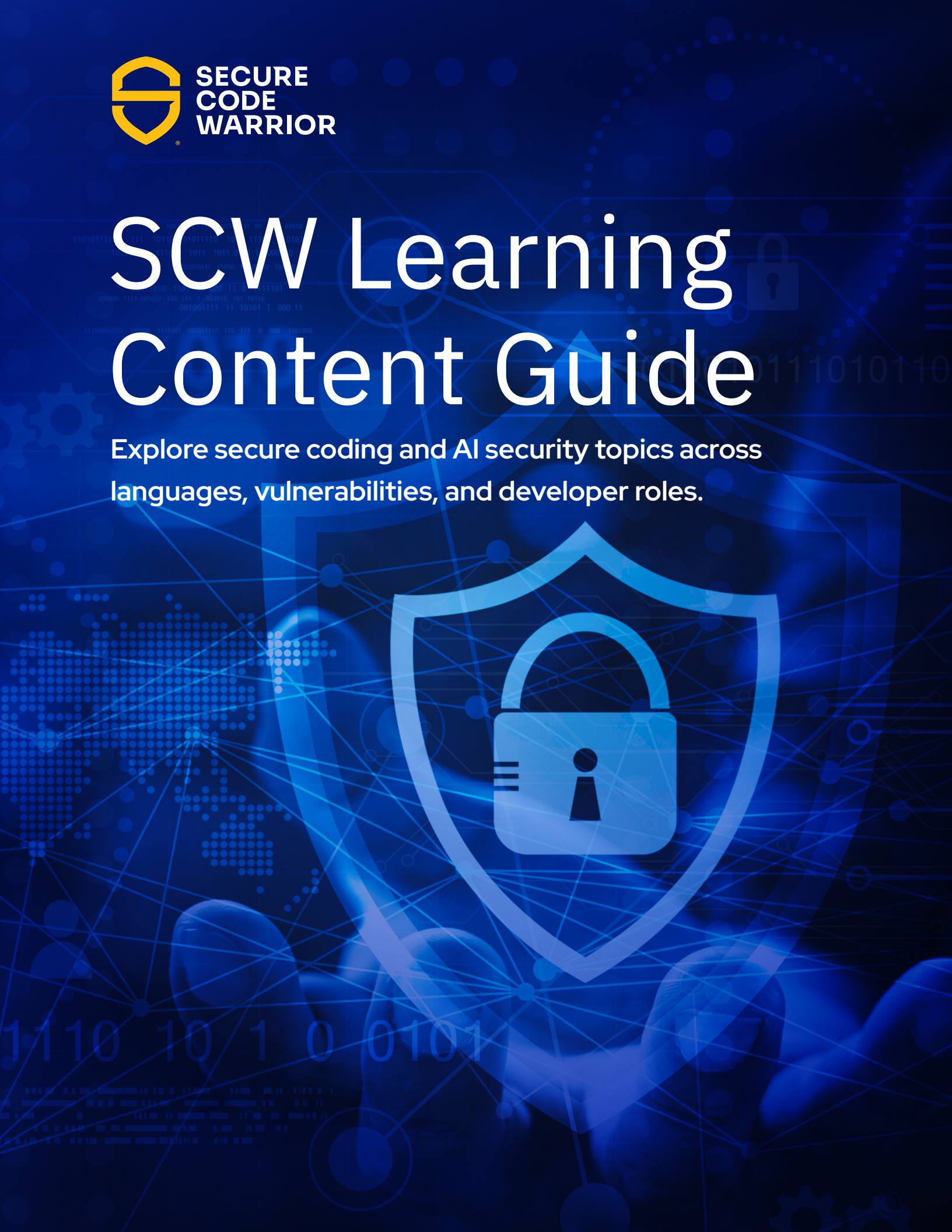
SECURE CODE WARRIOR

SCW Learning Content Guide

Explore secure coding and AI security topics across languages, vulnerabilities, and developer roles.



Introduction

SCW Learning offers the most comprehensive secure coding training library in the industry — designed for modern engineering teams building with AI. This guide highlights our featured coding languages and frameworks, vulnerability categories, and conceptual and role-based topics — so you can quickly understand what's available and how to map coverage to your AI Software Governance goals.

Table of Contents

Topics & Content as of March 2026. See the SCW Knowledge Base for current listings:
Quests Security Awareness and Design Concepts Catalogue and Course Templates Catalogue.

