OWASP 2017: Mitigating Sensitive Data
- DES 227 Applying OWASP 2017: Mitigating Security
- ENG 211 How to Create Application Security Design Requirements
- ENG 311 Attack Surface Analysis and Reduction
- ENG 312 How to Perform A Security Code Review
- TST 224 Testing for OWASP 2017: Sensitive Data Exposure
- TST 227 Testing for OWASP 2017: Security Misconfiguration

C Developer (36 Courses, 19 Hours, 19 CPE Credits)

The C Developer Learning Path is designed to provide a solid understanding of security features required to develop secure code that integrates into operating systems, operating system modules, embedded systems, or low-level libraries for other high-level languages.

C Developer provides a thorough grounding in application security concepts, including specific C coding and implementation practices like memory management and string handling. Users will understand common pitfalls and security flaws and be encouraged to utilize security best practices to help protect C code and eliminate vulnerabilities.

- AWA 101 Fundamentals of Application Security
- AWA 102 Secure Software Concepts
- COD 101 Fundamentals of Secure Development
- COD 201 Secure C Encrypted Network Communications
- COD 202 Secure C Run-Time Protection
- COD 301 Secure C Buffer Overflow Mitigation
- COD 302 Secure C Memory Management
- COD 303 Common C Vulnerabilities
- DES 101 Fundamentals of Secure Architecture
- DES 202 Cryptographic Suite Services: Encoding, Encrypting and Hashing
- DES 203 Cryptographic Components Randomness, Algorithms, And Key Management Cryptographic Components Randomness, Algorithms, And Key Management
- DES 204 The Role of Cryptography in Application Development
- DES 205 Message Integrity Cryptographic Functions
- DES 212 Architecture Risk Analysis and Remediation
- DES 311 Creating Secure Application Architecture
- ENG 191 Introduction to The Microsoft SDL

- ENG 192 Implementing the Agile MS SDL
- ENG 193 Implementing the MS SDL Optimization Model
- ENG 194 Implementing MS SDL Line of Business
- ENG 195 Implementing the MS SDL Threat Modeling Tool
- ENG 205 Fundamentals of Threat Modeling
- ENG 211 How to Create Application Security Design Requirements
- ENG 311 Attack Surface Analysis and Reduction
- ENG 312 How to Perform A Security Code Review
- TST 255 Testing for Missing Authentication for Critical Function
- TST 257 Testing for Use of Hard-Coded Credentials
- TST 259 Testing for Unrestricted Upload of File with Dangerous Type
- TST 260 Testing for Reliance on Untrusted Inputs in A Security Decision
- TST 261 Testing for Execution with Unnecessary Privileges
- TST 264 Testing for Download of Code Without Integrity Check
- TST 266 Testing for Inclusion of Functionality from Untrusted Control Sphere
- TST 267 Testing for Incorrect Permission Assignment for Critical Resource
- TST 268 Testing for Use of a Potentially Dangerous Function
- TST 271 Testing for Improper Restriction of Excessive Authentication Attempts
- TST 272 Testing for Open Redirect
- TST 273 Testing for Uncontrolled Format String

C# Developer (25 Courses, 18 Hours, 18 CPE Credits)

The C# Developer Learning Path provides a thorough grounding of security features necessary to develop modern applications that run on desktops or back-end processes powering modern web applications.

C# Developer learning path provides baseline knowledge needed to design, build, and maintain efficient and reliable C# code. It also provides secure coding best practices that enable learners to build secure enterprise systems, desktop applications, websites and mobile applications. Users will also understand how to develop scalable applications using multithreading features of .NET framework.

- AWA 101 Fundamentals of Application Security
- AWA 102 Secure Software Concepts
- COD 101 Fundamentals of Secure Development
- ENG 105 How to Integrate the Microsoft MS SDL Into Your SDLC
- COD 216 Leveraging .Net Framework Code Access Security (Cas)
- COD 217 Mitigating .NET Security Threats
- COD 225 Insecure IoT Web Interfaces
- COD 255 Creating Secure Code Web API Foundations
- COD 311 Creating Secure Asp.Net MVC Applications
- COD 321 Protecting C# From Integer Overflows and Canonicalization Issues

- COD 322 Protecting C# From SQL And XML Injection
- COD 323 Protecting Data in C#
- DES 101 Fundamentals of Secure Architecture
- DES 202 Cryptographic Suite Services: Encoding, Encrypting and Hashing
- DES 203 Cryptographic Components Randomness, Algorithms, And Key Management Cryptographic Components Randomness, Algorithms, And Key Management
- DES 204 The Role of Cryptography in Application Development
- DES 205 Message Integrity Cryptographic Functions
- DES 212 Architecture Risk Analysis and Remediation
- DES 311 Creating Secure Application Architecture
- ENG 191 Introduction to The Microsoft SDL
- ENG 192 Implementing the Agile MS SDL
- ENG 193 Implementing the MS SDL Optimization Model
- ENG 194 Implementing MS SDL Line of Business
- ENG 195 Implementing the MS SDL Threat Modeling Tool
- ENG 211 How to Create Application Security Design Requirements
- ENG 311 Attack Surface Analysis and Reduction
- ENG 312 How to Perform A Security Code Review

C++ Developer (36 Courses, 19 Hours, 19 CPE Credits)

The C++ Developer Learning Path is designed to provide continuous working knowledge of application security best practices for building applications that range from desktop applications to native mobile applications and embedded systems.

C++ Developer learning path provides baseline knowledge needed to design and build efficient, reusable, and reliable C++ code that interacts with low-level systems and hardware resources. Users will develop the knowledge and skills required to mitigate memory corruption vulnerabilities, protect data in transit using strong TLS ciphers, and protect data using cryptographic best practices while applying secure coding best practices.

- AWA 101 Fundamentals of Application Security
- AWA 102 Secure Software Concepts
- COD 101 Fundamentals of Secure Development
- COD 206 Creating Secure C++ Code
- COD 207 Communication Security in C++
- COD 262 Fundamentals of Secure Scripting
- COD 263 Secure Scripting with Perl, Python, Bash and Ruby
- COD 307 Protecting Data in C++
- DES 101 Fundamentals of Secure Architecture
- DES 202 Cryptographic Suite Services: Encoding, Encrypting and Hashing

- DES 203 Cryptographic Components: Randomness, Algorithms, And Key Management
- DES 204 The Role of Cryptography in Application Development
- DES 205 Message Integrity Cryptographic Functions
- DES 212 Architecture Risk Analysis and Remediation
- DES 311 Creating Secure Application Architecture
- ENG 191 Introduction to The Microsoft SDL
- ENG 192 Implementing the Agile MS SDL
- ENG 193 Implementing the MS SDL Optimization Model
- ENG 194 Implementing MS SDL Line of Business
- ENG 195 Implementing the MS SDL Threat Modeling Tool
- ENG 205 Fundamentals of Threat Modeling
- ENG 211 How to Create Application Security Design Requirements
- ENG 311 Attack Surface Analysis and Reduction
- ENG 312 How to Perform A Security Code Review
- TST 255 Testing for Missing Authentication for Critical Function
- TST 257 Testing for Use of Hard-Coded Credentials
- TST 259 Testing for Unrestricted Upload of File with Dangerous Type
- TST 261 Testing for Execution with Unnecessary Privileges
- TST 264 Testing for Download of Code Without Integrity Check
- TST 266 Testing for Inclusion of Functionality from Untrusted Control Sphere
- TST 267 Testing for Incorrect Permission Assignment for Critical Resource
- TST 268 Testing for Use of a Potentially Dangerous Function
- TST 271 Testing for Improper Restriction of Excessive Authentication Attempts
- TST 272 Testing for Open Redirect
- TST 273 Testing for Uncontrolled Format String

Front-End Developer (38 Courses, 23 Hours, 23 CPE Credits)

The Front-end Developer Learning Path provides a solid foundation for using markup languages, design and client-side scripts and framework to create secure environments for everything that users touch.

Front-end Developer learning path is designed to provide an understanding of how vulnerabilities are discovered and exploited and offers a thorough understanding of how to build a strong line of defense and provides a deep understanding of HTML, CSS and responsive web development.

- AWA 101 Fundamentals of Application Security
- AWA 102 Secure Software Concepts
- COD 101 Fundamentals of Secure Development
- COD 153 Fundamentals of Secure Ajax Code
- COD 255 Creating Secure Code Web API Foundations

- COD 256 Creating Secure Code Ruby on Rails Foundations
- COD 257 Creating Secure Python Web Applications
- COD 259 Node.Js Threats and Vulnerabilities
- COD 315 Creating Secure PHP Code
- COD 352 Creating Secure jQuery Code
- COD 361 Html5 Secure Threats
- COD 362 Html5 Built-In Security Features
- COD 363 Securing Html5 Data
- COD 364 Securing Html5 Connectivity
- DES 101 Fundamentals of Secure Architecture
- DES 202 Cryptographic Suite Services: Encoding, Encrypting and Hashing
- DES 203 Cryptographic Components Randomness, Algorithms, And Key Management
- DES 204 The Role of Cryptography in Application Development
- DES 205 Message Integrity Cryptographic Functions
- DES 212 Architecture Risk Analysis and Remediation
- DES 222 Applying OWASP 2017: Mitigating Injection
- DES 223 Applying OWASP 2017: Mitigating Broken Authentication
- DES 224 Applying OWASP 2017: Mitigating Sensitive Data Exposure
- DES 225 Applying OWASP 2017: Mitigating XML External Entities
- DES 226 Applying OWASP 2017: Mitigating Broken Access Control
- DES 227 Applying OWASP 2017: Mitigating Security Misconfiguration
- DES 228 Applying OWASP 2017: Mitigating Cross Site Scripting
- DES 229 Applying OWASP 2017: Mitigating Insecure Deserialization
- DES 230 Applying OWASP 2017: Mitigating Use of Components with Known Vulnerabilities
- DES 231 Applying OWASP 2017: Mitigating Insufficient Logging and Monitoring
- ENG 191 Introduction to The Microsoft SDL
- ENG 192 Implementing the Agile MS SDL
- ENG 193 Implementing the MS SDL Optimization Model
- ENG 194 Implementing MS SDL Line of Business
- ENG 195 Implementing the MS SDL Threat Modeling Tool
- ENG 205 Fundamentals of Threat Modeling
- ENG 211 How to Create Application Security Design Requirements
- ENG 311 Attack Surface Analysis and Reduction
- ENG 312 How to Perform A Security Code Review

HTML5 Developer (35 Courses, 23 Hours, 23 CPE Credits)

The HTML5 Developer Learning Path is designed to provide front-end developers responsible for holding the style and interactivity backbone together with a deeper understanding of HTML5.

HTML5 Developer learning path provides a solid foundation of HTML5 security features to help build applications with a strong line of defense. It also provides a deep understanding of how to infuse software security into the development lifecycle. Learners will develop a working knowledge of ASP.net, SWL, high-level scripting languages, version control and CMS systems.

- AWA 101 Fundamentals of Application Security
- AWA 102 Secure Software Concepts
- COD 101 Fundamentals of Secure Development
- COD 153 Fundamentals of Secure Ajax Code
- COD 255 Creating Secure Code Web API Foundations
- COD 256 Creating Secure Code Ruby on Rails Foundations
- COD 259 Node.Js Threats and Vulnerabilities
- COD 281 Java Security Model
- COD 282 Java Authentication and Authorization Services (Jaas)
- COD 283 Java Cryptography
- COD 311 Creating Secure Asp.Net MVC Applications
- COD 352 Creating Secure jQuery Code
- COD 361 Html5 Secure Threats
- COD 362 Html5 Built-In Security Features
- COD 363 Securing Html5 Data
- COD 364 Securing Html5 Connectivity
- DES 101 Fundamentals of Secure Architecture
- DES 202 Cryptographic Suite Services: Encoding, Encrypting and Hashing
- DES 203 Cryptographic Components Randomness, Algorithms, And Key Management
- DES 204 The Role of Cryptography in Application Development
- DES 205 Message Integrity Cryptographic Functions
- DES 212 Architecture Risk Analysis and Remediation
- DES 224 Applying OWASP 2017: Mitigating Sensitive Data Exposure
- DES 228 Applying OWASP 2017: Mitigating Cross Site Scripting
- ENG 191 Introduction to The Microsoft SDL
- ENG 192 Implementing the Agile MS SDL
- ENG 193 Implementing the MS SDL Optimization Model
- ENG 194 Implementing MS SDL Line of Business
- ENG 195 Implementing the MS SDL Threat Modeling Tool
- ENG 205 Fundamentals of Threat Modeling
- ENG 211 How to Create Application Security Design Requirements
- ENG 311 Attack Surface Analysis and Reduction
- ENG 312 How to Perform A Security Code Review
- TST 224 Testing for OWASP 2017: Sensitive Data Exposure
- TST 228 Testing for OWASP 2017: Cross-Site Scripting Testing for OWASP 2017: Cross-Site Scripting

iOS Developer (38 Courses, 19 Hours, 19 CPE Credits)

The iOS Developer Learning Path is designed to provide developers with a solid foundation of security features necessary to develop applications for devices powered by the iOS platform.

iOS Developer learning path provides secure coding best practices for designing and building iOS applications. Users will understand how to use mobile application development best practices, identify common iOS application risks, create a mobile application, threat model and apply iOS platform-specific knowledge.

