

SC4013

APPLICATION SECURITY - LECTURE 1

Introduction to
Application Security

AGENDA

- Course Overview & Learning Objectives
- The Digital Battlefield: Importance of Web Application Security
- Detailed Examples of Application Security Incidents 2020-2021
- Understanding Web Applications
- The CIA Triad in Web Application Context
- Web Application Security Landscape
- Introduction to Secure Development Lifecycle (SDLC)
- Wrap-up and Q&A



COURSE OVERVIEW & LEARNING OBJECTIVES

- Course Structure
 - Lectures, tutorials, quizzes, team project
- Learning Objectives
 - Understand core concepts of web application security
 - Analyze OWASP Top 10 risks and mitigation strategies
 - Apply manual exploitation methods for common vulnerabilities
 - Implement and assess secure coding practices



COURSE EVALUATION

- 2 Quizzes (35% each), form team of 2, mini project (30%)
- ALL students must take both quizzes & participate in project
- **Make up quiz ONLY** for students with legitimate excuse such as MC.
- No credit for missing any quiz and project
- Quizzes (short questions & T/F)
- **Quiz 1 date: 24 Sept Tue 630pm (LT)**
- **Quiz 2 date: 29 Oct Tue 630pm (LT)**
- **Project Presentation: 29 Oct Tues 7:30pm – 9:30pm (LT) wk12**
 - (10 minutes per team of 2)

PROJECT MATTERS

- Form teams of 2. Once formed and confirmed, email me & cc to buddy
- All must present before me at end of semester.
- List of Suggestions at end of class
- Hands-on performing Web Application testing is expected.
- My expectation: aboutt 2 weekend's effort
- Judging criteria:
 - Technical depth/merit of project
 - Quality of presentation
 - Answers in Q&A


HOW TO DO WELL

- Attend lectures & tutorials (keep up with your peers)
- TIME MANAGEMENT
 - How you spend your time from morning to 6pm
- Take note of important parts of lectures
- Try to work out some tutorial questions before attending tutorial
 - This will help you understand where your weak areas are
- Pls do some hands-on testing with the tools – quiz will test these
- Attending physical tutorial will give you great opportunity to ask me questions first-hand

BEFORE WE START – ABOUT ME


[HTTPS://WWW.LINKEDIN.COM/IN/VICTOR-KEONG-81A8043/](https://www.linkedin.com/in/victor-keong-81a8043/)

[VICTOR.KEONG@NTU.EDU.SG](mailto:victor.keong@ntu.edu.sg)



Victor Keong (He/Him)


GenAI Cybersecurity Leader | Certified Microsoft CoPilot for Security Ninja | 智能网安先锋 |


 Top Information Security Voice

Singapore · [Contact info](#)

3,462 followers · 500+ connections

[Open to](#) [Add profile section](#) [Enhance profile](#) [More](#)

 **Avanade**

 **Western University**



CyberSecurity + Modern Workplace Solutions Leader, Southeast Asia

Avanade · Full-time
Jan 2024 · Present · 8 mos
Singapore

Catalyzing digital revolutions and fortifying cyber frontiers at Avanade's South-East Asian nexus. Architecting tomorrow's workspace while battling digital dragons. Surfing the GenAI wave with the mightiest cyber-squad this side of the equator – we're not just ahead of the curve, we're bending it!



Senior Director & Field CISO, APJ

Cohesity · Full-time
Aug 2022 · Oct 2023 · 1 yr 3 mos
Singapore

Cohesity is a late-stage unicorn with amazingly powerful data management technology with three key focuses – data security, multi-cloud and generative AI. Grounded with my start in Vulnerability Management almost 30 years ago, I bring my considerable depth and breadth of Infosec experience to drive best-in-class cybersecurity practices for the customers at Cohesity's multi-cloud platform. Keynote speakers in several APJ events, partnered as well as leading security events in APJ.