Coding Languages & Frameworks

AI/LLM

Python:LangChain
Python:MCP
Terraform:AWS (Bedrock)

Front-end

Typescript:Basic
JavaScript:Basic
Typescript:React
JavaScript:React
JavaScript:Angular.io (2+)
JavaScript:Vue.js

Mobile

Swift:iOS SDK
JavaScript:React Native
Kotlin:Android SDK
Pseudocode:Mobile
Java:Android SDK
Objective-C:iOS SDK
Dart:Flutter

Other

C:Basic
C++:Basic
Bash:Basic
PowerShell:Basic
COBOL:Mainframe
PL/SQL:Basic
T-SQL:Basic
C:Embedded
C++:Embedded
SAP:ABAP
RPG:Basic
R:Basic

API

Python:API
C# (.NET):Web API
Java:Enterprise Edition API
Java:Spring API
JavaScript:Node.js API
GO:API
Kotlin:Spring API

Infrastructure as Code (IaC)

Docker:Basic
Kubernetes:Basic
Terraform:AWS
CloudFormation:Basic
Ansible:Basic
Terraform:Azure
Terraform:GCP

Web

Python:Basic
C# (.NET):Basic
C# (.NET):Core
Java:Enterprise Edition (Basic)
Java:Spring
Typescript:Node.js (express)
JavaScript:Node.js (Express)
GO:Basic
Pseudocode:Web
PHP:Basic
Rust:Basic
Ruby:Rails
Salesforce:Apex
Scala:Play
Perl:Dancer2
Python:Django
Python:Flask
C# (.NET):MVC
C# (.NET):Web Forms
Java:Enterprise Edition (JSP)
Java:Enterprise Edition (JSF)
Java:Servlets
Java:Struts
PHP:Symfony

Learn more: https://www.securecodewarrior.com/product/platform-content

Topics & Content by Category

Vulnerability Topics

Access Control
Insecure Direct Object Reference
Missing Function Level Access Control
Missing Object Level Access Control
Using Input from Untrusted Sources

Authentication
Improper Authentication
Insecure Password Change Function
Insecure Password Reset Function
Insufficient Anti-Automation
Insufficiently Protected Credentials
Use of Single-factor Authentication
Username Enumeration
Weak Password Policy

Business Logic
Insufficient Validation
Logical Error

Cross-Site Request Forgery
Cross-Site Request Forgery

Cross-Site Scripting (XSS)
DOM-Based Cross-Site Scripting
Reflected Cross-Site Scripting
Stored Cross-Site Scripting

Denial of Service
Failure to Release Resource
Uncaught Error Handling

File Upload Vulnerability
Unrestricted File Upload

Improper Assets Management
Improper Assets Management

Information Exposure
Debug Information
Error Details
Sensitive Data Exposure

Injection Flaws
Code Injection
CSS Injection
Deserialization of Untrusted Data
Email Injection
HTTP Injection
LDAP Injection
Log Forging
NoSQL Injection
OS Command Injection
Path Traversal
Resource Injection
SQL Injection
XML Injection
XPath Injection
XQuery Injection

Insecure Cryptography
Exposed Key
Insecure Randomness
Weak Algorithm Use

Insufficient Logging and Monitoring
Insufficient Logging and Monitoring

Insufficient Transport Layer Protection
Unprotected Transport of Credentials
Unprotected Transport of Sensitive Information
Weak Algorithm or Protocol Use

Lack of Resources & Rate Limiting
Lack of Resources & Rate Limiting

LLM
Direct Prompt Injection
Excessive Agency
Improper Output Handling
Indirect Prompt Injection
Sensitive Information Disclosure
Supply Chain
System Prompt Leakage
Vector and Embedding Weaknesses

Mass Assignment
Mass Assignment

Memory Corruption
Buffer Overflow
Double Free
Format String Vulnerabilities
Heap Overflow
Illegal Pointer Value
Integer Overflow
Null Dereference

Vulnerability Topics Continued

Race Conditions
Stack Overflow
Type Confusion
Uninitialized Variable
Use After Free

Security Misconfiguration
Clickjacking
Debug Features Enabled
Disabled Security Features
Improper or Missing HTTP Headers
Improper Permissions
Information Exposure

Sensitive Data Storage
Plaintext Storage of Passwords
Plaintext Storage of Sensitive Information

Server-Side Request Forgery
Server-Side Request Forgery

Session Handling
Exposed Session Tokens
Insufficient Session Expiration
Weak Session Token Generation

Side Channel Vulnerability
Data sent to 3rd parties (analytics/error handling)

Unvalidated Redirects and Forwards
Unvalidated Redirects and Forwards

Vulnerable Components
Using Components from Untrusted Source
Using Known Vulnerable Components

XML External Entities (XXE)
XML External Entities (XXE)

Mobile Vulnerability Topics

Broken Cryptography
Improper Use of Cryptography Algorithm
Insecure Generation of Encryption Keys
Insecure Storage of Encryption Keys
Reuse of Initialization Vector
Use of Encoding
Use of Hardcoded Keys
Use of Insecure/Deprecated Algorithms
Use of Short Encryption Keys
Use of Insecure/Deprecated Algorithms
Use of Short Encryption Keys

Client Side Injection
JavaScript Injection
Untrusted third party sites

Code Tampering
Backups Enabled
Tampering Detection

Extraneous Functionality
Autofill Password
Debugging Features Enabled

Improper Platform Usage
Incorrect Activity Configuration
Insecure File/Directory Permissions
Misuse of Broadcast Receivers
Misuse of Intents
Misuse of URL Schemes
Tapjacking
Webview Settings