- AWA 101 Fundamentals of Application Security
- AWA 102 Secure Software Concepts
- COD 110 Fundamentals of Secure Mobile Development
- COD 229 Insecure IoT Mobile Interface
- COD 234 Mobile Threats and Mitigations
- COD 235 Defending Mobile Data with Cryptography
- COD 236 Mobile App Authentication and Authorization
- COD 237 Defending Mobile App Code
- COD 316 Creating Secure iPhone Code in Objective-C
- COD 317 Creating Secure iOS Code in Swift
- DES 101 Fundamentals of Secure Architecture
- DES 202 Cryptographic Suite Services: Encoding, Encrypting and Hashing
- DES 203 Cryptographic Components Randomness, Algorithms, And Key Management
- DES 204 The Role of Cryptography in Application Development
- DES 205 Message Integrity Cryptographic Functions
- DES 212 Architecture Risk Analysis and Remediation
- ENG 112 Essential Access Control for Mobile Devices
- ENG 191 Introduction to The Microsoft SDL
- ENG 192 Implementing the Agile MS SDL
- ENG 193 Implementing the MS SDL Optimization Model
- ENG 194 Implementing MS SDL Line of Business
- ENG 195 Implementing the MS SDL Threat Modeling Tool
- ENG 205 Fundamentals of Threat Modeling
- ENG 211 How to Create Application Security Design Requirements
- ENG 311 Attack Surface Analysis and Reduction
- ENG 312 How to Perform A Security Code Review
- TST 252 Testing for OS Command Injection
- TST 257 Testing for Use of Hard-Coded Credentials
- TST 259 Testing for Unrestricted Upload of File with Dangerous Type
- TST 260 Testing for Reliance on Untrusted Inputs in A Security Decision
- TST 261 Testing for Execution with Unnecessary Privileges
- TST 264 Testing for Download of Code Without Integrity Check
- TST 266 Testing for Inclusion of Functionality from Untrusted Control Sphere

- TST 267 Testing for Incorrect Permission Assignment for Critical Resource
- TST 268 Testing for Use of a Potentially Dangerous Function
- TST 270 Testing for Incorrect Calculation of Buffer Size
- TST 271 Testing for Improper Restriction of Excessive Authentication Attempts
- TST 272 Testing for Open Redirect

Java Developer (51 Courses, 27 Hours, 27 CPE Credits)

The Java Developer Learning path is designed to provide a working knowledge for developing solid and secure Java applications as well as recognizing and remediating common Java web software security vulnerabilities.

Java Developer learning path explores specific Java, JRE, and J2EE constructs including core implementation practices. Additionally, this path provides an understanding of best practices for designing, developing, and testing Java based solutions using common standards and frameworks.

- AWA 101 Fundamentals of Application Security
- AWA 102 Secure Software Concepts
- COD 101 Fundamentals of Secure Development
- COD 153 Fundamentals of Secure Ajax Code
- COD 219 Creating Secure Code Sap ABAP Foundations
- COD 225 Insecure IoT Web Interfaces
- COD 226 Insecure IoT Authentication and Authorization
- COD 227 Insecure IoT Network Services
- COD 228 Insecure IoT Communications
- COD 229 Insecure IoT Mobile Interface
- COD 230 Insecure IoT Firmware
- COD 252 Creating Secure Ajax Code - Java Foundations
- COD 256 Creating Secure Code Ruby on Rails Foundations
- COD 259 Node.Js Threats and Vulnerabilities
- COD 281 Java Security Model
- COD 282 Java Authentication & Authorization Services (Jaas)
- COD 283 Java Cryptography
- COD 352 Creating Secure jQuery Code
- COD 361 Html5 Secure Threats
- COD 362 Html5 Built-In Security Features
- COD 363 Securing Html5 Data
- COD 364 Securing Html5 Connectivity
- COD 380 Protecting Java Code: SQLi And Integer Overflows

- COD 381 Protecting Java Code: Canonicalization, Information Disclosure And TOCTOU
- COD 382 Protecting Data in Java
- DES 101 Fundamentals of Secure Architecture
- DES 202 Cryptographic Suite Services: Encoding, Encrypting and Hashing
- DES 203 Cryptographic Components Randomness, Algorithms, And Key Management
- DES 204 The Role of Cryptography in Application Development
- DES 205 Message Integrity Cryptographic Functions
- DES 212 Architecture Risk Analysis and Remediation
- DES 222 Applying OWASP 2017: Mitigating Injection
- DES 223 Applying OWASP 2017: Mitigating Broken Authentication
- DES 224 Applying OWASP 2017: Mitigating Sensitive Data Exposure
- DES 225 Applying OWASP 2017: Mitigating XML External Entities
- DES 226 Applying OWASP 2017: Mitigating Broken Access Control
- DES 227 Applying OWASP 2017: Mitigating Security Misconfiguration
- DES 228 Applying OWASP 2017: Mitigating Cross Site Scripting
- DES 229 Applying OWASP 2017: Mitigating Insecure Deserialization
- DES 230 Applying OWASP 2017: Mitigating Use of Components with Known Vulnerabilities
- DES 231 Applying OWASP 2017: Mitigating Insufficient Logging and Monitoring
- DES 311 Creating Secure Application Architecture
- ENG 191 Introduction to The Microsoft SDL
- ENG 192 Implementing the Agile MS SDL
- ENG 193 Implementing the MS SDL Optimization Model
- ENG 194 Implementing MS SDL Line of Business
- ENG 195 Implementing the MS SDL Threat Modeling Tool
- ENG 205 Fundamentals of Threat Modeling
- ENG 211 How to Create Application Security Design Requirements
- ENG 311 Attack Surface Analysis and Reduction
- ENG 312 How to Perform A Security Code Review

JavaScript Developer (39 Courses, 26 Hours, 26 CPE Credits)

The JavaScript Developer Learning Path is intended for those responsible for implementing the front-end logic that defines the behavior of the visual elements of a web application and connecting this with services that may reside on the back-end.

JavaScript Developer learning path provides a thorough grounding in application security concepts and implementation practices. Designed to encourage security best practices that can help protect JavaScript and eliminate vulnerabilities, users will have a solid understanding of common pitfalls and security flaws.

- AWA 101 Fundamentals of Application Security
- AWA 102 Secure Software Concepts
- COD 101 Fundamentals of Secure Development
- COD 153 Fundamentals of Secure Ajax Code
- COD 241 Creating Secure Oracle Database Applications
- COD 255 Creating Secure Code Web API Foundations
- COD 256 Creating Secure Code Ruby on Rails Foundations
- COD 259 Node.js Threats and Vulnerabilities
- COD 281 Java Security Model
- COD 282 Java Authentication & Authentication Services (Jaas)
- COD 283 Java Cryptography
- COD 311 Creating Secure Asp.Net MVC Applications
- COD 315 Creating Secure PHP Code
- COD 352 Creating Secure jQuery Code
- COD 361 Html5 Secure Threats
- COD 362 Html5 Built-In Security Features
- COD 363 Securing Html5 Data
- COD 364 Securing Html5 Connectivity
- DES 101 Fundamentals of Secure Architecture
- DES 202 Cryptographic Suite Services: Encoding, Encrypting and Hashing
- DES 203 Cryptographic Components Randomness, Algorithms, And Key Management
- DES 204 The Role of Cryptography in Application Development
- DES 205 Message Integrity Cryptographic Functions
- DES 212 Architecture Risk Analysis and Remediation
- DES 224 Applying OWASP 2017: Mitigating Sensitive Data Exposure
- DES 225 Applying OWASP 2017: Mitigating XML External Entities
- DES 228 Applying OWASP 2017: Mitigating Cross Site Scripting
- ENG 191 Introduction to The Microsoft SDL
- ENG 192 Implementing the Agile MS SDL
- ENG 193 Implementing the MS SDL Optimization Model
- ENG 194 Implementing MS SDL Line of Business
- ENG 195 Implementing the MS SDL Threat Modeling Tool
- ENG 205 Fundamentals of Threat Modeling
- ENG 211 How to Create Application Security Design Requirements
- ENG 311 Attack Surface Analysis and Reduction
- ENG 312 How to Perform A Security Code Review
- TST 224 Testing for OWASP 2017: Sensitive Data Exposure
- TST 225 Testing for OWASP 2017: XML External Entities
- TST 228 Testing for OWASP 2017: Cross-Site Scripting Testing for OWASP 2017: Cross-Site Scripting

Mobile Developer (43 Courses, 23 Hours, 23 CPE Credits)

The Mobile Developer Learning Path is designed to provide developers with a solid foundation of security features necessary to develop applications for mobile devices.

Mobile Developer learning path explains how to identify common mobile application risks and utilize best practices for designing and building mobile applications.

- AWA 101 Fundamentals of Application Security
- AWA 102 Secure Software Concepts
- COD 110 Fundamentals of Secure Mobile Development
- COD 229 Insecure IoT Mobile Interface
- COD 234 Mobile Threats and Mitigations
- COD 235 Defending Mobile Data with Cryptography
- COD 236 Mobile App Authentication and Authorization
- COD 237 Defending Mobile App Code
- COD 261 Threats to Scripts
- COD 316 Creating Secure iPhone Code in Objective-C
- COD 317 Creating Secure iOS Code in Swift
- COD 318 Creating Secure Android Code in Java
- DES 101 Fundamentals of Secure Architecture
- DES 202 Cryptographic Suite Services: Encoding, Encrypting and Hashing
- DES 203 Cryptographic Components Randomness, Algorithms, And Key Management
- DES 204 The Role of Cryptography in Application Development
- DES 205 Message Integrity Cryptographic Functions
- DES 212 Architecture Risk Analysis and Remediation
- DES 311 Creating Secure Application Architecture
- ENG 112 Essential Access Control for Mobile Devices
- ENG 205 Fundamentals of Threat Modeling
- ENG 211 How to Create Application Security Design Requirements
- ENG 311 Attack Surface Analysis and Reduction
- ENG 312 How to Perform A Security Code Review
- TST 252 Testing for OS Command Injection
- TST 253 Testing for Classic Buffer Overflow
- TST 255 Testing for Missing Authentication for Critical Function
- TST 257 Testing for Use of Hard-Coded Credentials
- TST 258 Testing for Missing Encryption of Sensitive Data
- TST 259 Testing for Unrestricted Upload of File with Dangerous Type
- TST 260 Testing for Reliance on Untrusted Inputs in A Security Decision
- TST 261 Testing for Execution with Unnecessary Privileges
- TST 264 Testing for Download of Code Without Integrity Check
- TST 266 Testing for Inclusion of Functionality from Untrusted Control Sphere
- TST 267 Testing for Incorrect Permission Assignment for Critical Resource

- TST 268 Testing for Use of a Potentially Dangerous Function
- TST 269 Testing for Use of a Broken or Risky Cryptographic Algorithm
- TST 270 Testing for Incorrect Calculation of Buffer Size
- TST 271 Testing for Improper Restriction of Excessive Authentication Attempts
- TST 272 Testing for Open Redirect
- TST 273 Testing for Uncontrolled Format String
- TST 275 Testing for Use of a One-Way Hash without a Salt

PHP Developer (51 Courses, 27 Hours, 27 CPE Credits)

The PHP Developer Learning Path is designed to provide developers with a solid foundation of security features necessary to develop server-side web application logic.

PHP Developer learning path provides secure coding best practices to develop back-end web services connection components and support front-end developers. Learners will be able to apply these best practices to the entire web application development life cycle from concept stage to delivery and post launch.