Senior Director & Global CISO Whisperer

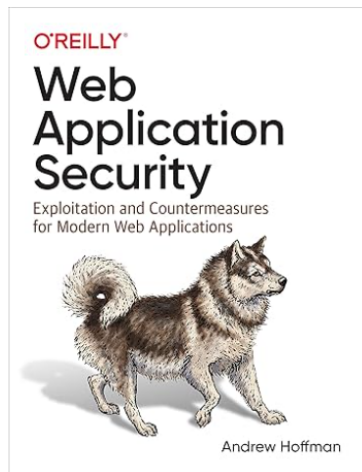
Checkmarx · Full-time
Mar 2021 · Jun 2022 · 1 yr 4 mos
Singapore

Hired as the very first Global CISO Whisperer for Checkmarx – continuing to build on trust building and key stay-awake issues for CISOs. I also helped Checkmarx re-start in Japan, with a significant focus on the Big 3 automotive companies in Japan, plus their associated companies and subsidiaries. The focus on AppSec is returning me to my roots in vulnerability management, however this time round, my focus is to help customers build a sustainable DevSecOps platform/environment/process

BEFORE WE START – COMPANION TEXTBOOK

amazon.sg/dp/1492053112?ref_=mr_referred_us_sg_sg

Books › Computing and Internet › Digital Lifestyle



Roll over image to zoom in

Look inside

Web Application Security: Exploitation and Countermeasures for Modern Web Applications Paperback – 17

March 2020

by [Andrew Hoffman](#) (Author)

4.3 114 ratings

[See all formats and editions](#)

Get \$5 Off with Mastercard W/WE Cards 1 applicable promotion

There is a newer edition of this item:



[Web Application Security: Exploitation and Countermeasures for Modern Web Applications](#)

\$578.96

(5)

In stock

While many resources for network and IT security are available, detailed knowledge regarding modern web application security has been lacking, until now. This practical guide provides both offensive and defensive security concepts that software engineers can easily learn and apply.

Andrew Hoffman, a senior security engineer at Salesforce, introduces three pillars of web application security: recon, offense, and defense. You'll learn methods for effectively researching and analyzing modern web applications, including those you don't have direct access to. You'll also learn how to break into web applications using the latest hacking techniques. Finally, you'll learn how to develop mitigations for use in your own web applications to protect against hackers.

- Explore common vulnerabilities plaguing today's web applications
- Learn essential hacking techniques attackers use to exploit applications
- Map and document web applications for which you don't have direct access
- Develop and deploy customized exploits that can bypass common defenses
- Develop and deploy mitigations to protect your applications against hackers
- Integrate secure coding best practices into your development lifecycle
- Get practical tips to help you improve the overall security of your web applications

[Read less](#)

Paperback
\$57.72

Other Used and New from **\$57.72**

\$57.72

FREE delivery Wednesday, 7 August.
[Details](#)

Or fastest delivery **Tuesday, 6 August.**
[Details](#)

Deliver to Victor - Singapore 808295

Only 1 left in stock.

[Add to Cart](#)

[Buy Now](#)

Secure transaction

Ships from **Amazon SG**

Sold by **Amazon SG**

Return policy:
Eligible for Return, Refund or Replacement within 15 days of receipt

☐ **Yes, I want FREE Delivery**

Enjoy FREE & FAST delivery with
Amazon Prime

amazon prime

BEFORE WE START

- Current topics in Information Security
 - The Crowdstrike Incident



LinkedIn application security Singapore

Jobs Entry level 1 Date posted Company Remote Easy Apply All filters Reset

application security in Singapore 168 results Set alert

- Consultant, Manager, Director** JAC Recruitment Singapore, Singapore (On-site) Promoted • Be an early applicant
- Assistant Director/Senior Manager (Cybersecurity Oversight)** CAAS Civil Aviation Authority of Singapore Singapore, Singapore (On-site) 1 connection works here Promoted • Be an early applicant
- System Assurance Consultant** Ricardo plc Singapore, Singapore (Remote) Promoted
- Information Security Officer - Scientific IT, ITSS (3 years renewable contract)** A*STAR - Agency for Science, Technology and Research Singapore, Singapore (On-site) 2 connections work here Promoted
- GENERAL APPLICATION - SECURITY ROLES** G4S Singapore, Singapore (On-site)

