STIG Series | Application Security and Development (STIG) - TT8810

Recognize Actual and Potential Software Vulnerabilities, Implement and Test Defenses & More

Duration: 5 Days
Skill Level: Intermediate
Available Format: Instructor-Led Online ; On Public Schedule

DISA's Application Security and Development STIG, in conjunction with the associated checklist, provides a comprehensive listing of requirements and needs for improving and maintaining the security of software applications and systems within the Department of Defense. This course fills in the context, background, and best practices for fulfilling those requirements and needs. As with all of our courses, we maintain tight synchronization between the latest DISA releases and our materials. A key component to our coverage of DISA's Security Technical Implementation Guides (STIGS), this course is a companion course with several developer-oriented courses and seminars

Application Security and Development (STIGs) is an immersive, comprehensive, STIG-driven application security training course essential for developers, designers, architects, QA, Testing, and other personnel who need to deliver and/or evaluate secure applications within the DOD. In addition to teaching a solid understanding of applicable ASD STIGS, this course digs deep into the features to examine for those STIGs with hands-on programming.

The ASD STIGs are covered in a series of categories that cover groups for SITGs related to a general topic area such as access control, cryptography and information protection, authentication, etc. Each category will start with a listing of applicable STIGs, followed by the technical foundations for those STIGs. These technical foundations will be supplemented by labs and examples to stimulate critical thinking. Each category will end with an extensive lab of each covered STIG during which students will evaluate the STIG itself, the check text (verification), and fix text.

What You'll Learn

Overview

DISA's Application Security and Development STIG, in conjunction with the associated checklist, provides a comprehensive listing of requirements and needs for improving and maintaining the security of software applications and systems within the Department of Defense. This course fills in the context, background, and best practices for fulfilling those requirements and needs. As with all of our courses, we maintain tight synchronization between the latest DISA releases and our materials. A key component to our coverage of DISA's Security Technical Implementation Guides (STIGS), this course is a companion course with several developer-oriented courses and seminars

Application Security and Development (STIGs) is an immersive, comprehensive, STIG-driven application security training course essential for developers, designers, architects, QA, Testing, and other personnel who need to deliver and/or evaluate secure applications within the DOD. In addition to teaching a solid understanding of applicable ASD STIGS, this course digs deep into the features to examine for those STIGs with hands-on programming.

The ASD STIGs are covered in a series of categories that cover groups for SITGs related to a general topic area such as access control, cryptography and information protection, authentication, etc. Each category will start with a listing of applicable STIGs, followed by the technical foundations for those STIGs. These technical foundations will be supplemented by labs and examples to stimulate critical thinking. Each category will end with an extensive lab of each covered STIG during which students will evaluate the STIG itself, the check text (verification), and fix text.

Objectives

Students who attend this course will leave armed with the skills required to recognize actual and potential software vulnerabilities, implement defenses for those vulnerabilities, and test those defenses for sufficiency. This course introduces developers to the most common security vulnerabilities faced by web applications today. Each vulnerability is examined from a Java/JEE perspective through a process of describing the threat and attack mechanisms, recognizing associated vulnerabilities, and, finally, designing, implementing, and testing effective defenses.

Multiple practical labs reinforce these concepts with real vulnerabilities and attacks. Students are then challenged to design and implement the layered defenses they will need in defending their own applications.

Students who attend this course will leave armed with the skills required to recognize actual and potential software vulnerabilities based on STIGs and STIG checklists. This course introduces developers to the most common security vulnerabilities faced by web applications today.

Working in a dynamic, lab-intensive hands-on coding environment students will learn to:

- Understand potential sources for untrusted data
- Understand the consequences for not properly handling untrusted data such as denial of service, cross-site scripting, and injections
- Establish the first axiom in security analysis of ALL web applications for this course and beyond
- Establish the first axiom in addressing ALL security concerns for this course and beyond
- To test web applications with various attack techniques to determine the existence of and effectiveness of layered defenses
- Prevent and defend the many potential vulnerabilities associated with untrusted data
- Understand the vulnerabilities of associated with authentication and authorization
- To detect, attack, and implement defenses for authentication and authorization functionality and services
- Understand the dangers and mechanisms behind Cross-Site Scripting (XSS) and Injection attacks
- To detect, attack, and implement defenses against XSS and Injection attacks
- Understand the concepts and terminology behind defensive, secure, coding
- Perform both static code reviews and dynamic application testing to uncover vulnerabilities in web applications
- Design and develop strong, robust authentication and authorization implementations

The course provides a solid foundation in basic terminology and concepts, extended and built upon throughout the engagement. Students will examine various recognized attacks against web applications. Processes and best practices are discussed and illustrated through both discussions and group activities. Attending students will be led through a series of advanced topics comprised of integrated lectures, group discussions and comprehensive demonstrations.