- AWA 101 Fundamentals of Application Security
- AWA 102 Secure Software Concepts
- COD 101 Fundamentals of Secure Development
- COD 153 Fundamentals of Secure Ajax Code
- COD 255 Creating Secure Code Web API
- COD 256 Creating Secure Code Ruby on Rails Foundations
- COD 261 Threats to Scripts
- COD 262 Fundamentals of Secure Scripting
- COD 263 Secure Scripting with Perl, Python, Bash and Ruby
- COD 264 Protecting Sensitive Data While Scripting
- COD 315 Creating Secure PHP Code
- COD 361 Html5 Secure Threats
- COD 362 Html5 Built-In Security Features
- COD 363 Securing Html5 Data
- COD 364 Securing Html5 Connectivity
- DES 101 Fundamentals of Secure Architecture
- DES 202 Cryptographic Suite Services: Encoding, Encrypting and Hashing
- DES 203 Cryptographic Components Randomness, Algorithms, And Key Management
- DES 204 The Role of Cryptography in Application Development
- DES 205 Message Integrity Cryptographic Functions
- DES 212 Architecture Risk Analysis and Remediation
- DES 222 Applying OWASP 2017: Mitigating Injection
- DES 223 Applying OWASP 2017: Mitigating Broken Authentication
- DES 224 Applying OWASP 2017: Mitigating Sensitive Data Exposure

- DES 225 Applying OWASP 2017: Mitigating XML External Entities
- DES 226 Applying OWASP 2017: Mitigating Broken Access Control
- DES 227 Applying OWASP 2017: Mitigating Security Misconfiguration
- DES 228 Applying OWASP 2017: Mitigating Cross Site Scripting
- DES 229 Applying OWASP 2017: Mitigating Insecure Deserialization
- DES 230 Applying OWASP 2017: Mitigating Use of Components with Known Vulnerabilities
- DES 231 Applying OWASP 2017: Mitigating Insufficient Logging and Monitoring
- DES 311 Creating Secure Application Architecture
- ENG 191 Introduction to The Microsoft SDL
- ENG 192 Implementing the Agile MS SDL
- ENG 193 Implementing the MS SDL Optimization Model
- ENG 194 Implementing MS SDL Line of Business
- ENG 195 Implementing the MS SDL Threat Modeling Tool
- ENG 205 Fundamentals of Threat Modeling
- ENG 211 How to Create Application Security Design Requirements
- ENG 311 Attack Surface Analysis and Reduction
- ENG 312 How to Perform A Security Code Review
- TST 222 Testing for OWASP 2017: Injection
- TST 223 Testing for OWASP 2017: Broken Authentication
- TST 224 Testing for OWASP 2017: Sensitive Data Exposure
- TST 225 Testing for OWASP 2017: XML External Entities
- TST 226 Testing for OWASP 2017: Broken Access Control
- TST 227 Testing for OWASP 2017: Security Misconfiguration
- TST 228 Testing for OWASP 2017: Cross-Site Scripting
- TST 229 Testing for OWASP 2017: Insecure Deserialization
- TST 230 Testing for OWASP 2017: Use of Components with Known Vulnerabilities
- TST 231 Testing for OWASP 2017: Insufficient Logging and Monitoring

Python Web Developer (37 Courses, 21 Hours, 21 CPE Credits)

The Python Developer Learning Path is designed for those responsible for programming and development of web applications or applications that are run over HTTP from a web server to a web browser.

Python Web Developer learning path provide secure coding best practices and knowledge of platform configuration. Additionally, this path also provides learners with a thorough understanding of how to identify and mitigate vulnerabilities.

- AWA 101 Fundamentals of Application Security
- AWA 102 Secure Software Concepts
- COD 101 Fundamentals of Secure Development
- COD 153 Fundamentals of Secure Ajax Code

- COD 255 Creating Secure Code Web API Foundations
- COD 256 Creating Secure Code Ruby on Rails Foundations
- COD 257 Creating Secure Python Web Applications
- COD 259 Node.Js Threats and Vulnerabilities
- COD 361 Html5 Secure Threats
- COD 362 Html5 Built-In Security Features
- COD 363 Securing Html5 Data
- COD 364 Securing Html5 Connectivity
- DES 101 Fundamentals of Secure Architecture
- DES 202 Cryptographic Suite Services: Encoding, Encrypting and Hashing
- DES 203 Cryptographic Components Randomness, Algorithms, And Key Management
- DES 204 The Role of Cryptography in Application Development
- DES 205 Message Integrity Cryptographic Functions
- DES 212 Architecture Risk Analysis and Remediation
- DES 222 Applying OWASP 2017: Mitigating Injection
- DES 223 Applying OWASP 2017: Mitigating Broken Authentication
- DES 224 Applying OWASP 2017: Mitigating Sensitive Data Exposure
- DES 225 Applying OWASP 2017: Mitigating XML External Entities
- DES 226 Applying OWASP 2017: Mitigating Broken Access Control
- DES 227 Applying OWASP 2017: Mitigating Security Misconfiguration
- DES 228 Applying OWASP 2017: Mitigating Cross Site Scripting
- DES 229 Applying OWASP 2017: Mitigating Insecure Deserialization
- DES 230 Applying OWASP 2017: Mitigating Use of Components with Known Vulnerabilities
- DES 231 Applying OWASP 2017: Mitigating Insufficient Logging and Monitoring
- ENG 191 Introduction to The Microsoft SDL
- ENG 192 Implementing the Agile MS SDL
- ENG 193 Implementing the MS SDL Optimization Model
- ENG 194 Implementing MS SDL Line of Business
- ENG 195 Implementing the MS SDL Threat Modeling Tool
- ENG 205 Fundamentals of Threat Modeling
- ENG 211 How to Create Application Security Design Requirements
- ENG 311 Attack Surface Analysis and Reduction
- ENG 312 How to Perform A Security Code Review

Ruby on Rails Developer (35 Courses, 22 Hours, 22 CPE Credits)

The Ruby on Rails Developer Learning Path is designed for those responsible for writing server-side web application logic in Ruby, around the framework rails.

Ruby on Rails Developer learning paths provides best practices and techniques for secure application development with Ruby on rails. Learners will be able to understand various types of vulnerabilities, build strong session management, and prevent common vulnerabilities in Rails applications.

- AWA 101 Fundamentals of Application Security
- AWA 102 Secure Software Concepts
- COD 101 Fundamentals of Secure Development
- COD 153 Fundamentals of Secure Ajax Code
- COD 255 Creating Secure Code Web API Foundations
- COD 256 Creating Secure Code Ruby on Rails Foundations
- COD 257 Creating Secure Python Web Applications
- COD 259 Node.Js Threats and Vulnerabilities
- COD 281 Java Security Model
- COD 282 Java Authentication and Authorization Service (Jaas)
- COD 283 Java Cryptography
- COD 352 Creating Secure jQuery Code
- COD 361 Html5 Secure Threats
- COD 362 Html5 Built-In Security Features
- COD 363 Securing Html5 Data
- COD 364 Securing Html5 Connectivity
- DES 101 Fundamentals of Secure Architecture
- DES 202 Cryptographic Suite Services: Encoding, Encrypting and Hashing
- DES 203 Cryptographic Components Randomness, Algorithms, And Key Management
- DES 204 The Role of Cryptography in Application Development
- DES 205 Message Integrity Cryptographic Functions
- DES 212 Architecture Risk Analysis and Remediation
- DES 224 Applying OWASP 2017: Mitigating Sensitive Data Exposure
- DES 228 Applying OWASP 2017: Mitigating Cross Site Scripting
- ENG 191 Introduction to The Microsoft SDL
- ENG 192 Implementing the Agile MS SDL
- ENG 193 Implementing the MS SDL Optimization Model
- ENG 194 Implementing MS SDL Line of Business
- ENG 195 Implementing the MS SDL Threat Modeling Tool
- ENG 205 Fundamentals of Threat Modeling
- ENG 312 How to Perform A Security Code Review
- TST 224 Testing for OWASP 2017: Sensitive Data Exposure

- TST 228 Testing for OWASP 2017: Cross-Site Scripting

Web Developer (51 Courses, 27 Hours, 27 CPE Credits)

The Web Developer Learning Path is designed for those who are responsible for the development of web applications or applications that are run over HTTP from a web server to a web browser.

Web Developer learning path provides developers with a solid foundation of security features necessary to develop applications. Learners will develop an understanding of responsive web design, enterprise integration, and learn to protect data with security best practices.

- AWA 101 Fundamentals of Application Security
- AWA 102 Secure Software Concepts
- COD 101 Fundamentals of Secure Development
- COD 153 Fundamentals of Secure Ajax Code
- COD 255 Creating Secure Code Web API
- COD 256 Creating Secure Code Ruby on Rails Foundations
- COD 261 Threats to Scripts
- COD 262 Fundamentals of Secure Scripting
- COD 263 Secure Scripting with Perl, Python, Bash and Ruby
- COD 264 Protecting Sensitive Data While Scripting
- COD 315 Creating Secure PHP Code
- COD 361 Html5 Secure Threats
- COD 362 Html5 Built-In Security Features
- COD 363 Securing Html5 Data
- COD 364 Securing Html5 Connectivity
- DES 101 Fundamentals of Secure Architecture
- DES 202 Cryptographic Suite Services: Encoding, Encrypting and Hashing
- DES 203 Cryptographic Components Randomness, Algorithms, And Key Management
- DES 204 The Role of Cryptography in Application Development
- DES 205 Message Integrity Cryptographic Functions
- DES 212 Architecture Risk Analysis and Remediation
- DES 222 Applying OWASP 2017: Mitigating Injection
- DES 223 Applying OWASP 2017: Mitigating Broken Authentication
- DES 224 Applying OWASP 2017: Mitigating Sensitive Data Exposure
- DES 225 Applying OWASP 2017: Mitigating XML External Entities
- DES 226 Applying OWASP 2017: Mitigating Broken Access Control
- DES 227 Applying OWASP 2017: Mitigating Security Misconfiguration
- DES 228 Applying OWASP 2017: Mitigating Cross Site Scripting
- DES 229 Applying OWASP 2017: Mitigating Insecure Deserialization
- DES 230 Applying OWASP 2017: Mitigating Use of Components with Known Vulnerabilities
- DES 231 Applying OWASP 2017: Mitigating Insufficient Logging and Monitoring

- DES 311 Creating Secure Application Architecture
- ENG 191 Introduction to The Microsoft SDL
- ENG 192 Implementing the Agile MS SDL
- ENG 193 Implementing the MS SDL Optimization Model
- ENG 194 Implementing MS SDL Line of Business
- ENG 195 Implementing the MS SDL Threat Modeling Tool
- ENG 205 Fundamentals of Threat Modeling
- ENG 211 How to Create Application Security Design Requirements
- ENG 311 Attack Surface Analysis and Reduction
- ENG 312 How to Perform A Security Code Review
- TST 222 Testing for OWASP 2017: Injection
- TST 223 Testing for OWASP 2017: Broken Authentication
- TST 224 Testing for OWASP 2017: Sensitive Data Exposure
- TST 225 Testing for OWASP 2017: XML External Entities
- TST 226 Testing for OWASP 2017: Broken Access Control
- TST 227 Testing for OWASP 2017: Security Misconfiguration Testing for OWASP 2017: Security Misconfiguration
- TST 228 Testing for OWASP 2017: Cross-Site Scripting
- TST 229 Testing for OWASP 2017: Insecure Deserialization
- TST 230 Testing for OWASP 2017: Use of Components with Known Vulnerabilities
- TST 231 Testing for OWASP 2017: Insufficient Logging and Monitoring

Node.js Developer (36 Courses, 25 Hours, 25 CPE Credits)

The Node.js Developer Learning Path is designed for those responsible for managing the interchange of data between the server and the users.

The Node.js Developer learning path provides developers a solid foundation of security features necessary to code, test and operate Node.js based services. Learners will develop a working knowledge of web libraries, frameworks and the whole web stack while protecting data using secure coding best practices.

- AWA 101 Fundamentals of Application Security
- AWA 102 Secure Software Concepts
- COD 101 Fundamentals of Secure Development
- COD 153 Fundamentals of Secure Ajax Code
- COD 241 Creating Secure Oracle Database Applications
- COD 255 Creating Secure Code Web API Foundations
- COD 256 Creating Secure Code Ruby on Rails Foundations
- COD 259 Node.Js Threats and Vulnerabilities
- COD 311 Creating Secure Asp.Net MVC Applications
- COD 315 Creating Secure PHP Code
- COD 352 Creating Secure jQuery Code

- COD 361 Html5 Secure Threats
- COD 362 Html5 Built-In Security Features
- COD 363 Securing Html5 Data
- COD 364 Securing Html5 Connectivity
- DES 101 Fundamentals of Secure Architecture
- DES 202 Cryptographic Suite Services: Encoding, Encrypting and Hashing
- DES 203 Cryptographic Components Randomness, Algorithms, And Key Management
- DES 204 The Role of Cryptography in Application Development
- DES 205 Message Integrity Cryptographic Functions
- DES 212 Architecture Risk Analysis and Remediation
- DES 224 Applying OWASP 2017: Mitigating Sensitive Data Exposure
- DES 225 Applying OWASP 2017: Mitigating XML External Entities
- DES 228 Applying OWASP 2017: Mitigating Cross Site Scripting
- ENG 191 Introduction to The Microsoft SDL
- ENG 192 Implementing the Agile MS SDL
- ENG 193 Implementing the MS SDL Optimization Model
- ENG 194 Implementing MS SDL Line of Business
- ENG 195 Implementing the MS SDL Threat Modeling Tool
- ENG 205 Fundamentals of Threat Modeling
- ENG 211 How to Create Application Security Design Requirements
- ENG 311 Attack Surface Analysis and Reduction
- ENG 312 How to Perform A Security Code Review
- TST 224 Testing for OWASP 2017: Sensitive Data Exposure
- TST 225 Testing for OWASP 2017: XML External Entities
- TST 228 Testing for OWASP 2017: Cross-Site Scripting Testing for OWASP 2017: Cross-Site Scripting

Swift Developer (37 Courses, 18 Hours, 18 CPE Credits)

The Swift Developer Learning Path is designed for those responsible for the development of applications aimed towards iOS and OS X and the integration with back-end services.