Consultant, Manager, Director JAC Recruitment
Singapore, Singapore · 2 weeks ago · 7 people clicked Apply
On-site · Full-time · Entry level
1 connection works here
Skills: Team Management, Teamwork
Am I a good fit for this job? How can I best position myself?
Apply Save

About the job
PR/093857
アジア最大級のコンサルティングファームにて、戦略コンサルタントを募集! 日系企業の海外事業戦略 / 進出戦略 / 新規事業戦略等のプロジェクトリーダーを担当頂きます
【 Roles and Responsibilities 】
■ M&A アドバイザリー業務
(案件開拓、Business Due-Diligence、買収実務支援、PMI 等)
■ 経営コンサルティング業務

THE DIGITAL BATTLEFIELD: IMPORTANCE OF WEB APPLICATION SECURITY

- Evolving cybersecurity landscape
 - 50% of all security vulnerabilities are in web applications
 - Average cost of a data breach: \$4.24 million
- Recent real-world examples
 - SolarWinds Supply Chain Attack (2020-2021)
 - Microsoft Exchange Server Vulnerabilities (2021)
 - Log4j Vulnerability (2021)
 - Kaseya VSA Supply Chain Ransomware Attack (2021)

DETAILED EXAMPLES OF APPLICATION SECURITY INCIDENTS 2020-2021: SOLARWINDS SUPPLY CHAIN ATTACK (2020- 2021)

- What Happened
 - Attackers inserted malicious code into SolarWinds' Orion software updates
 - Updates were distributed to thousands of customers
- How It Happened
 - Attackers gained access to SolarWinds' development environment
 - Malicious code was inserted into the Orion software build process
 - Code was compiled, signed, and delivered through legitimate software update process
 - Malware was activated when customers installed updates, creating a backdoor in their systems
- Relation to Application Security
 - Code Integrity: Importance of securing entire software development lifecycle
 - Supply Chain Security: Far-reaching consequences of compromising a single point in software supply chain

DETAILED EXAMPLES OF APPLICATION SECURITY INCIDENTS 2020- 2021: MICROSOFT EXCHANGE SERVER VULNERABILITIES (2021)

- What Happened
 - Microsoft disclosed four zero-day vulnerabilities in Exchange Server
 - Actively exploited by HAFNIUM and other threat actors
- How It Happened
 - Vulnerabilities allowed attackers to bypass authentication and execute code as privileged users
 - Attackers chained vulnerabilities to gain initial access, create web shells, steal data, and further compromise networks
 - Attack was wormable and could spread automatically
- Relation to Application Security
 - Emphasizes importance of timely patching and updates
 - Highlights need for robust authentication mechanisms
 - Demonstrates risks of running applications with excessive privileges
 - Some vulnerabilities stemmed from improper input handling

DETAILED EXAMPLES OF APPLICATION SECURITY INCIDENTS 2020-2021: LOG4J VULNERABILITY (2021)

- What Happened
 - A critical zero-day vulnerability (CVE-2021-44228) was discovered in Log4j
 - Affected millions of applications and devices worldwide
- How It Happened
 - Vulnerability existed in Log4j's JNDI lookup feature
 - Attackers could exploit this by sending a specially crafted request
 - When logged, this payload could trigger remote code execution on the server
 - Widespread use of Log4j made this vulnerability particularly severe
- Relation to Application Security
 - Highlights the risks associated with using third-party libraries
 - Demonstrates the critical need to sanitize all input
 - The vulnerability's impact was exacerbated in systems where applications had unnecessary privileges

DETAILED EXAMPLES OF APPLICATION SECURITY INCIDENTS 2020-2021: KASEYA VSA SUPPLY CHAIN RANSOMWARE ATTACK (2021)