Improper Session Handling
Client Side Session Token Generation

Insecure Authentication
Client Side Authentication for Authenticating to Server
Hardcoded API Keys
Misuse of Fingerprint
Storing Credentials With 'Remember Me' Functionality
Use of Spoofable Parameters for Authentication
Use of Weak Passwords
Weak Lockout Mechanism

Insecure Authorization
Insecure Direct Object Reference
Using inputs from untrusted sources

Insecure Data Storage
Plaintext Storage of Credentials
Storage in Plist or XML Files
Storage in SQLite Databases
Storage on SDCard/External Storage

Insufficient Transport Layer Protection
Communication Over Cleartext Protocol
Improper Certificate Pinning Configuration
Trusting Self-Signed or Untrusted Certificates
Weak Certificate Validation
Weak Cipher Suites

Vulnerability Topics Continued

Lack of Binary Protections
Lack of Adequate Security Controls
No Code Obfuscation
No Protection from Debuggers
No Protection from Piracy

Reverse Engineering
Code Information Leakage
Emulation Detection

Unintended Data Leakage
Analytics Data Sent to 3rd Parties
Application Backgrounding Screenshots
Copy/Paste Buffer Caching (Pasteboard)
Keyboard Caching
Logging Sensitive Information
Request/Response Caching

Conceptual Content

AI/LLM Security
AI Agents and their Protocols (MCP, A2A and ACP)
Coding With AI
Introduction to AI Risk & Security
LLM Security Design Patterns
OWASP Top 10 for Large Language Model (LLM) Applications
Threat Modeling with AI
Vibe Coding: Risk Management Framework

Authentication and Authorisation Protocols
Essential API Security: Authentication & Authorization
OAuth 2.0 Advanced Topics
OAuth 2.0 Security
SAML Security
Securely Accessing APIs Using OAuth 2.0
Securely Granting Access to an API Using OAuth 2.0

Cloud Security
Introduction to Cloud Security
Kubernetes Security
Secure Implementation of Docker
Secure Implementation of Kubernetes
Securing Amazon Web Services (AWS)
Securing Google Cloud Platform (GCP)
Securing Infrastructure as Code
Securing Microsoft Azure

Cryptography
Introduction to Cryptography for Developers and Architects

Cyber Resilience Act (CRA)
Architecture Risk Analysis
Foundations of Software Security
Introduction to the Cyber Resilience Act (CRA)
Open-Source Software (OSS)
Threat Modelling
Vulnerability Handling

Cybermon 2025: Beat the Boss
Bypassaur: Direct Prompt Injection
Keykracken: Indirect Prompt Injection
Promptgeist: Vector and Embedding Weaknesses
Proxysurfa: Excessive Agency

Data Security
Database Security
Data Security: Secure Password Storage
Data Security: Security for Data Scientists & Analysts
DevSecOps Security
Container Security

DevSecOps Security
Introduction to Typescript Pulumi Security for AWS
Non-Human Identities (NHI)
OWASP Top 10 CI/CD (GitHub Actions)
OWASP Top 10 CI/CD (Jenkins Declarative)
OWASP Top 10 CI/CD (Jenkins Scripted)

Embedded Security
Advanced Embedded Security
Introduction to Embedded Security

Industry-Specific Security
Introduction to Automotive Security
Secure Development for Healthcare

Internet of Things
Introduction to Securing the Internet of Things
Secure Communication for the Internet of Things

Mainframe Security
Defensive Programming for COBOL
Mainframe Security: Foundations of COBOL Security

Mobile Security
Foundations of Mobile Security
Fundamentals of iOS (Objective-C)
Fundamentals of iOS (Swift)
Java Android Security
Kotlin Advanced Android Security

Kotlin Android Security
Secure Programming for iOS (Objective-C)
Secure Programming for iOS (Swift)

NIS2
Architecture Risk Analysis
Foundations of Software Security
Open-Source Software (OSS)
Threat Modelling
Vulnerability Handling

Open-Source and Supply Chain
Open-Source Policies and Risks
Open-Source Software (OSS)
Vetting Your Digital Supply Chain

Payment Card Industry (PCI)
PCI DSS v4.0 Concepts and Compliance

Privacy
GDPR for Developers and Architects
GDPR for Development and Project Managers
Introduction to CCPA
Introduction to GDPR

Secure Software Design
Architecture Risk Analysis
Risk-Based Security Testing Strategy
Security Requirements
Threat Modeling with AI
Threat Modelling

Security Foundations
Application Security Concepts
Attack and Defence
Foundations of Software Security
Hardening Your APIs
Low-Code and No-Code (LCNC)
Non-Human Identities (NHI)
Overview of Application Security Testing

Systems Programming Security
Modern C Security
Modern C++ Security

Web Security
Defensive Programming for HTML5 Security
Introduction to HTML5 Security
Secure Programming for Go
Web Application Security 101

Interactive Topics & Content

SCW Missions

Deeply immersive interactive learning delivering coding simulations to developers in a risk-free environment.