The Swift Developer learning path explains how to identify common mobile application risks and utilize best practices for designing and building applications for iOS and OS X. The learning path is divided into RESTful API's, embedded databases, and object-oriented programming to provide learners with a solid foundation of security features needed to develop secure Swift code.

- AWA 101 Fundamentals of Application Security
- AWA 102 Secure Software Concepts
- COD 110 Fundamentals of Secure Mobile Development
- COD 229 Insecure IoT Mobile Interface
- COD 234 Mobile Threats and Mitigations

- COD 235 Defending Mobile Data with Cryptography
- COD 236 Mobile App Authentication and Authorization
- COD 237 Defending Mobile App Code
- COD 317 Creating Secure iOS Code in Swift
- DES 101 Fundamentals of Secure Architecture
- DES 202 Cryptographic Suite Services: Encoding, Encrypting and Hashing
- DES 203 Cryptographic Components Randomness, Algorithms, And Key Management
- DES 204 The Role of Cryptography in Application Development
- DES 205 Message Integrity Cryptographic Functions
- DES 212 Architecture Risk Analysis and Remediation
- ENG 112 Essential Access Control for Mobile Devices
- ENG 191 Introduction to The Microsoft SDL
- ENG 192 Implementing the Agile MS SDL
- ENG 193 Implementing the MS SDL Optimization Model
- ENG 194 Implementing MS SDL Line of Business
- ENG 195 Implementing the MS SDL Threat Modeling Tool
- ENG 205 Fundamentals of Threat Modeling
- ENG 211 How to Create Application Security Design Requirements
- ENG 311 Attack Surface Analysis and Reduction
- ENG 312 How to Perform A Security Code Review
- TST 252 Testing for OS Command Injection
- TST 257 Testing for Use of Hard-Coded Credentials
- TST 259 Testing for Unrestricted Upload of File with Dangerous Type
- TST 260 Testing for Reliance on Untrusted Inputs in A Security Decision
- TST 261 Testing for Execution with Unnecessary Privileges Testing for Execution with Unnecessary Privileges
- TST 264 Testing for Download of Code Without Integrity Check
- TST 266 Testing for Inclusion of Functionality from Untrusted Control Sphere
- TST 267 Testing for Incorrect Permission Assignment for Critical Resource
- TST 268 Testing for Use of a Potentially Dangerous Function
- TST 270 Testing for Incorrect Calculation of Buffer Size
- TST 271 Testing for Improper Restriction of Excessive Authentication Attempts
- TST 272 Testing for Open Redirect

Linux Administrator (18 Courses, 7 Hours, 7 CPE Credits)

The Linux Administrator Learning Path is designed for those responsible for maintaining and developing Linux Infrastructure technology.

Linux Administrator learning path will dive into operating system configuration and administration of virtual servers. Learners will develop working knowledge needed to support development, testing and

systems integration. Additionally, the learning path will provide learners with a solid understanding of secure development best practices.

- COD 261 Threats to Scripts Threats to Scripts
- COD 262 Fundamentals of Secure Scripting
- COD 263 Secure Scripting with Perl, Python, Bash and Ruby
- COD 264 Protecting Sensitive Data While Scripting
- DES 214 Secure Network Access
- DES 215 Securing Operating System Access
- DES 260 Fundamentals of IoT Architecture and Design
- ENG 110 Essential Account Management Security
- ENG 114 Essential Risk Assessment
- ENG 115 Essential System and Information Integrity
- ENG 119 Essential Security Audit and Accountability
- ENG 121 Essential Identification and Authentication
- ENG 191 Introduction to The Microsoft SDL
- ENG 192 Implementing the Agile MS SDL
- ENG 193 Implementing the MS SDL Optimization Model
- ENG 194 Implementing MS SDL Line of Business
- ENG 195 Implementing the MS SDL Threat Modeling Tool
- ENG 205 Fundamentals of Threat Modeling

Network Engineer (24 Courses,13 Hours, 13 CPE Credits)

The Network Engineer Learning Path is designed for those responsible for planning, implementing and overseeing computer networks that support in-house voice, data, video and wireless network services.

Network Engineer learning path provides managing systems best practices and services across all environments. This learning path dives into applying these best practices to improve the stability, security, efficiency, and scalability of environments. Learners will also develop working knowledge of how to create and modify scripts or applications to perform tasks.

- AWA 101 Fundamentals of Application Security
- AWA 102 Secure Software Concepts
- COD 110 Fundamentals of Secure Mobile Development
- COD 261 Threats to Scripts
- COD 262 Fundamentals of Secure Scripting
- COD 263 Secure Scripting with Perl, Python, Bash and Ruby
- COD 264 Protecting Sensitive Data While Scripting
- DES 214 Securing Network Access
- DES 215 Securing Operating System Access
- DES 216 Securing Cloud Instances
- DES 217 Application, Technical and Physical Access Controls
- DES 260 Fundamentals of IoT Architecture and Design

- ENG 110 Essential Account Management Security
- ENG 114 Essential Risk Assessment
- ENG 115 Essential System and Information Integrity
- ENG 119 Essential Security Audit and Accountability
- ENG 121 Essential Identification and Authentication
- ENG 191 Introduction to The Microsoft SDL
- ENG 192 Implementing the Agile MS SDL
- ENG 193 Implementing the MS SDL Optimization Model
- ENG 194 Implementing MS SDL Line of Business
- ENG 195 Implementing the MS SDL Threat Modeling Tool
- ENG 205 Fundamentals of Threat Modeling
- TST 101 Fundamentals of Security Testing

Application/Product Owner (24 Courses, 11 Hours, 11 CPE Credits)

The Product Owner Learning Path is designed for those responsible for setting, prioritizing, and evaluating the work generated by a software scrum team in order to ensure impeccable features and functionality of the product.

The Product Owner learning path introduces learners to the basics of application security and essentials goals and controls needed to create secure software and manage risk in the software development lifecycle.

- AWA 101 Fundamentals of Application Security
- AWA 102 Secure Software Concepts
- DES 212 Architecture Risk Analysis and Remediation
- DES 222 Applying OWASP 2017: Mitigating Injection
- DES 223 Applying OWASP 2017: Mitigating Broken Authentication
- DES 224 Applying OWASP 2017: Mitigating Sensitive Data Exposure
- DES 225 Applying OWASP 2017: Mitigating XML External Entities
- DES 226 Applying OWASP 2017: Mitigating Broken Access Control
- DES 227 Applying OWASP 2017: Mitigating Security Misconfiguration
- DES 228 Applying OWASP 2017: Mitigating Cross Site Scripting
- DES 229 Applying OWASP 2017: Mitigating Insecure Deserialization
- DES 230 Applying OWASP 2017: Mitigating Use of Components with Known Vulnerabilities
- DES 231 Applying OWASP 2017: Mitigating Insufficient Logging and Monitoring
- Dees 260 Fundamentals of IoT Architecture and Design
- ENG 124 Essential Application Protection
- ENG 125 Essential Data Protection
- ENG 191 Introduction to The Microsoft SDL
- ENG 192 Implementing the Agile MS SDL
- ENG 193 Implementing the MS SDL Optimization Model

- ENG 194 Implementing MS SDL Line of Business
- ENG 195 Implementing the MS SDL Threat Modeling Tool
- ENG 211 How to Create Application Security Design Requirements
- ENG 311 Attack Surface Analysis and Reduction Attack Surface Analysis and Reduction
- TST 101 Fundamentals of Security Testing

Project Manager (17 Courses,12 Hours, 12 CPE Credits)

The Project Manager learning path introduces project managers to the essentials of access control, configuration management, risk assessment, auditing and authentication.

Project Manager Learning Path provides the knowledge and skills necessary to ensure adherence to your organization's system and information security policies. Compliance with relevant governmental and industry standards is also addressed.

- AWA 101 Fundamentals of Application Security
- AWA 102 Secure Software Concepts
- COD 101 Fundamentals of Secure Development
- COD 141 Fundamentals of Secure Database Development
- COD 152 Fundamentals of Secure Cloud Development
- DES 101 Fundamentals of Secure Architecture
- DES 202 Cryptographic Suite Services: Encoding, Encrypting and Hashing
- DES 203 Cryptographic Components Randomness, Algorithms, And Key Management
- DES 204 The Role of Cryptography in Application Development
- DES 205 Message Integrity Cryptographic Functions
- DES 217 Application, Technical and Physical Access Control
- ENG 123 Essential Security Engineering Principles
- ENG 124 Essential Application Protection
- ENG 125 Essential Data Protection
- ENG 205 Fundamentals of Threat Modeling
- ENG 211 How to Create Application Security Design Requirements
- ENG 311 Attack Surface Analysis and Reduction

Quality Assurance (QA)/Test Engineer (59 Courses, 20 Hours, 20 CPE Credits)

The Quality Assurance (QA)/Test Engineer is designed for those responsible for assessing and testing the quality of specifications and technical design.

Quality Assurance (QA)/Test Engineer learning path provides software testers and quality assurance (QA) professionals with the knowledge and skills required to verify and assure application security standards are met. Learners will dive into applied testing techniques and best practices for planning and implementing strategies for quality management and testing.

- AWA 101 Fundamentals of Application Security

- AWA 102 Secure Software Concepts
- DES 202 Cryptographic Suite Services: Encoding, Encrypting and Hashing
- DES 203 Cryptographic Components Randomness, Algorithms, And Key Management
- DES 204 The Role of Cryptography in Application Development
- DES 205 Message Integrity Cryptographic Functions
- DES 217 Application, Technical and Physical Access Control
- DES 222 Applying OWASP 2017: Mitigating Injection
- DES 223 Applying OWASP 2017: Mitigating Broken Authentication
- DES 224 Applying OWASP 2017: Mitigating Sensitive Data Exposure
- DES 225 Applying OWASP 2017: Mitigating XML External Entities
- DES 226 Applying OWASP 2017: Mitigating Broken Access Control
- DES 227 Applying OWASP 2017: Mitigating Security Misconfiguration
- DES 228 Applying OWASP 2017: Mitigating Cross Site Scripting
- DES 229 Applying OWASP 2017: Mitigating Insecure Deserialization
- DES 230 Applying OWASP 2017: Mitigating Use of Components with Known Vulnerabilities
- DES 231 Applying OWASP 2017: Mitigating Insufficient Logging and Monitoring
- ENG 114 Essential Risk Assessment
- ENG 123 Essential Security Engineering Principles
- ENG 205 Fundamentals of Threat Modeling
- ENG 211 How to Create Application Security Design Requirements
- ENG 312 How to Perform A Security Code Review
- TST 101 Fundamentals of Security Testing
- TST 222 Testing for OWASP 2017: Injection
- TST 223 Testing for OWASP 2017: Broken Authentication
- TST 224 Testing for OWASP 2017: Sensitive Data Exposure
- TST 225 Testing for OWASP 2017: XML External Entities
- TST 226 Testing for OWASP 2017: Broken Access Control
- TST 227 Testing for OWASP 2017: Security Misconfiguration
- TST 228 Testing for OWASP 2017: Cross-Site Scripting
- TST 229 Testing for OWASP 2017: Insecure Deserialization
- TST 230 Testing for OWASP 2017: Use of Components with Known Vulnerabilities
- TST 231 Testing for OWASP 2017: Insufficient Logging and Monitoring
- TST 251 Testing for SQL Injection
- TST 252 Testing for OS Command Injection
- TST 253 Testing for Classic Buffer Overflow
- TST 254 Testing for Cross-Site Scripting
- TST 255 Testing for Missing Authentication for Critical Function
- TST 256 Testing for Missing Authorization
- TST 257 Testing for Use of Hard-Coded Credentials
- TST 258 Testing for Missing Encryption of Sensitive Data
- TST 259 Testing for Unrestricted Upload of File with Dangerous Type

- TST 260 Testing for Reliance on Untrusted Inputs in A Security Decision
- TST 261 Testing for Execution with Unnecessary Privileges
- TST 262 Testing for Cross-Site Request Forgery
- TST 263 Testing for Path Traversal
- TST 264 Testing for Download of Code Without Integrity Check
- TST 265 Testing for Incorrect Authorization
- TST 266 Testing for Inclusion of Functionality from Untrusted Control Sphere
- TST 267 Testing for Incorrect Permission Assignment for Critical Resource
- TST 268 Testing for Use of a Potentially Dangerous Function
- TST 269 Testing for Use of a Broken or Risky Cryptographic Algorithm
- TST 270 Testing for Incorrect Calculation of Buffer Size
- TST 271 Testing for Improper Restriction of Excessive Authentication Attempts
- TST 272 Testing for Open Redirect
- TST 273 Testing for Uncontrolled Format String
- TST 274 Testing for Integer Overflow or Wraparound
- TST 275 Testing for Use of a One-Way Hash Without A Salt

Software Architect (48 Courses, 22 Hours, 22 CPE Credits)

The Software Architect Learning Path is designed for those making design choices, coordinating and overseeing technical standards and includes software coding standards, tools, and platforms.