Need different skills or topics? If your team requires different topics or tools, additional skills or custom approach, this course may be further adjusted to accommodate. We offer additional STIG, application security, secure coding, secure software development, hacking, database security, bug hunting and other related topics that may be blended with this course for a track that best suits your needs. Our team will collaborate with you to understand your needs and will target the course to focus on your specific learning objectives and goals.

Audience

This is an **intermediate -level** programming course, designed for experienced Java developers who wish to get up and running on developing well defended software applications using the STIG guidelines. Familiarity with Java and JEE is required and real world programming experience is highly recommended. Ideally students should have approximately 6 months to a year of Java working knowledge.

Pre-Requisites

This is an **intermediate -level** programming course, designed for experienced Java developers who wish to get up and running on developing well defended software applications using the STIG guidelines. Familiarity with Java and JEE is required and real world programming experience is highly recommended. Ideally students should have approximately 6 months to a year of Java working knowledge.

Agenda

Please note that this list of topics is based on our standard course offering, evolved from typical industry uses and trends. We'll work with you to tune this course and level of coverage to target the skills you need most. Topics, agenda and labs are subject to change, and may adjust during live delivery based on audience interests and skill-level.

Session: STIG Foundation

Lesson: DISA's Security Technical Implementation Guides (STIGs)

- The motivations behind STIGs
- Requirements that the various software development roles must meet
- Implementing STIG requirements and guidelines

Lesson: Why Hunt Bugs?

- The Language of Cybersecurity

- The Changing Cybersecurity Landscape
- The Human Perimeter
- Starting Point for ALL Security Analysis of Web Applications
- Lab: Case Study in Failure

Lesson: Safe and Appropriate Bug Hunting/Hacking

- Warning to All Bug Hunters
- Working Ethically
- Respecting Privacy
- Bug/Defect Notification
- Bug Hunting Pitfalls

Lesson: Removing Bugs

- Open Web Application Security Project (OWASP)
- OWASP Top Ten Overview
- Web Application Security Consortium (WASC)
- Common Weaknesses Enumeration (CWE)
- CERT Secure Coding Standard
- Microsoft Security Response Center
- Software-Specific Threat Intelligence

Session: Foundation for Securing Web Applications

Lesson: Principles of Information Security

- Security Is a Lifecycle Issue
- Minimize Attack Surface Area
- Layers of Defense: Tenacious D
- Compartmentalize
- Consider All Application States
- Do NOT Trust the Untrusted
- AppSec Dissection of the Verkada Exploit
- Tutorial: Working with Eclipse (JEE Version) and Apache TomEE 7x
- Tutorial: Working with the HSQL Database
- Lab: Case Study Setup and Review

Session: STIG Stomping 101

Lesson: Unvalidated Data

- Applicable STIGs

- Buffer Overflows
- Integer Arithmetic Vulnerabilities
- Defining and Defending Trust Boundaries
- Rigorous., Positive Specifications
- Whitelisting vs Blacklisting
- Challenges: Free-Form Text, Email Addresses, and Uploaded Files
- Lab: Defending Trust Boundaries
- Lab: Toolbox
- Lab: STIG Walk-Throughs

Lesson: Access Control

- Applicable STIGs
- Elevation of Privileges
- Insufficient Flow Control
- Unprotected URL/Resource Access/Forceful Browsing
- Lab: Unsafe Direct Object References
- Session Management
- HttpOnly and Security Headers
- Cross Site Request Forgeries (CSRF)
- CSRF Defenses
- Lab: Cross-Site Request Forgeries
- Spotlight: Verizon
- Lab: STIG Walk-Throughs

Lesson: Cryptographic Failures

- Applicable STIGs
- Identifying Protection Needs
- Evolving Privacy Considerations
- Options for Protecting Data
- Transport/Message Level Security
- Weak Cryptographic Processing
- Keys and Key Management
- NIST Recommendations
- Threats of Quantum Computing
- Steal Now, Crack Later Threat
- Lab: Defending Sensitive Data
- Lab: STIG Walk-Throughs

Lesson: Injection

- Applicable STIGs
- Injection Flaws
- SQL Injection Attacks Evolve
- Drill Down on Stored Procedures
- Other Forms of Server-Side Injection
- Minimizing Injection Flaws
- Lab: Defending Against SQL Injection
- Client-side Injection: XSS
- Persistent, Reflective, and DOM-Based XSS
- Best Practices for Untrusted Data
- Lab: Defending Against XSS
- Lab: STIG Walk-Throughs

Lesson: Security Misconfiguration

- Applicable STIGs
- System Hardening
- Risks with Internet-Connected Resources (Servers to Cloud)
- Minimalist Configurations
- Application Whitelisting
- Secure Baseline
- Segmentation with Containers and Cloud
- Resolution of External References
- Safe XML Processing
- Lab: Safe XML Processing
- Lab: STIG Walk-Throughs