Access Control
Insecure Direct Object Reference
Missing Function Level Access Control
Missing Object Level Access Control

Authentication
Improper Authentication
Insecure Password Reset Function

Business Logic
Insufficient Validation

Cross-Site Request Forgery
Cross Site Request Forgery

Cross-Site Scripting (XSS)
DOM-Based Cross-Site Scripting
Reflected Cross-Site Scripting
Stored Cross-Site Scripting

File Upload Vulnerability
Unrestricted File Upload

Information Exposure
Error Details
Sensitive Data Exposure

Injection Flaws
CSS Injection
Deserialization of Untrusted Data
NoSQL Injection
OS Command Injection
Path Traversal
SQL Injection

Insecure Cryptography
Exposed Key
Insecure Randomness
Weak Algorithm Use

Insufficient Logging and Monitoring
Insufficient Logging and Monitoring

LLM
Direct Prompt Injection
Excessive Agency
Improper Output Handling
Indirect Prompt Injection
LLM awareness
Sensitive Information Disclosure
Vector and Embedding Weaknesses

Mass Assignment
Mass Assignment

Memory Corruption
Buffer Overflow
Null Deference

Security Misconfiguration
Clickjacking
Debug Features Enabled
Disabled Security Features
Information Exposure

Sensitive Data Storage
Plaintext Storage of Passwords
Plantext Storage of Sensitive Information

Server-Side Request Forgery
Server-Side Request Forgery

Session Handling
Exposed Session Tokens
Weak Session Token Generation

Unvalidated Redirects and Forwards
Unvalidated Redirects and Forwards

Vulnerable Components
Log4j
Using Components from Untrusted Source
Using Known Vulnerable Components

XML External Entities (XXE)
XML External Entities (XXE)

SCW Coding Labs

Hands-on interactive learning with intuitive feedback delivered through a powerful in-browser IDE.

Access Control
Missing Function Level Access Control
Missing Object Level Access Control

Authentication
Improper Authentication
Insecure Password Reset Function
Insufficient Anti-Automation
Insufficiently Protected Credentials
Use of Single-factor Authentication

Business Logic
Insufficient Validation
Logical Error

Cross-Site Scripting (XSS)
DOM-Based Cross-Site Scripting
Stored Cross-Site Scripting

Denial of Service
Uncaught Error Handling

File Upload Vulnerability
Unrestricted File Upload

Information Exposure
Error Details
Sensitive Data Exposure

Injection Flaws
Deserialization of Untrusted Data
LDAP Injection
OS Command Injection
Path Traversal
SQL Injection

Insecure Cryptography
Insecure Randomness
Weak Algorithm Use

Insufficient Logging and Monitoring
Insufficient Logging and Monitoring

Insufficient Transport Layer Protection
Unprotected Transport of Sensitive Information
Weak Algorithm or Protocol Use

Lack of Resources & Rate Limiting
Lack of Resources & Rate Limiting

LLM
Direct Prompt Injection
Excessive Agency
Sensitive Information Disclosure

Mass Assignment
Mass Assignment

Memory Corruption
Buffer Overflow
Double Free
Format String Vulnerabilities
Integer Overflow
Null Dereference
Race Conditions
Type Confusion
Uninitialized Variable
Use After Free

Security Misconfiguration
Disabled Security Features
Improper or Missing HTTP Headers
Improper Permissions
Information Exposure

Sensitive Data Storage
Plaintext Storage of Passwords
Plaintext Storage of Sensitive Information

Server-Side Request Forgery
Server-Side Request Forgery

Unvalidated Redirects and Forwards
Unvalidated Redirects and Forwards

Vulnerable Components
Using Components from Untrusted Source
Using Known Vulnerable Components

XML External Entities (XXE)
XML External Entities (XXE)

Topics & Content by Role

AI/LLM Topics & Content

Audience: Engineers, Architects, Technical Managers, Product Managers

Vulnerability Topics

Bedrock
Direct Prompt Injection
Excessive Agency
Insufficient Logging and Monitoring
Sensitive Information Disclosure

LLM
Direct Prompt Injection
Excessive Agency
Improper Output Handling
Indirect Prompt Injection
Injection Flaws: Code Injection
Injection Flaws: Deserialization of Untrusted Data
Injection Flaws: SQL Injection
Sensitive Information Disclosure
Supply Chain
System Prompt Leakage
Vector and Embedding Weaknesses

MCP
Access Control: Missing Function Level Access Control
Authentication: Improper Authentication
Authentication: Insufficiently Protected Credentials
Direct Prompt Injection
Indirect Prompt Injection
Information Exposure: Sensitive Data Exposure
Insufficient Logging and Monitoring: Insufficient Logging and Monitoring
Insufficient Transport Layer Protection: Unprotected Transport of Sensitive Information
Server-Side Request Forgery: Server-Side Request Forgery
Vulnerable Components: Using Known Vulnerable Components

Conceptual Content

AI Agents and their Protocols (MCP & A2A)
Coding With AI
Introduction to AI Risk & Security
LLM Security Design Patterns
OWASP Top 10 for Large Language Model (LLM) Applications
Threat Modeling with AI
Vibe Coding: Risk Management Framework