Software Architect learning path provides architects with working knowledge of secure software architecture best practices. They will learn to apply these best practices to requirements, design, and implementation phases of the software development lifecycle. Courses will emphasize early phases of the SDLC and defensive coding techniques. Additionally, they will learn how to build software to avoid systemic issues found in insecure software.

- AWA 101 Fundamentals of Application Security
- AWA 102 Secure Software Concepts
- COD 141 Fundamentals of Secure Database Development
- COD 225 Insecure IoT Web Interfaces
- COD 226 Insecure IoT Authentication and Authorization
- COD 227 Insecure IoT Network Services
- COD 228 Insecure IoT Communications
- COD 229 Insecure IoT Mobile Interface
- COD 230 Insecure IoT Firmware
- COD 261 Threats to Scripts
- DES 101 Fundamentals of Secure Architecture
- DES 202 Cryptographic Suite Services: Encoding, Encrypting and Hashing
- DES 203 Cryptographic Components Randomness, Algorithms, And Key Management
- DES 204 The Role of Cryptography in Application Development
- DES 205 Message Integrity Cryptographic Functions

- DES 212 Architecture Risk Analysis and Remediation
- DES 214 Securing Network Access
- DES 215 Securing Operating System Access
- DES 216 Securing Cloud Instances
- DES 217 Application, Technical and Physical Access Controls
- DES 222 Applying OWASP 2017: Mitigating Injection
- DES 223 Applying OWASP 2017: Mitigating Broken Authentication
- DES 224 Applying OWASP 2017: Mitigating Sensitive Data Exposure
- DES 225 Applying OWASP 2017: Mitigating XML External Entities
- DES 226 Applying OWASP 2017: Mitigating Broken Access Control
- DES 227 Applying OWASP 2017: Mitigating Security Misconfiguration
- DES 228 Applying OWASP 2017: Mitigating Cross Site Scripting
- DES 229 Applying OWASP 2017: Mitigating Insecure Deserialization
- DES 230 Applying OWASP 2017: Mitigating Use of Components with Known Vulnerabilities
- DES 231 Applying OWASP 2017: Mitigating Insufficient Logging and Monitoring
- DES 260 Fundamentals of IoT Architecture and Design
- DES 311 Creating Secure Application Architecture
- ENG 211 How to Create Application Security Design Requirements
- ENG 311 Attack Surface Analysis and Reduction
- ENG 312 How to Perform A Security Code Review
- TST 255 Testing for Testing for Missing Authentication for Critical Function
- TST 257 Testing for Use of Hard-Coded Credentials
- TST 259 Testing for Unrestricted Upload of File with Dangerous Type
- TST 260 Testing for Reliance on Untrusted Inputs in A Security Decision
- TST 261 Testing for Execution with Unnecessary Privileges
- TST 264 Testing for Download of Code Without Integrity Check
- TST 266 Testing for Inclusion of Functionality from Untrusted Control Sphere
- TST 267 Testing for Incorrect Permission Assignment for Critical Resource
- TST 268 Testing for Use of a Potentially Dangerous Function
- TST 271 Testing for Improper Restriction of Excessive Authentication Attempts
- TST 272 Testing for Open Redirect
- TST 273 Testing for Uncontrolled Format String

DevOps Engineer (16 Courses, 9 Hours, 9 CPE Credits)

The DevOps Engineer Learning Path is designed for those who work closely with Software Engineers to help them deploy and operate different systems.

The DevOps Engineer learning path provides learners with a solid foundation of security features necessary to automate and streamline operations and processes while keeping security top of mind. Learners will apply best practices to develop new features and write scripts across various technologies.

- COD 101 Fundamentals of Secure Development

- DES 101 Fundamentals of Secure Architecture
- DES 214 Securing Network Access
- DES 215 Securing Operating System Access
- DES 216 Securing Cloud Instances
- DES 217 Application, Technical and Physical Access Controls
- ENG 123 Essential Security Engineering Principles
- ENG 124 Essential Application Protection
- ENG 125 Essential Data Protection
- ENG 191 Introduction to The Microsoft SDL
- ENG 192 Implementing the Agile MS SDL
- ENG 193 Implementing the MS SDL Optimization Model
- ENG 194 Implementing MS SDL Line of Business
- ENG 195 Implementing the MS SDL Threat Modeling Tool
- ENG 205 Fundamentals of Threat Modeling
- TST 101 Fundamentals of Security Testing

Business Analyst (7 Courses, 4 Hours, 4 CPE Credits)

The Business Analyst Learning Path is designed for those responsible for defining, analyzing and documenting requirements in the software development lifecycle.

Business Analyst learning path provides learners with the knowledge and skills necessary to ensure adherence to system and information security policies as well as compliance with relevant governmental and industry standards. This learning path also introduces learners to the essentials of access control, configuration management, risk assessment, auditing and authentication.

- AWA 101 Fundamentals of Application Security
- AWA 102 Secure Software Concepts
- DES 101 Fundamentals of Secure Architecture
- ENG 114 Essential Risk Assessment
- ENG 116 Essential Security Planning Policy and Procedures
- ENG 117 Essential Information Security Program Planning
- ENG 211 How to Create Application Security Design Requirements

Automation Engineer (36 Courses, 9 Hours, 9 CPE Credits)

The Automation Engineer Learning Path is designed for those who design, program, simulate and test automated machinery and processes in order to complete exact tasks.

Automation Engineer learning path introduces learners to essential goals and controls needed to create secure software and manage risk in the software development lifecycle. Courses will also expose learners to cryptography, handling input and output and the OWASP Top Ten.

- DES 222 Applying OWASP 2017: Mitigating Injection

- DES 223 Applying OWASP 2017: Mitigating Broken Authentication
- DES 224 Applying OWASP 2017: Mitigating Sensitive Data Exposure
- DES 225 Applying OWASP 2017: Mitigating XML External Entities
- DES 226 Applying OWASP 2017: Mitigating Broken Access Control
- DES 227 Applying OWASP 2017: Mitigating Security Misconfiguration
- DES 228 Applying OWASP 2017: Mitigating Cross Site Scripting
- DES 229 Applying OWASP 2017: Mitigating Insecure Deserialization
- DES 230 Applying OWASP 2017: Mitigating Use of Components with Known Vulnerabilities
- DES 231 Applying OWASP 2017: Mitigating Insufficient Logging and Monitoring
- ENG 110 Essential Account Management Security
- ENG 113 Essential Secure Configuration Management
- ENG 114 Essential Risk Assessment
- ENG 119 Essential Security Audit and Accountability
- ENG 120 Essential Security Assessment and Authorization
- ENG 123 Essential Security Engineering Principles
- ENG 124 Essential Application Protection
- ENG 125 Essential Data Protection
- TST 252 Testing for OS Command Injection
- TST 253 Testing for Classic Buffer Overflow
- TST 255 Testing for Missing Authentication for Critical Function
- TST 257 Testing for Use of Hard-Coded Credentials
- TST 258 Testing for Missing Encryption of Sensitive Data
- TST 259 Testing for Unrestricted Upload of File with Dangerous Type
- TST 260 Testing for Reliance on Untrusted Inputs in A Security Decision
- TST 261 Testing for Execution with Unnecessary Privileges
- TST 264 Testing for Download of Code Without Integrity Check
- TST 266 Testing for Inclusion of Functionality from Untrusted Control Sphere
- TST 267 Testing for Incorrect Permission Assignment for Critical Resource
- TST 268 Testing for Use of a Potentially Dangerous Function
- TST 269 Testing for Use of a Broken or Risky Cryptographic Algorithm
- TST 270 Testing for Incorrect Calculation of Buffer Size
- TST 271 Testing for Improper Restriction of Excessive Authentication Attempts
- TST 272 Testing for Open Redirect
- TST 273 Testing for Uncontrolled Format String
- TST 275 Testing for Use of a One-Way Hash without a Salt

Cyber Security Professional (15 Courses, 6 Hours, 6 CPE Credits)

The Cybersecurity Professional Learning Path is designed for those tasked with everything from the technical aspects of security, security policy and everything in between.

The Cybersecurity learning path provides learners with fundamental security skills required to develop and design security devices and software. Learners will explore how to manage security measures, operate inspections of systems and process, initiate security and safety measures, and maintain policies and procedures, and information security and privacy best practices.

- AWA 008 Information Privacy – Classifying Data
- AWA 009 Information Privacy – Protecting Data
- AWA 010 Email Security
- AWA 012 Malware Awareness
- AWA 013 Mobile Security
- AWA 014 Password Security
- AWA 016 Phishing Awareness
- AWA 018 Social Engineering Awareness
- AWA 019 Travel Security
- AWA 101 Fundamentals of Application Security
- AWA 102 Secure Software Concepts
- ENG 117 Essential Information Security Program Planning
- ENG 118 Essential Incident Response
- ENG 124 Essential Application Protection
- TST 101 Fundamentals of Software Security Testing

Operations/IT Manager (23 Courses, 8 Hours, 8 CPE Credits)

The Operations/IT Learning Path is designed for those responsible for managing existing operations and implementing new operations processes, managing expectations across stakeholder groups, sharing responsibility for project success, and managing day-to-day IT processes.

Operations/IT learning path introduces learners to the basics of application security and essential goals and controls needed to manage the development of secure software. Courses will also explore management of risks associated with the software development lifecycle while diving into developing, implementing, and ensuring compliance with operational application security policies and procedures.

- DES 214 Securing Network Access
- DES 215 Securing Operating System Access
- DES 216 Securing Cloud Instances
- DES 217 Application, Technical and Physical Access Controls
- ENG 110 Essential Account Management Security
- ENG 111 Essential Session Management Security
- ENG 112 Essential Access Control for Mobile Devices
- ENG 113 Essential Secure Configuration Management
- ENG 114 Essential Risk Assessment
- ENG 115 Essential System and Information Integrity
- ENG 116 Essential Security Planning Policy and Procedures
- ENG 117 Essential Information Security Program Planning

- ENG 118 Essential Incident Response
- ENG 119 Essential Security Audit and Accountability
- ENG 120 Essential Security Assessment and Authorization
- ENG 121 Essential Identification and Authentication
- ENG 122 Essential Physical and Environmental Protection
- ENG 123 Essential Security Engineering Principles
- ENG 124 Essential Application Protection
- ENG 125 Essential Data Protection
- ENG 126 Essential Security Maintenance Policies
- ENG 127 Essential Media Protection
- ENG 205 Fundamentals of Threat Modeling

Systems Analyst (37 Courses, 12 Hours, 12 CPE Credits)

Our Systems Analyst Learning Path is designed for those who specialize in the implementation of computer system requirements by defining and analyzing system problems; designing and testing standards and solutions.

Systems Analyst learning path provides learners fundamental knowledge required to secure networks and systems. This learning path is designed to present a holistic approach to network and system security with an exploration of controls, monitoring access, operational procedure and formal auditing and logging.

- AWA 101 Fundamentals of Application Security
- AWA 102 Secure Software Concepts
- DES 222 Applying OWASP 2017: Mitigating Injection
- DES 223 Applying OWASP 2017: Mitigating Broken Authentication
- DES 224 Applying OWASP 2017: Mitigating Sensitive Data Exposure
- DES 225 Applying OWASP 2017: Mitigating XML External Entities
- DES 226 Applying OWASP 2017: Mitigating Broken Access Control
- DES 227 Applying OWASP 2017: Mitigating Security Misconfiguration
- DES 228 Applying OWASP 2017: Mitigating Cross Site Scripting
- DES 229 Applying OWASP 2017: Mitigating Insecure Deserialization
- DES 230 Applying OWASP 2017: Mitigating Use of Components with Known Vulnerabilities
- DES 231 Applying OWASP 2017: Mitigating Insufficient Logging and Monitoring
- ENG 110 Essential Account Management Security
- ENG 111 Essential Session Management Security
- ENG 112 Essential Access Control for Mobile Devices
- ENG 113 Essential Secure Configuration Management
- ENG 114 Essential Risk Assessment
- ENG 115 Essential System and Information Integrity
- ENG 116 Essential Security Planning Policy and Procedures

- ENG 117 Essential Information Security Program Planning
- ENG 118 Essential Incident Response
- ENG 119 Essential Security Audit and Accountability
- ENG 120 Essential Security Assessment and Authorization
- ENG 121 Essential Identification and Authentication
- ENG 122 Essential Physical and Environmental Protection
- ENG 123 Essential Security Engineering Principles
- ENG 124 Essential Application Protection
- ENG 125 Essential Data Protection
- ENG 126 Essential Security Maintenance Policies
- ENG 127 Essential Media Protection
- ENG 191 Introduction to The Microsoft SDL
- ENG 192 Implementing the Agile MS SDL
- ENG 193 Implementing the MS SDL Optimization Model
- ENG 194 Implementing MS SDL Line of Business
- ENG 195 Implementing the MS SDL Threat Modeling Tool
- ENG 205 Fundamentals of Threat Modeling
- ENG 211 How to Create Application Security Design Requirements

Systems Administrator (38 Courses, 16 Hours, 16 CPE Credits)

The Systems Administrator Learning Path is designed for those responsible for preventing and mitigating security breaches that may arise within computer systems.