Session: STIG Stomping 102

Lesson: Vulnerable and Outdated Components

- Applicable STIGs
- Vulnerable Components
- Software Inventory
- Managing Updates: Balancing Risk and Timeliness
- AppSec Dissection of Ongoing Microsoft Exchange Exploits
- Spotlight: Equifax
- Lab: STIG Walk-Throughs

Lesson: Identification and Authentication Failures

- Applicable STIGs

- Quality and Protection of Authentication Data
- Proper hashing of passwords
- Handling Passwords on Server Side
- Lab: Argon2 Hashing
- Lab: Identifying and Defending Authentication Assets
- Lab: Spotlight: SQL Server Administrators
- Lab: STIG Walk-Throughs

Lesson: Software and Data Integrity Failures

- Applicable STIGs
- Serialization/Deserialization
- Issues with Consuming Vulnerable Software
- Using Trusted Repositories
- CI/CD Pipeline Issues
- Protecting Software Development Resources
- Lab: STIG Walk-Throughs

Lesson: Security Logging and Monitoring Failures

- Applicable STIGs
- Detecting Threats and Active Attacks
- Best Practices for Determining What to Log
- Safe Logging in Support of Forensics
- Lab: STIG Walk-Throughs

Lesson: Server-Side Request Forgery (SSRF)

- Applicable STIGs
- Understanding SSRF
- Remote Resource Access Scenarios
- Complexity of Cloud Services
- SSRF Defense in Depth
- Positive Allow Lists
- Lab: STIG Walk-Throughs

Session: Additional Concerns, Services, and Rich Interfaces

Lesson: Database Security

- Applicable STIGs
- Design and Configuration
- Identification and Authentication

- Computing Environment
- Database Auditing
- Boundary Defenses
- Continuity of Service
- Vulnerability and Incident Management
- Lab: STIG Walk-Throughs

Lesson: Defending Web Services

- Applicable STIGs
- Web Service Security Exposures
- When Transport-Level Alone is NOT Enough
- Message-Level Security
- WS-Security Roadmap
- Web Service Attacks
- Web Service Appliance/Gateways
- Lab: Web Service Attacks
- Lab: STIG Walk-Throughs

Lesson: Defending Rich Interfaces and REST

- Applicable STIGs
- How Attackers See Rich Interfaces
- Attack Surface Changes When Moving to Rich Interfaces and REST
- Bridging and its Potential Problems
- Three Basic Tenets for Safe Rich Interfaces
- OWASP REST Security Recommendations
- Lab: STIG Walk-Throughs

Session: Moving Forward with Application Security

Lesson: Application Security and Development Checklists

- Checklist Overview, Conventions, and Best Practices
- Leveraging Common AppSec Practices and Control
- Actionable Application Security
- Additional Tools for the Toolbox
- Demo: Actionable AppSec
- Strength Training: Project Teams/Developers
- Strength Training: IT Organizations
- Lab: Recent Incidents

- Spotlight: Capital One

Lesson: SDL Process Overview

- Revisiting Attack/Defense Basics
- Types of Security Controls
- Attack Phases: Offensive Actions and Defensive Controls
- Secure Software Development Processes
- Shifting Left
- Lab: Risk Escalators

Optional Content

Emerging Game Changers in Web Application Security

- Quantum Computing: Timeframe
- Threats to Current Cryptography
- Post-Quantum vs. Quantum Cryptography
- Today's Defenses Against Tomorrow's Quantum Computing
- AI in Web Application Security
- AI-Powered Threat Detection
- AI for Secure Coding
- AI in Incident Response
- Challenges and Ethical Considerations in AI for Security

Related Courses

TT8120	Web Application Security Essentials: Understanding OWASP Risks and Fixes That Really Work

Student Materials: Each student will receive a Student Guide with course notes, code samples, software tutorials, diagrams and related reference materials and links (as applicable). Our courses also include step by step hands-on lab instructions and and solutions, clearly illustrated for users to complete hands-on work in class, and to revisit to review or refresh skills at any time. Students will also receive related (as applicable) project files, code files, data sets and solutions required for the hands-on work.

Classroom Setup Made Simple: Our dedicated tech team will work with you to ensure your classroom and lab environment is setup, tested and ready to go well in advance of the course delivery date, ensuring a smooth start to class and seamless hands-on experience for your students. We offer several flexible student machine setup options including guided manual set up for simple installation directly on student machines, or cloud based / remote hosted lab solutions where students can log in to a complete separate lab environment minus any installations, or we can supply complete turn-key, pre-loaded equipment to bring ready-to-go student machines to your facility.

Please inquire for details, options and pricing.

For More Information

Please contact us or call 844-475-4559 toll free for more information about our training services (instructor-led, self-paced or blended), coaching and mentoring services, public course enrollment or questions, partner programs, courseware licensing options and more.