Architecture Topics & Content

Audience: Architects, Engineering Managers, Platform Engineers, Lead Engineers

Vulnerability Topics

Access Control
Insecure Direct Object Reference
Missing Function Level Access Control
Missing Object Level Access Control
Using input from untrusted sources

Authentication
Forceful Browsing
Improper Authentication
Insecure Password Change Function
Insecure Password Reset Function
Insufficient Anti-Automation
Insufficiently Protected Credentials
Password Enumeration
Use of Single-factor Authentication
Username Enumeration
Weak Password Policy

Broken Cryptography
Improper Use of Cryptography Algorithm
Insecure Generation of Encryption Keys
Insecure Storage of Encryption Keys
Reuse of Initialization Vector
Use of Encoding
Use of Hardcoded Keys
Use of Insecure/Deprecated Algorithms
Use of Short Encryption Keys

Business Logic
Insufficient Validation
Logical Error
Client Side

Client Side Injection
DOM-Based Cross-Site Scripting
JavaScript Injection
SQL Injection
Untrusted third party sites
XML Injection

Cross-Site Request Forgery
Cross-Site Request Forgery

Cross-Site Scripting (XSS)
DOM-Based Cross-Site Scripting
Mixed server and Client Side Rendering
Reflected Cross-Site Scripting
Stored Cross-Site Scripting

Injection Flaws
Code Injection
CSS Injection
Deserialization of Untrusted Data
Email Injection
HTTP Injection
LDAP Injection
Local File Inclusion
Log Forging
NoSQL Injection
OS Command Injection
Path Traversal
Remote File Inclusion
Resource Injection

Insecure Authentication
Client Side Authentication for Authenticating to Server
Hardcoded API Keys
Misuse of Fingerprint
Password Enumeration
Storing Credentials With 'Remember Me' Functionality
Use of Spoofable Parameters for Authentication
Use of Weak Passwords
Username Enumeration
Weak Lockout Mechanism

Insecure Authorization
Broken Access Control
Insecure Direct Object Reference
Using inputs from untrusted sources

Architecture Topics & Content Continued

Insecure Cryptography
Decryption Oracle
Encryption Oracle
Exposed Key
Improper Post-Quantum Cryptography (PQC)
Insecure Randomness
Insufficiently Protected Credentials
Padding Oracle
Reuse of Initialization Vector
Weak Algorithm Use

Secure Misconfiguration
Clickjacking
Debug Features Enabled
Disabled Security Features
Improper or Missing HTTP Headers
Improper Permissions
Information Exposure
Local File Inclusion
Log Forging
NoSQL Injection
OS Command Injection
Path Traversal
Remote File Inclusion
Resource Injection

Conceptual Content

Architecture Risk Analysis
Attack and Defence
Database Security
Essential API Security: Authentication & Authorization
GDPR for Developers and Architects
Introduction to AI Risk & Security
Introduction to CCPA
Introduction to Cryptography for Developers & Architects
Introduction to the Cyber Resilience Act (CRA)
LLM Security Design Patterns
Non-Human Identities (NHI)
OAuth 2.0 Advanced Topics
OAuth 2.0 Security
Open-Source Policies and Risks
Open-Source Software (OSS)
PCI DSS v4.0 Concepts and Compliance
Risk-Based Security Testing Strategy
SAML Security
Secure Development for Healthcare
Secure Password Storage
Securely Accessing APIs Using OAuth 2.0
Securely Granting Access to an API Using OAuth 2.0
Security for Data Scientists & Analysts
Security Requirements
Threat Modeling with AI
Threat Modelling
Vetting Your Digital Supply Chain

Backend & API Topics & Content

Audience: Backend Engineers, API Engineers

Vulnerability Topics

Access Control
Insecure Direct Object Reference
Missing Function Level Access Control
Missing Object Level Access Control
Using input from untrusted sources

Authentication
Forceful Browsing
Improper Authentication
Insecure Password Change Function
Insecure Password Reset Function
Insufficient Anti-Automation
Insufficiently Protected Credentials
Use of Single-factor Authentication
Username Enumeration
Weak Password Policy

Business Logic
Insufficient Validation
Logical Error

Cross-Site Request Forgery
Cross-Site Request Forgery

Cross-Site Scripting (XSS)
DOM-Based Cross-Site Scripting
Mixed server and client side rendering
Reflected Cross-Site Scripting
Stored Cross-Site Scripting

Denial of Service
Regular Expression DoS
Uncaught Error Handling

File Upload Vulnerability
Unrestricted File Upload

Improper Assets Management
Improper Assets Management

Information Exposure
Debug Information
Error Details
Sensitive Data Exposure

Injection Flaws
Code Injection
CSS Injection
Deserialization of Untrusted Data
Email Injection
HTTP Injection
LDAP Injection
Log Forging
NoSQL Injection
OS Command Injection
Path Traversal
Resource Injection
SQL Injection
XML Injection
XPath Injection
XQuery Injection

Insecure Cryptography
Exposed Key
Improper Post-Quantum Cryptography (PQC)
Insecure Randomness
Insufficiently Protected Credentials
Padding Oracle
Weak Algorithm Use