Systems Administrator learning path provides learners with fundamental knowledge necessary to secure networks and systems. This learning path is designed to present a holistic approach to network and system security with an exploration of controls, monitoring access, operational procedure and formal auditing and logging.

- AWA 101 Fundamentals of Application Security
- AWA 102 Secure Software Concepts
- COD 141 Fundamentals of Secure Database Development
- COD 219 Creating Secure Code SAP ABAP Foundations
- COD 261 Threats to Scripts
- COD 262 Fundamentals of Secure Scripting
- COD 263 Secure Scripting with Perl, Python, Bash and Ruby
- COD 264 Protecting Sensitive Data while Scripting
- DES 214 Securing Network Access
- DES 215 Securing Operating System Access
- DES 216 Securing Cloud Instances
- DES 217 Application, Technical and Physical Access Controls
- DES 222 Applying OWASP 2017: Mitigating Injection
- DES 223 Applying OWASP 2017: Mitigating Broken Authentication

- DES 224 Applying OWASP 2017: Mitigating Sensitive Data Exposure
- DES 225 Applying OWASP 2017: Mitigating XML External Entities
- DES 226 Applying OWASP 2017: Mitigating Broken Access Control
- DES 227 Applying OWASP 2017: Mitigating Security Misconfiguration
- DES 228 Applying OWASP 2017: Mitigating Cross Site Scripting
- DES 229 Applying OWASP 2017: Mitigating Insecure Deserialization
- DES 230 Applying OWASP 2017: Mitigating Use of Components with Known Vulnerabilities
- DES 231 Applying OWASP 2017: Mitigating Insufficient Logging and Monitoring
- ENG 110 Essential Account Management Security
- ENG 111 Essential Session Management Security
- ENG 113 Essential Secure Configuration Management
- ENG 118 Essential Incident Response
- ENG 119 Essential Security Audit and Accountability
- ENG 121 Essential Identification and Authentication
- ENG 122 Essential Physical and Environmental Protection
- ENG 123 Essential Security Engineering Principles
- ENG 125 Essential Data Protection
- ENG 127 Essential Media Protection
- ENG 191 Introduction to The Microsoft SDL
- ENG 192 Implementing the Agile MS SDL
- ENG 193 Implementing the MS SDL Optimization Model
- ENG 194 Implementing MS SDL Line of Business
- ENG 195 Implementing the MS SDL Threat Modeling Tool
- ENG 205 Fundamentals of Threat Modeling

Application Security Champion (24 Courses, 12 Hours, 12 CPE Credits)

Our Application Security Champion Learning Path is designed for those chartered with driving a culture of “Security Built-in” to the software development lifecycle.

The Application Security Champion learning path exposes learners to concepts around all aspects of security including privacy, secure development and architecture, security testing, threat modeling, cryptography and cyber threat analysis and remediation.

- AWA 101 Fundamentals of Application Security
- AWA 102 Secure Software Concepts
- COD 101 Fundamentals of Secure Development
- DES 212 Architecture Risk Analysis and Remediation
- DES 222 Applying OWASP 2017: Mitigating Injection
- DES 223 Applying OWASP 2017: Mitigating Broken Authentication
- DES 224 Applying OWASP 2017: Mitigating Sensitive Data Exposure
- DES 225 Applying OWASP 2017: Mitigating XML External Entities

- DES 226 Applying OWASP 2017: Mitigating Broken Access Control
- DES 227 Applying OWASP 2017: Mitigating Security Misconfiguration
- DES 228 Applying OWASP 2017: Mitigating Cross Site Scripting
- DES 229 Applying OWASP 2017: Mitigating Insecure Deserialization
- DES 230 Applying OWASP 2017: Mitigating Use of Components with Known Vulnerabilities
- DES 231 Applying OWASP 2017: Mitigating Insufficient Logging and Monitoring
- ENG 124 Essential Application Protection
- ENG 125 Essential Data Protection
- ENG 191 Introduction to The Microsoft SDL
- ENG 192 Implementing the Agile MS SDL
- ENG 193 Implementing the MS SDL Optimization Model
- ENG 194 Implementing MS SDL Line of Business
- ENG 195 Implementing the MS SDL Threat Modeling Tool
- ENG 211 How to Create Application Security Design Requirements
- ENG 311 Attack Surface Analysis and Reduction
- TST 101 Fundamentals of Security Testing

Database Administrator (46 Courses, 23 Hours, 23 CPE Credits)

The Database Administrator Learning Path is designed for those responsible for capacity planning, installation, configuration, database design, migration, performance monitoring, security, troubleshooting, as well as back end data recovery.

Database Administrator learning path provides learners with fundamental knowledge of secure database development and the common database attacks that can be used to cause significant loss to an organization. Courses will also dive into platform-specific threats and secure coding best practices.

- AWA 101 Fundamentals of Application Security
- AWA 102 Secure Software Concepts
- COD 141 Fundamentals of Secure Database Development
- COD 241 Creating Secure Code - Oracle Database Applications
- COD 242 Creating Secure Code - SQL Server Foundations
- COD 254 Creating Secure Cloud Code - Azure Foundations
- COD 261 Threats to Scripts
- COD 262 Fundamentals of Secure Scripting
- COD 352 Creating Secure jQuery Code
- DES 101 Fundamentals of Secure Architecture
- DES 202 Cryptographic Suite Services: Encoding, Encrypting and Hashing
- DES 203 Cryptographic Components Randomness, Algorithms, And Key Management
- DES 204 The Role of Cryptography in Application Development
- DES 205 Message Integrity Cryptographic Functions
- DES 212 Architecture Risk Analysis and Remediation

- DES 222 Applying OWASP 2017: Mitigating Injection
- DES 223 Applying OWASP 2017: Mitigating Broken Authentication
- DES 224 Applying OWASP 2017: Mitigating Sensitive Data Exposure
- DES 225 Applying OWASP 2017: Mitigating XML External Entities
- DES 226 Applying OWASP 2017: Mitigating Broken Access Control
- DES 227 Applying OWASP 2017: Mitigating Security Misconfiguration
- DES 228 Applying OWASP 2017: Mitigating Cross Site Scripting
- DES 229 Applying OWASP 2017: Mitigating Insecure Deserialization
- DES 230 Applying OWASP 2017: Mitigating Use of Components with Known Vulnerabilities
- DES 231 Applying OWASP 2017: Mitigating Insufficient Logging and Monitoring
- ENG 191 Introduction to The Microsoft SDL
- ENG 192 Implementing the Agile MS SDL
- ENG 193 Implementing the MS SDL Optimization Model
- ENG 194 Implementing MS SDL Line of Business
- ENG 195 Implementing the MS SDL Threat Modeling Tool
- ENG 205 Fundamentals of Threat Modeling
- ENG 211 How to Create Application Security Design Requirements
- ENG 311 Attack Surface Analysis and Reduction
- ENG 312 How to Perform a Security Code Review
- TST 255 Testing for Missing Authentication for Critical Function
- TST 257 Testing for Use of Hard-Coded Credentials
- TST 259 Testing for Unrestricted Upload of File with Dangerous Type
- TST 260 Testing for Reliance on Untrusted Inputs in A Security Decision
- TST 261 Testing for Execution with Unnecessary Privileges
- TST 264 Testing for Download of Code Without Integrity Check
- TST 266 Testing for Inclusion of Functionality from Untrusted Control Sphere
- TST 267 Testing for Incorrect Permission Assignment for Critical Resource
- TST 268 Testing for Use of a Potentially Dangerous Function
- TST 271 Testing for Improper Restriction of Excessive Authentication Attempts
- TST 272 Testing for Open Redirect
- TST 273 Testing for Uncontrolled Format String

Information Security Specialist (38 Courses, 21 Hours, 21 CPE Credits)

The Information Security Specialist Learning Path is designed for those responsible for protecting systems, defining access privileges, control structures, and resources.

Information Security Specialist learning path provides learners with the knowledge and skills required to identify, protect, detect and recover from risks, vulnerabilities, and threats to the secure of information and/or data.

- AWA 101 Fundamentals of Application Security

- AWA 102 Secure Software Concepts
- COD 141 Fundamentals of Secure Database Development
- COD 222 PCI DSS v3.2 Best Practices for Developers
- COD 234 Mobile Threats and Mitigations
- COD 241 Creating Secure Code Oracle Foundations
- COD 242 Creating Secure Code SQL Server Foundations
- COD 256 Creating Secure Code Ruby on Rails Foundations
- COD 261 Threats to Scripts
- DES 212 Architecture Risk Analysis and Remediation
- ENG 110 Essential Account Management Security
- ENG 111 Essential Session Management Security
- ENG 112 Essential Access Control for Mobile Devices
- ENG 113 Essential Secure Configuration Management
- ENG 114 Essential Risk Assessment
- ENG 115 Essential System and Information Integrity
- ENG 116 Essential Security Planning Policy and Procedures
- ENG 117 Essential Information Security Program Planning
- ENG 118 Essential Incident Response
- ENG 119 Essential Security Audit and Accountability
- ENG 120 Essential Security Assessment and Authorization
- ENG 121 Essential Identification and Authentication
- ENG 122 Essential Physical and Environmental Protection
- ENG 123 Essential Security Engineering Principles
- ENG 124 Essential Application Protection
- ENG 125 Essential Data Protection
- ENG 126 Essential Security Maintenance Policies
- ENG 127 Essential Media Protection
- ENG 191 Introduction to The Microsoft SDL
- ENG 192 Implementing the Agile MS SDL
- ENG 193 Implementing the MS SDL Optimization Model
- ENG 194 Implementing MS SDL Line of Business
- ENG 195 Implementing the MS SDL Threat Modeling Tool
- ENG 205 Fundamentals of Threat Modeling
- ENG 211 How to Create Application Security Design Requirements
- ENG 311 Attack Surface Analysis and Reduction
- ENG 312 How to Perform a Security Code Review
- TST 101 Fundamentals of Security Testing

Embedded QA/Test Engineer (41 Courses, 16 Hours, 16 CPE Credits)

The Embedded QA/Test Engineer Learning Path is designed for those responsible for verifying and assuring application security of embedded systems.

Embedded QA/Test Engineer learning path provides learners with a solid understanding of applied testing techniques and a well-rounded base of knowledge to perform their tasks. This path also explores security best practices for conducting penetration tests and vulnerability assessment activities on embedded systems.

- AWA 101 Fundamentals of Application Security
- AWA 102 Secure Software Concepts
- DES 260 Fundamentals of IoT Architecture and Design
- ENG 114 Essential Risk Assessment
- ENG 123 Essential Security Engineering Principles
- ENG 191 Introduction to The Microsoft SDL
- ENG 192 Implementing the Agile MS SDL
- ENG 193 Implementing the MS SDL Optimization Model
- ENG 194 Implementing MS SDL Line of Business
- ENG 195 Implementing the MS SDL Threat Modeling Tool
- ENG 205 Fundamentals of Threat Modeling
- ENG 211 How to Create Application Security Design Requirements
- ENG 311 Attack Surface Analysis and Reduction
- ENG 312 How to Perform A Security Code Review
- TST 101 Fundamentals of Security Testing
- TST 222 Testing for OWASP 2017: Injection
- TST 223 Testing for OWASP 2017: Broken Authentication
- TST 224 Testing for OWASP 2017: Sensitive Data Exposure
- TST 225 Testing for OWASP 2017: XML External Entities
- TST 226 Testing for OWASP 2017: Broken Access Control
- TST 227 Testing for OWASP 2017: Security Misconfiguration
- TST 228 Testing for OWASP 2017: Cross-Site Scripting
- TST 229 Testing for OWASP 2017: Insecure Deserialization
- TST 230 Testing for OWASP 2017: Use of Components with Known Vulnerabilities
- TST 231 Testing for OWASP 2017: Insufficient Logging and Monitoring
- TST 253 Testing for Classic Buffer Overflow
- TST 256 Testing for Missing Authorization Testing for Missing Authorization
- TST 257 Testing for Use of Hard-Coded Credentials
- TST 258 Testing for Missing Encryption of Sensitive Data
- TST 259 Testing for Unrestricted Upload of File with Dangerous Type
- TST 260 Testing for Reliance on Untrusted Inputs in A Security Decision
- TST 261 Testing for Execution with Unnecessary Privileges
- TST 262 Testing for Cross-Site Request Forgery
- TST 264 Testing for Download of Code Without Integrity Check
- TST 266 Testing for Inclusion of Functionality from Untrusted Control Sphere
- TST 267 Testing for Incorrect Permission Assignment for Critical Resource
- TST 268 Testing for Use of a Potentially Dangerous Function
- TST 270 Testing for Incorrect Calculation of Buffer Size

- TST 273 Testing for Uncontrolled Format String
- TST 274 Testing for Integer Overflow or Wraparound
- TST 275 Testing for Use of One-Way Hash Without A Salt

Systems Leadership (13 Courses, 6 Hours, 6 CPE Credits)

The Systems Leadership Learning Path is designed for those responsible for computers and their complex operating systems.

Systems Leadership learning path provides learners with comprehensive baseline application security knowledge necessary for leading application development and design projects. Courses explore application security best practices necessary to ensure strategies and plans support business needs and align with departmental and organizational objectives and goals.

- AWA 101 Fundamentals of Application Security
- AWA 102 Secure Software Concepts
- DES 222 Applying OWASP 2017: Mitigating Injection
- DES 223 Applying OWASP 2017: Mitigating Broken Authentication
- DES 224 Applying OWASP 2017: Mitigating Sensitive Data Exposure
- DES 225 Applying OWASP 2017: Mitigating XML External Entities
- DES 226 Applying OWASP 2017: Mitigating Broken Access Control
- DES 227 Applying OWASP 2017: Mitigating Security Misconfiguration
- DES 228 Applying OWASP 2017: Mitigating Cross Site Scripting
- DES 229 Applying OWASP 2017: Mitigating Insecure Deserialization
- DES 230 Applying OWASP 2017: Mitigating Use of Components with Known Vulnerabilities
- DES 231 Applying OWASP 2017: Mitigating Insufficient Logging and Monitoring
- DES 311 Creating Secure Application Architecture

Development Manager (14 Courses, 7 Hours, 7 CPE Credits)

The Development Manager Learning Path is designed for those responsible for planning, preparing and ensuring that projects are completed.

Development Manager learning path introduces learners to application security best practices required to adhere to system and information security policies and compliance. Learners will apply these best practices to requirements, design, and implementation phases of the software development lifecycle.

- AWA 101 Fundamentals of Application Security
- AWA 102 Secure Software Concepts
- DES 101 Fundamentals of Secure Architecture
- DES 260 Fundamentals of IoT Architecture and Design
- ENG 110 Essential Account Management Security
- ENG 114 Essential Risk Assessment

- ENG 117 Essential Information Security Program Planning
- ENG 191 Introduction to The Microsoft SDL
- ENG 192 Implementing the Agile MS SDL
- ENG 193 Implementing the MS SDL Optimization Model
- ENG 194 Implementing MS SDL Line of Business
- ENG 195 Implementing the MS SDL Threat Modeling Tool
- ENG 205 Fundamentals of Threat Modeling
- ENG 211 How to Create Application Security Design Requirements

Cloud Developer (45 Courses, 25 Hours, 25 CPE Credits)

The Cloud Developer Learning Path is designed for those responsible for the design, development, and deployment of cloud applications.

Cloud Developer learning path provides learners with a clear understand of the risk associated with cloud computing. Learners will explore coverage of "Big Data", cloud computing characteristics, service and deployment models, and regulatory requirements. Courses will also dive into platform-specific secure coding best practices, including AWS and/or Azure.

- AWA 101 Fundamentals of Application Security
- AWA 102 Secure Software Concepts
- COD 152 Fundamentals of Secure Cloud Development
- COD 225 Insecure IoT Web Interfaces
- COD 226 Insecure IoT Authentication and Authorization
- COD 227 Insecure IoT Network Services
- COD 228 Insecure IoT Communications
- COD 229 Insecure IoT Mobile Interface
- COD 230 Insecure IoT Firmware
- COD 241 Creating Secure Oracle Database Applications
- COD 253 Creating Secure Cloud Code - AWS Foundations
- COD 254 Creating Secure Azure Applications
- COD 255 Creating Secure Code Web API Foundations
- COD 259 Node.js Threats and Vulnerabilities
- COD 261 Threats to Scripts
- DES 101 Fundamentals of Secure Architecture
- DES 202 Cryptographic Suite Services: Encoding, Encrypting and Hashing
- DES 203 Cryptographic Components Randomness, Algorithms, And Key Management
- DES 204 The Role of Cryptography in Application Development
- DES 205 Message Integrity Cryptographic Functions
- DES 214 Securing Network Access
- DES 215 Securing Operating System Access
- DES 216 Securing Cloud Instances
- DES 217 Application, Technical and Physical Access Controls

- DES 222 Applying OWASP 2017: Mitigating Injection
- DES 223 Applying OWASP 2017: Mitigating Broken Authentication
- DES 224 Applying OWASP 2017: Mitigating Sensitive Data Exposure
- DES 225 Applying OWASP 2017: Mitigating XML External Entities
- DES 226 Applying OWASP 2017: Mitigating Broken Access Control
- DES 227 Applying OWASP 2017: Mitigating Security Misconfiguration
- DES 228 Applying OWASP 2017: Mitigating Cross Site Scripting
- DES 229 Applying OWASP 2017: Mitigating Insecure Deserialization
- DES 230 Applying OWASP 2017: Mitigating Use of Components with Known Vulnerabilities
- DES 231 Applying OWASP 2017: Mitigating Insufficient Logging and Monitoring
- DES 311 Creating Secure Application Architecture
- ENG 191 Introduction to The Microsoft SDL
- ENG 192 Implementing the Agile MS SDL
- ENG 193 Implementing the MS SDL Optimization Model
- ENG 194 Implementing MS SDL Line of Business
- ENG 195 Implementing the MS SDL Threat Modeling Tool
- ENG 205 Fundamentals of Threat Modeling
- ENG 211 How to Create Application Security Design Requirements
- ENG 311 Attack Surface Analysis and Reduction
- ENG 312 How to Perform a Security Code Review

Automotive Developer (15 Courses, 13 Hours, 13 CPE Credits)

The Automotive Developer Learning Path is designed for those responsible for the various engineering fields that are applied in the design, development and production of automotive vehicle applications.

Automotive Developer learning path provides automotive embedded systems professionals with the knowledge and skills required to deploy security throughout the development process from design to deployment. Courses will dive into creating secure embedded code, threat modeling, key management, encrypting sensitive data and securing communication channels.

- AWA 101 Fundamentals of Application Security
- AWA 102 Secure Software Concepts
- COD 101 Fundamentals of Secure Development
- COD 160 Fundamentals of Secure Embedded Software Development
- DES 101 Fundamentals of Secure Architecture
- DES 202 Cryptographic Suite Services: Encoding, Encrypting and Hashing
- DES 203 Cryptographic Components Randomness, Algorithms, And Key Management
- DES 204 The Role of Cryptography in Application Development
- DES 205 Message Integrity Cryptographic Functions
- DES 212 Architecture Risk Analysis and Remediation

- DES 260 Fundamentals of IoT Architecture and Design
- DES 352 Creating Secure Ota (Over the Air) Automotive System Updates
- ENG 211 How to Create Application Security Design Requirements
- ENG 311 Attack Surface Analysis and Reduction
- ENG 312 How to Perform A Security Code Review

PCI Developer (58 Courses, 26 Hours, 26 CPE Credits)

The PCI Developer Learning Path is designed for those responsible for developing applications that process credit and debit card payments and/or any type of cardholder data.

PCI Developer learning path provides learners with the tools required to meet the Payment Card Industry Data Security Standards (PCI DSS) for systems that transmit, process, and/or store cardholder data. Courses provide a framework for developing secure applications, explain testing procedures and provide guidance for mitigating issues.

- AWA 101 Fundamentals of Application Security
- AWA 102 Secure Software Concepts
- COD 101 Fundamentals of Secure Development
- COD 141 FUNDAMENTALS OF SECURE DATABASE DEVELOPMENT
- COD 152 Fundamentals of Secure Cloud Development
- COD 153 Fundamentals of Secure Database Development
- COD 222 PCI DSS v3.2 Best Practices for Developers
- COD 225 Insecure IoT Web Interfaces
- COD 226 Insecure IoT Authentication and Authorization
- COD 227 Insecure IoT Network Services
- COD 228 Insecure IoT Communications
- COD 229 Insecure IoT Mobile Interface
- COD 230 Insecure IoT Firmware
- COD 241 Creating Secure Oracle Database Applications
- DES 101 Fundamentals of Secure Architecture
- DES 202 Cryptographic Suite Services: Encoding, Encrypting and Hashing
- DES 203 Cryptographic Components Randomness, Algorithms, And Key Management
- DES 204 The Role of Cryptography in Application Development
- DES 205 Message Integrity Cryptographic Functions
- DES 212 Architecture Risk Analysis and Remediation
- DES 214 Securing Network Access
- DES 215 Securing Operating System Access
- DES 216 Securing Cloud Instances
- DES 217 Application, Technical and Physical Access Controls
- DES 222 Applying OWASP 2017: Mitigating Injection
- DES 223 Applying OWASP 2017: Mitigating Broken Authentication

- DES 224 Applying OWASP 2017: Mitigating Sensitive Data Exposure
- DES 225 Applying OWASP 2017: Mitigating XML External Entities
- DES 226 Applying OWASP 2017: Mitigating Broken Access Control
- DES 227 Applying OWASP 2017: Mitigating Security Misconfiguration
- DES 228 Applying OWASP 2017: Mitigating Cross Site Scripting
- DES 229 Applying OWASP 2017: Mitigating Insecure Deserialization
- DES 230 Applying OWASP 2017: Mitigating Use of Components with Known Vulnerabilities
- DES 231 Applying OWASP 2017: Mitigating Insufficient Logging and Monitoring
- ENG 191 Introduction to The Microsoft SDL
- ENG 192 Implementing the Agile MS SDL
- ENG 193 Implementing the MS SDL Optimization Model
- ENG 194 Implementing MS SDL Line of Business
- ENG 195 Implementing the MS SDL Threat Modeling Tool
- ENG 205 Fundamentals of Threat Modeling
- ENG 211 How to Create Application Security Design Requirements
- ENG 311 Attack Surface Analysis and Reduction
- ENG 312 How to Perform a Security Code Review
- TST 253 Testing for Classic Buffer Overflow
- TST 256 Testing for Missing Authorization
- TST 257 Testing for Use of Hard-Coded Credentials
- TST 258 Testing for Missing Encryption of Sensitive Data
- TST 259 Testing for Unrestricted Upload of File with Dangerous Type
- TST 260 Testing for Reliance on Untrusted Inputs in A Security Decision
- TST 261 Testing for Execution with Unnecessary Privileges
- TST 262 Testing for Cross-Site Request Forgery
- TST 264 Testing for Download of Code Without Integrity Check
- TST 266 Testing for Inclusion of Functionality from Untrusted Control Sphere
- TST 267 Testing for Incorrect Permission Assignment for Critical Resource
- TST 268 Testing for Use of a Potentially Dangerous Function
- TST 269 Testing for Use of a Broken or Risky Cryptographic Algorithm
- TST 272 Testing for Open Redirect
- TST 273 Testing for Uncontrolled Format String

Embedded Developer (25 Courses, 12 Hours, 12 CPE Credits)

The Embedded Developer Learning Path is designed for those responsible for designing and implementing software of embedded devices and systems.

Embedded Developer learning path provides learners with a thorough grounding in application security concepts across the fundamental courses with special attention to coding within embedded systems and includes secure mobile development.

- AWA 101 Fundamentals of Application Security
- AWA 102 Secure Software Concepts
- COD 110 Fundamentals of Secure Mobile Development
- COD 160 Fundamentals of Secure Embedded Software Development
- COD 201 Secure C Encrypted Network Communications
- COD 202 Secure C Run-Time Protection
- COD 225 Insecure IoT Web Interfaces
- COD 226 Insecure IoT Authentication and Authorization
- COD 227 Insecure IoT Network Services
- COD 228 Insecure IoT Communications
- COD 229 Insecure IoT Mobile Interface
- COD 230 Insecure IoT Firmware
- COD 206 Creating Secure C++ Code
- COD 207 Communication Security in C++
- COD 261 Threats to Scripts
- COD 301 Secure C Buffer Overflow Mitigations
- COD 302 Secure C memory Management
- COD 303 Common C Vulnerabilities
- COD 307 Protecting Data in C++
- DES 202 Cryptographic Suite Services: Encoding, Encrypting and Hashing
- DES 203 Cryptographic Components Randomness, Algorithms, And Key Management
- DES 204 The Role of Cryptography in Application Development
- DES 205 Message Integrity Cryptographic Functions
- DES 260 Fundamentals of IoT Architecture and Design
- DES 201 Fundamentals of Cryptography
- ENG 205 Fundamentals of Threat Modeling

Core Developer (41 Courses, 19 Hours, 19 CPE Credits)

The Core Developer Learning Path is designed for those responsible for the design, development, and management of applications across various environments and operating platforms.

Core Developer learning path provides learners with a solid foundation of application security best practices. To make the life of the software development team easier, courses will dive into main drivers for application security, including concepts of application security risk management, and key application security and coding principles. We will also explore threat modeling, application security design requirements, and learn to identify and mitigate CWE's 25 most dangerous software errors.