Insufficient Logging and Monitoring
Insufficient Logging and Monitoring

Insufficient Transport Layer Protection
Unprotected Transport of Credentials
Unprotected Transport of Sensitive Information
Weak Algorithm or Protocol Use

Lack of Resources & Rate Limiting
Lack of Resources & Rate Limiting

Mass Assignment
Mass Assignment

Memory Corruption
Integer Overflow
Null Dereference
Race Conditions

Backend & API Topics & Content Continued

Security Misconfiguration
Clickjacking
Debug Features Enabled
Disabled Security Features
Improper or Missing HTTP Headers
Improper Permissions
Information Exposure

Sensitive Data Storage
Plaintext Storage of Passwords
Plaintext Storage of Sensitive Information

Server-Side Request Forgery
Server-Side Request Forgery

Session Handling
Exposed Session Tokens
Insufficient Session Expiration
Weak Session Token Generation

Side Channel Vulnerability
Timing Attack
URL caching

Unvalidated Redirects and Forwards
Unvalidated Redirects and Forwards

Vulnerable Components
Log4j
Using Components from Untrusted Source
Using Known Vulnerable Components

XML External Entities (XXE)
XML External Entities (XXE)

Conceptual Content

Application Security Concepts
Essential API Security: Authentication & Authorization
Foundations of Software Security
GDPR for Developers and Architects
Hardening Your APIs
Introduction to Cryptography for Developers & Architects
OAuth 2.0 Advanced Topics
OAuth 2.0 Security
Open-Source Software (OSS)
PCI DSS v4.0 Concepts and Compliance
SAML Security
Secure Password Storage
Secure Programming for Go
Securely Accessing APIs Using OAuth 2.0
Securely Granting Access to an API Using OAuth 2.0
Vulnerability Handling
Web Application Security 101

Data Topics & Content

Audience: Data Scientists, Data Analysts, Data Engineers

Vulnerability Topics

Access Control
Missing Function Level Access Control
Missing Object Level Access Control

Authentication
Improper Authentication
Insufficiently Protected Credentials

Business Logic
Insufficient Validation

Denial of Service
Failure to Release Resource
Regular Expression DoS
Uncaught Error Handling

File Upload Vulnerability
Unrestricted File Upload

Information Exposure
Debug Information
Error Details
Sensitive Data Exposure

Injection Flaws
Code Injection
Email Injection
Log Forging
OS Command Injection
Path Traversal
Resource Injection
SQL Injection

Insecure Authorization
Using inputs from untrusted sources

Insecure Cryptography
Decryption Oracle
Encryption Oracle
Exposed Key
Improper Post-Quantum Cryptography (PQC)
Insecure Randomness
Insufficiently Protected Credentials
Padding Oracle
Reuse of Initialization Vector
Weak Algorithm Use

Insufficient Logging and Monitoring
Insufficient Logging and Monitoring

Insufficient Transport Layer Protection
Unprotected Transport of Sensitive Information

LLM
Improper Output Handling
Supply Chain
Vector and Embedding Weaknesses

Security Misconfiguration
Debug Features Enabled
Disabled Security Features
Improper Permissions

Sensitive Data Storage
Plaintext Storage of Passwords
Plaintext Storage of Sensitive Information

XML External Entities (XXE)
XML External Entities (XXE)

Conceptual Content

AI Agents and their Protocols (MCP & A2A)
Architecture Risk Analysis
Database Security
GDPR for Development and Project Managers
Introduction to AI Risk & Security
Introduction to CCPA
Introduction to Cryptography for Developers & Architects
Introduction to GDPR
Introduction to the Cyber Resilience Act (CRA)
LLM Security Design Patterns
Secure Development for Healthcare
Secure Password Storage
Security for Data Scientists & Analysts

DevOps & Cloud Topics & Content

Audience: DevOps Engineers, System Administrators, Platform Engineers, Kubernetes Engineers, Docker Engineers, CI/CD Engineers, Cloud Engineers, AWS Engineers, Azure Engineers

Vulnerability Topics

Access Control
Missing Function Level Access Control
Missing Object Level Access Control
Using input from untrusted sources

Authentication
Improper Authentication
Insufficiently Protected Credentials

Business Logic
Insufficient Validation
Logical Error

Denial of Service
Uncaught Error Handling

Information Exposure
Debug Information
Error Details
Sensitive Data Exposure

Injection Flaws
Code Injection
Email Injection
NoSQL Injection
OS Command Injection
Path Traversal
SQL Injection

Insecure Cryptography
Exposed Key
Improper Post-Quantum Cryptography (PQC)
Insecure Randomness
Weak Algorithm Use

Insufficient Logging and Monitoring
Insufficient Logging and Monitoring

Insufficient Transport Layer Protection
Unprotected Transport of Sensitive Information
Weak Algorithm or Protocol Use

Lack of Resources & Rate Limiting
Lack of Resources & Rate Limiting

LLM
Direct Prompt Injection

Memory Corruption
Integer Overflow
Null Dereference

Security Misconfiguration
Debug Features Enabled
Disabled Security Features
Improper Permissions
Information Exposure

Sensitive Data Storage
Plaintext Storage of Sensitive Information

Vulnerable Components
Using Components from Untrusted Source
Using Known Vulnerable Components

Conceptual Content

Container Security
Database Security
DevSecOps Security
Introduction to Cloud Security
Introduction to Typescript Pulumi Security for AWS
Kubernetes Security
Non-Human Identities (NHI)
OAuth 2.0 Advanced Topics
OAuth 2.0 Security
OWASP Top 10 CI/CD (GitHub Actions)
OWASP Top 10 CI/CD (Jenkins Declarative)
OWASP Top 10 CI/CD (Jenkins Scripted)
SAML Security
Secure Implementation of Docker
Secure Implementation of Kubernetes
Secure Password Storage
Securely Accessing APIs Using OAuth 2.0
Securely Granting Access to an API Using OAuth 2.0
Securing Amazon Web Services (AWS)
Securing Google Cloud Platform (GCP)
Securing Infrastructure as Code
Securing Microsoft Azure

Embedded Topics & Content

Audience: Embedded and Automotive Industry Engineers

Vulnerability Topics

Access Control
Insecure Direct Object Reference
Missing Function Level Access Control
Using input from untrusted sources

Authentication
Improper Authentication
Insecure Password Change Function
Insufficient Anti-Automation
Insufficiently Protected Credentials
Weak Password Policy

Business Logic
Insufficient Validation
Logical Error

Denial of Service
Failure to Release Resource
Regular Expression DoS
Uncaught Error Handling

Information Exposure
Debug Information
Error Details
Sensitive Data Exposure

Injection Flaws
Code Injection
Email Injection
LDAP Injection
Log Forging
OS Command Injection
Path Traversal
Resource Injection
SQL Injection
XPath Injection

Insecure Cryptography
Exposed Key
Insecure Randomness
Reuse of Initialization Vector
Weak Algorithm Use

Insufficient Logging and Monitoring
Insufficient Logging and Monitoring

Insufficient Transport Layer Protection
Unprotected Transport of Credentials
Unprotected Transport of Sensitive Information
Weak Algorithm or Protocol Use

Memory Corruption
Buffer Overflow
Double Free
Format String Vulnerabilities
Heap Overflow
Illegal Pointer Value
Integer Overflow
Null Dereference
Race Conditions
Stack Overflow
Type Confusion
Uninitialized Variable
Use After Free

Security Misconfiguration
Debug Features Enabled
Disabled Security Features
Improper Permissions

Sensitive Data Storage
Plaintext Storage of Passwords
Plaintext Storage of Sensitive Information

Session Handling
Insufficient Session Expiration
Weak Session Token Generation

Side Channel Vulnerability
Data sent to 3rd parties (analytics, error handling)

Vulnerable Components
Using Components from Untrusted Source
Using Known Vulnerable Components

XML External Entities (XXE)
XML External Entities (XXE)

Embedded Topics & Content Continued

Conceptual Content

Advanced Embedded Security
Introduction to Automotive Security
Introduction to Embedded Security
Introduction to Securing the Internet of Things
Secure Communication for the Internet of Things

Frontend Topics & Content

Audience: Frontend Engineers

Vulnerability Topics

Cross-Site Scripting (XSS)
DOM-Based Cross-Site Scripting

Information Exposure
Sensitive Data Exposure

Injection Flaws
Code Injection
CSS Injection

Security Misconfiguration
Clickjacking
Disabled Security Features
Improper or Missing HTTP Headers

Side Channel Vulnerability
Clipboard buffer caching
Data sent to 3rd parties (analytics/error handling)

Unvalidated Redirects and Forwards
Unvalidated Redirects and Forwards

Vulnerable Components
Using Components from Untrusted Source
Using Known Vulnerable Components

Conceptual Content

Application Security Concepts
Defensive Programming for HTML5 Security
Foundations of Software Security
GDPR for Developers and Architects
Introduction to HTML5 Security
Open-Source Software (OSS)
PCI DSS v4.0 Concepts and Compliance
Securely Accessing APIs Using OAuth 2.0
Vulnerability Handling
Web Application Security 101

Mobile Topics & Content

Audience: Mobile Engineers, Android Engineers, iOS Engineers

Vulnerability Topics

Broken Cryptography
Improper Use of Cryptography Algorithm
Insecure Generation of Encryption Keys
Insecure Storage of Encryption Keys
Reuse of Initialization Vector
Use of Encoding
Use of Hardcoded Keys
Use of Insecure/Deprecated Algorithms
Use of Short Encryption Keys

Client Code Quality
Format String Vulnerabilities
Improper Memory Management
Overflow Vulnerabilities

Client Side Injection
DOM-Based Cross-Site Scripting
JavaScript Injection
SQL Injection
Untrusted third party sites
XML Injection