- AWA 101 Fundamentals of Application Security
- AWA 102 Secure Software Concepts
- COD 101 Fundamentals of Secure Development
- COD 141 Fundamentals of Secure Database Development
- DES 101 Fundamentals of Secure Architecture

- DES 202 Cryptographic Suite Services: Encoding, Encrypting and Hashing
- DES 203 Cryptographic Components Randomness, Algorithms, And Key Management
- DES 204 The Role of Cryptography in Application Development
- DES 205 Message Integrity Cryptographic Functions
- DES 212 Architecture Risk Analysis and Remediation
- DES 222 Applying OWASP 2017: Mitigating Injection
- DES 223 Applying OWASP 2017: Mitigating Broken Authentication
- DES 224 Applying OWASP 2017: Mitigating Sensitive Data Exposure
- DES 225 Applying OWASP 2017: Mitigating XML External Entities
- DES 226 Applying OWASP 2017: Mitigating Broken Access Control
- DES 227 Applying OWASP 2017: Mitigating Security Misconfiguration
- DES 228 Applying OWASP 2017: Mitigating Cross Site Scripting
- DES 229 Applying OWASP 2017: Mitigating Insecure Deserialization
- DES 230 Applying OWASP 2017: Mitigating Use of Components with Known Vulnerabilities
- DES 231 Applying OWASP 2017: Mitigating Insufficient Logging and Monitoring
- ENG 191 Introduction to The Microsoft SDL
- ENG 192 Implementing the Agile MS SDL
- ENG 193 Implementing the MS SDL Optimization Model
- ENG 194 Implementing MS SDL Line of Business
- ENG 195 Implementing the MS SDL Threat Modeling Tool
- ENG 205 Fundamentals of Threat Modeling
- ENG 211 How to Create Application Security Design Requirements
- ENG 311 Attack Surface Analysis and Reduction
- ENG 312 How to Perform A Security Code Review
- TST 255 Testing for Missing Authentication for Critical Function
- TST 257 Testing for Use of Hard-Coded Credentials
- TST 259 Testing for Unrestricted Upload of File with Dangerous Type
- TST 260 Testing for Reliance on Untrusted Inputs in A Security Decision
- TST 261 Testing for Execution with Unnecessary Privileges
- TST 264 Testing for Download of Code Without Integrity Check
- TST 266 Testing for Inclusion of Functionality from Untrusted Control Sphere
- TST 267 Testing for Incorrect Permission Assignment for Critical Resource
- TST 268 Testing for Use of a Potentially Dangerous Function
- TST 271 Testing for Improper Restriction of Excessive Authentication Attempts
- TST 272 Testing for Open Redirect
- TST 273 Testing for Uncontrolled Format String

IT Architect (18 Courses, 11 Hours, 11 CPE Credits)

The IT Architect Learning Path is designed for those responsible for designing and maintaining computer networks.

IT Architect learning path provides learners with best practices for the design of secure software. Learners will learn to apply these best practices to the creation of integrated architecture across business and technology and protect data and resources from disclosure, modification and deletion.

- AWA 101 Fundamentals of Application Security
- AWA 102 Secure Software Concepts
- COD 253 Creating Secure Cloud Code - AWS Foundations
- DES 101 Fundamentals of Secure Architecture
- DES 212 Architecture Risk Analysis and Remediation
- DES 214 Securing Network Access
- DES 215 Securing Operating System Access
- DES 216 Securing Cloud Instances
- DES 217 Application, Technical and Physical Access Controls
- DES 260 Fundamentals of IoT Architecture and Design
- ENG 191 Introduction to The Microsoft SDL
- ENG 192 Implementing the Agile MS SDL
- ENG 193 Implementing the MS SDL Optimization Model
- ENG 194 Implementing MS SDL Line of Business
- ENG 195 Implementing the MS SDL Threat Modeling Tool
- ENG 211 How to Create Application Security Design Requirements
- ENG 311 Attack Surface Analysis and Reduction
- ENG 392 Attack Surface Analysis and Reduction for Embedded Systems

Embedded Architect (11 Courses, 8 Hours, 8 CPE Credits)

The Embedded Architect Learning Path is designed for those responsible for designing and implementing software of embedded devices and systems.

Embedded Architect learning path provides learners with best practices for the design of secure software for embedded systems. Learners will explore the unique resource requirements of embedded environments and best practices for the design and architecting of secure software for embedded systems.

- DES 101 Fundamentals of Secure Architecture
- DES 212 Architecture Risk Analysis and Remediation
- DES 260 Fundamentals of IoT Architecture and Design
- DES 311 Creating Secure Application Architecture
- ENG 191 Introduction to The Microsoft SDL
- ENG 192 Implementing the Agile MS SDL
- ENG 193 Implementing the MS SDL Optimization Model
- ENG 194 Implementing MS SDL Line of Business
- ENG 195 Implementing the MS SDL Threat Modeling Tool
- ENG 311 Attack Surface Analysis and Reduction
- ENG 312 How to Perform a Security Code Review

Microsoft SDL (21 Courses, 15 Hours, 15 CPE Credits)

The Microsoft SDL Learning Path is designed for those responsible for implementing the industry-leading software security assurance process.

Microsoft SDL learning path provides learners with a baseline understanding of implementing the holistic and practical approach of the Microsoft Security Development Lifecycle (SDL). Upon completion of this learning path learners will be able to apply security and privacy early and throughout all phases of the development process.

- AWA 101 Fundamentals of Application Security
- AWA 102 Secure Software Concepts
- COD 101 Fundamentals of Secure Development
- COD 216 Leveraging .Net Framework Code Access Security (Cas)
- COD 217 Mitigating .NET Security Threats
- COD 242 Creating Secure MS SQL Server Applications
- COD 254 Creating Secure Azure Applications
- DES 101 Fundamentals of Secure Architecture
- DES 202 Cryptographic Suite Services: Encoding, Encrypting and Hashing
- DES 203 Cryptographic Components Randomness, Algorithms, And Key Management
- DES 204 The Role of Cryptography in Application Development
- DES 205 Message Integrity Cryptographic Functions
- DES 212 Architecture Risk Analysis and Remediation
- ENG 191 Introduction to The Microsoft SDL
- ENG 192 Implementing the Agile MS SDL
- ENG 193 Implementing the MS SDL Optimization Model
- ENG 194 Implementing MS SDL Line of Business
- ENG 195 Implementing the MS SDL Threat Modeling Tool
- ENG 211 How to Create Application Security Design Requirements
- ENG 311 Attack Surface Analysis and Reduction
- ENG 312 How to Create Application Security Design Requirements

GDPR (39 Courses, 18 Hours, 18 CPE Credits)

The GDPR Learning Path is designed for those responsible for ensuring software development lifecycles adhere to GDPR.

GDPR learning path is intended to provide a baseline level of preparedness for GDPR compliance. This path is designed to provide a natural progression for becoming proficient in GDPR compliance.

- AWA 101 Fundamentals of Application Security
- AWA 102 Secure Software Concepts

- DES 101 Fundamentals of Secure Architecture
- DES 202 Cryptographic Suite Services: Encoding, Encrypting and Hashing
- DES 203 Cryptographic Components Randomness, Algorithms, And Key Management
- DES 204 The Role of Cryptography in Application Development
- DES 205 Message Integrity Cryptographic Functions
- DES 212 Architecture Risk Analysis and Remediation
- DES 222 Applying OWASP 2017: Mitigating Injection
- DES 223 Applying OWASP 2017: Mitigating Broken Authentication
- DES 224 Applying OWASP 2017: Mitigating Sensitive Data Exposure
- DES 225 Applying OWASP 2017: Mitigating XML External Entities
- DES 226 Applying OWASP 2017: Mitigating Broken Access Control
- DES 227 Applying OWASP 2017: Mitigating Security Misconfiguration
- DES 228 Applying OWASP 2017: Mitigating Cross Site Scripting
- DES 229 Applying OWASP 2017: Mitigating Insecure Deserialization
- DES 230 Applying OWASP 2017: Mitigating Use of Components with Known Vulnerabilities
- DES 231 Applying OWASP 2017: Mitigating Insufficient Logging and Monitoring
- DES 311 Creating Secure Application Architecture
- ENG 191 Introduction to The Microsoft SDL
- END 192 Implementing the Agile MS SDL
- ENG 193 Implementing the MS SDL Optimization Model
- ENG 194 Implementing MS SDL Line of Business
- ENG 195 Implementing the MS SDL Threat Modeling Tool
- ENG 205 Fundamentals of Threat Modeling
- ENG 311 Attack Surface Analysis
- TST 101 Fundamentals of Security Testing
- TST 255 Testing for Missing Authentication for Critical Function
- TST 257 Testing for Use of Hard-Coded Credentials
- TST 259 Testing for Unrestricted Upload of File with Dangerous Type
- TST 260 Testing for Reliance on Untrusted Inputs in A Security Decision
- TST 261 Testing for Execution with Unnecessary Privileges
- TST 264 Testing for Download of Code Without Integrity Check
- TST 266 Testing for Inclusion of Functionality from Untrusted Control Sphere
- TST 267 Testing for Incorrect Permission Assignment for Critical Resource
- TST 268 Testing for Use of a Potentially Dangerous Function
- TST 271 Testing for Improper Restriction of Excessive Authentication Attempts
- TST 272 Testing for Open Redirect
- TST 273 Testing for Uncontrolled Format String

OWASP (30 Courses, 10 Hours, 10 CPE Credits)

The OWASP Learning Path is designed for those who need to thoroughly understand the consequences of the most common and most important web application security weaknesses.

OWASP learning path introduces learners to the need for secure software development, as well as the models, standards, and guidelines to understand security issues and improve the security posture of applications. The learning path also teaches students to identify and mitigate risks associated with the 2017 OWASP Top 10 and how to test for these critical vulnerabilities.

- AWA 101 Fundamentals of Application Security
- AWA 102 Secure Software Concepts
- COD 101 Fundamentals of Secure Development
- COD 259 Node.Js Threats and Vulnerabilities
- DES 222 Applying OWASP 2017: Mitigating Injection
- DES 223 Applying OWASP 2017: Mitigating Broken Authentication
- DES 224 Applying OWASP 2017: Mitigating Sensitive Data Exposure
- DES 225 Applying OWASP 2017: Mitigating XML External Entities
- DES 226 Applying OWASP 2017: Mitigating Broken Access Control
- DES 227 Applying OWASP 2017: Mitigating Security Misconfiguration
- DES 228 Applying OWASP 2017: Mitigating Cross Site Scripting
- DES 229 Applying OWASP 2017: Mitigating Insecure Deserialization
- DES 230 Applying OWASP 2017: Mitigating Use of Components with Known Vulnerabilities
- DES 231 Applying OWASP 2017: Mitigating Insufficient Logging and Monitoring
- ENG 191 Introduction to The Microsoft SDL
- ENG 192 Implementing the Agile MS SDL
- ENG 193 Implementing the MS SDL Optimization Model
- ENG 194 Implementing MS SDL Line of Business
- ENG 195 Implementing the MS SDL Threat Modeling Tool
- ENG 205 Fundamentals of Threat Modeling
- TST 222 Testing for OWASP 2017: Injection
- TST 223 Testing for OWASP 2017: Broken Authentication
- TST 224 Testing for OWASP 2017: Sensitive Data Exposure
- TST 225 Testing for OWASP 2017: XML External Entities
- TST 226 Testing for OWASP 2017: Broken Access Control
- TST 227 Testing for OWASP 2017: Security Misconfiguration
- TST 228 T Testing for OWASP 2017: Cross-Site Scripting
- TST 229 Testing for OWASP 2017: Insecure Deserialization
- TST 230 Testing for OWASP 2017: Use of Components with Known Vulnerabilities
- TST 231 Testing for OWASP 2017: Insufficient Logging and Monitoring

SQL Developer (21 Courses, 17 Hours, 17 CPE Credits)

The SQL Developer Learning Path is designed for those developing SQL databases and writing applications to interface with SQL Databases, as well as writing and testing code.

SQL Developer learning path introduces learners to secure best practices for designing tables, storing procedures, views and functions. The learning paths also teaches students to build secure applications and encrypt data using features in SQL server.

- Awa 101 Fundamentals of Application Security
- Awa 102 Secure Software Concepts
- COD 101 Fundamentals of Secure Development
- COD 141 Fundamentals of Secure Database Development
- COD 241 Creating Secure Oracle Database Applications
- DES 101 Fundamentals of Secure Architecture
- DES 202 Cryptographic Suite Services: Encoding, Encrypting and Hashing
- DES 203 Cryptographic Components Randomness, Algorithms, And Key Management
- DES 204 The Role of Cryptography in Application Development
- DES 205 Message Integrity Cryptographic Functions
- DES 212 Architecture Risk Analysis and Remediation
- DES 311 Creating Secure Application Architecture
- ENG 191 Introduction to The Microsoft SDL
- ENG 192 Implementing the Agile MS SDL
- ENG 193 Implementing the MS SDL Optimization Model
- ENG 194 Implementing MS SDL Line of Business
- ENG 195 Implementing the MS SDL Threat Modeling Tool
- ENG 205 Fundamentals of Threat Modeling
- ENG 211 How to Create Application Security Design Requirements
- ENG 311 Attack Surface Analysis and Reduction
- ENG 312 How to Perform a Security Code Review