Code Tampering
Backups Enabled
Tampering Detection

Extraneous Functionality
Autofill Password
Debugging Features Enabled

Improper Platform Usage
Incorrect Activity Configuration
Insecure File/Directory Permissions
Insecure Use of Pasteboard
Misuse of Broadcast Receivers
Misuse of Intents
Misuse of Keychain
Misuse of URL Schemes
Tapjacking
Webview Settings

Improper Session Handling
Client Side Session Token Generation
Improper Flags in Cookie Headers
Improper Timeout of Session ID
Same Session ID With Change in Privilege
Weak Session Token Generation Algorithm

Insecure Authentication
Client Side Authentication for Authenticating to Server
Hardcoded API Keys
Misuse of Fingerprint
Password Enumeration
Storing Credentials With 'Remember Me' Functionality
Use of Spoofable Parameters for Authentication
Use of Weak Passwords
Username Enumeration
Weak Lockout Mechanism

Insecure Authorization
Insecure Direct Object Reference
Using inputs from untrusted sources

Insecure Data Storage
Plaintext Storage of Credentials
Storage in Log Files
Storage in Plist or XML Files
Storage in SQLite Databases
Storage of Binary Cookies
Storage on SDCard/External Storage

Insufficient Transport Layer Protection
Communication Over Cleartext Protocol
Improper Certificate Pinning Configuration
Trusting Self-Signed or Untrusted Certificates
Weak Certificate Validation
Weak Cipher Suites

Lack of Binary Protections
Lack of Adequate Security Controls
No Code Obfuscation
No Protection from Debuggers
No Protection from Piracy
No Protection from Runtime Injection

Mobile Topics & Content Continued

Reverse Engineering
Code Information Leakage
Emulation Detection

Unintended Data Leakage
Analytics Data Sent to 3rd Parties
Application Backgrounding Screenshots
Browser Cookie Objects
Copy/Paste Buffer Caching (Pasteboard)
Keyboard Caching
Logging Sensitive Information
Request/Response Caching
Use of Hardcoded Secrets

Conceptual Content

Foundations of Mobile Security
Fundamentals of iOS (Objective-C)
Fundamentals of iOS (Swift)
Java Android Security
Kotlin Advanced Android Security
Kotlin Android Security
Secure Programming for iOS (Objective-C)
Secure Programming for iOS (Swift)

Product Management Topics & Content

Audience: Product Management, Project Management, Business Analysts

Vulnerability Topics

Authentication
Improper Authentication
Username Enumeration
Insufficient Anti-Automation
Weak Password Policy
Insecure Password Change Function
Use of Single-factor Authentication

Business Logic
Insufficient Validation
Logical Error

Extraneous Functionality
Autofill Password

File Upload Vulnerability
File Upload Vulnerability: Unrestricted File Upload

Insecure Authentication
Weak Lockout Mechanism
Misuse of Fingerprint
Using inputs from untrusted sources

LLM
Direct Prompt Injection
Indirect Prompt Injection
Excessive Agency
Misinformation
Vector and Embedding Weaknesses

Conceptual Content

AI Agents and their Protocols (MCP, A2A and ACP)
Foundations of Mobile Security
Foundations of Software Security
GDPR for Development and Project Managers
Introduction to AI Risk & Security
Introduction to CCPA
Introduction to GDPR
Introduction to the Cyber Resilience Act (CRA)
Low-Code and No-Code (LCNC)
PCI DSS v4.0 Concepts and Compliance
Security Requirements
Vibe Coding: Risk Management Framework
Web Application Security 101

Quality Assurance Topics & Content

Audience: Quality Assurance Engineers, Quality Assurance Managers, Quality Assurance Testers

Vulnerability Topics

Access Control
Insecure Direct Object Reference
Missing Function Level Access Control

Authentication
Improper Authentication

Business Logic
Insufficient Validation

Cross-Site Scripting (XSS)
DOM-Based Cross-Site Scripting
Reflected Cross-Site Scripting
Stored Cross-Site Scripting

File Upload Vulnerability
Unrestricted File Upload

Information Exposure
Error Details
Sensitive Data Exposure

Injection Flaws
Deserialization of Untrusted Data
NoSQL Injection
OS Command Injection
Path Traversal
SQL Injection

Insecure Cryptography
Insecure Randomness
Weak Algorithm Use

Insufficient Logging and Monitoring
Insufficient Logging and Monitoring

LLM
Direct Prompt Injection
Excessive Agency
Improper Output Handling
Indirect Prompt Injection
Sensitive Information Disclosure
Vector and Embedding Weaknesses

Mass Assignment
Mass Assignment

Memory Corruption
Buffer Overflow|Null Dereference

Security Misconfiguration
Disabled Security Features

Sensitive Data Storage
Plaintext Storage of Passwords
Plaintext Storage of Sensitive Information

Server-Side Request Forgery
Server-Side Request Forgery

Session Handling
Exposed Session Tokens

Vulnerable Components
Log4j
Using Components from Untrusted Source
Using Known Vulnerable Components

XML External Entities (XXE)
XML External Entities (XXE)

Conceptual Content

Attack and Defence
Overview of Application Security Testing
Risk-Based Security Testing Strategy
Vulnerability Handling

securecodewarrior.com