- What Happened
 - REvil ransomware group exploited zero-day vulnerabilities in Kaseya's VSA software
- How It Happened
 - Attackers exploited vulnerabilities in the Kaseya VSA web interface to bypass authentication and upload malicious payloads
 - Used Kaseya's legitimate functionality to deploy ransomware to MSPs' clients
 - Timed just before a holiday weekend in the US, aiming to maximize impact and ransom potential
 - Up to 1,500 businesses were affected, with ransom demands of up to \$70 million
- Relation to Application Security
 - Authentication and Access Control: Highlights the critical importance of robust authentication mechanisms in web applications
 - Input Validation: The vulnerabilities likely involved improper handling of user input in the web interface

UNDERSTANDING WEB APPLICATIONS: CLIENT-SIDE (FRONT-END)



- Components
 - Web Browser
 - HTML (Structure)
 - CSS (Styling)
 - JavaScript (Client-side functionality)
- Potential Attack Vectors
 - Cross-Site Scripting (XSS)
 - Cross-Site Request Forgery (CSRF)
 - Client-side validation bypass
 - DOM-based vulnerabilities
 - Clickjacking
 - Local storage/cookie theft



UNDERSTANDING WEB APPLICATIONS: NETWORK LAYER

- Components
 - HTTP/HTTPS protocols
 - DNS (Domain Name System)
 - TCP/IP
- Potential Attack Vectors
 - Man-in-the-Middle (MITM) attacks
 - DNS spoofing/cache poisoning
 - SSL/TLS vulnerabilities
 - Session hijacking
 - Traffic sniffing

UNDERSTANDING WEB APPLICATIONS: SERVER-SIDE (BACK-END)



- Components
 - Web Server (e.g., Apache, Nginx)
 - Application Server (e.g., Tomcat, Gunicorn)
 - Server-side Programming Language (e.g., PHP, Python, Java)
- Potential Attack Vectors
 - Remote Code Execution (RCE)
 - File Inclusion vulnerabilities
 - Server misconfiguration
 - Improper error handling
 - Authentication bypass
 - Authorization flaws
 - Server-Side Request Forgery (SSRF)



UNDERSTANDING WEB APPLICATIONS: DATABASE

- Components
 - Database Management System (e.g., MySQL, PostgreSQL, MongoDB)
 - Stored data
 - Database queries
- Potential Attack Vectors
 - SQL Injection
 - NoSQL Injection
 - Excessive privilege exploitation
 - Insecure direct object references
 - Sensitive data exposure



UNDERSTANDING WEB APPLICATIONS: THIRD-PARTY COMPONENTS AND APIS

- Components
 - External libraries and frameworks
 - Third-party APIs
 - Content Delivery Networks (CDNs)
- Potential Attack Vectors
 - Supply chain attacks
 - Vulnerable dependencies
 - API key exposure
 - Insecure API endpoints
 - Malicious package injection

UNDERSTANDING WEB APPLICATIONS: DEVELOPMENT AND DEPLOYMENT PIPELINE



- Components
 - Version Control Systems
 - CI/CD tools
 - Containerization and orchestration platforms
- Potential Attack Vectors
 - Code repository breaches
 - Insecure build processes
 - Container escape vulnerabilities
 - Misconfigured cloud services
 - Secrets management issues



THE CIA TRIAD IN WEB APPLICATION CONTEXT: CONFIDENTIALITY IN WEB APPLICATIONS

- Definition: Ensuring data is kept secret from unauthorized parties
- Key Points:
 - Data Protection
 - Access Controls
 - Information Disclosure Prevention
 - Examples of Confidentiality Breaches
 - Best Practices

THE CIA TRIAD IN WEB APPLICATION CONTEXT: INTEGRITY IN WEB APPLICATIONS

- Data Integrity
 - Ensuring data remains accurate, consistent, and trustworthy
- Code Integrity
 - Ensuring code is free from vulnerabilities and malicious code
- Transaction Integrity
 - Ensuring transactions are processed correctly and securely
- Examples of Integrity Breaches
 - Examples of how integrity can be compromised in web applications
- Best Practices
 - Recommended practices for maintaining integrity in web applications



THE CIA TRIAD IN WEB APPLICATION CONTEXT: AVAILABILITY IN WEB APPLICATIONS

- Definition: Ensuring information and resources are accessible when needed
 - System Uptime
 - Performance Optimization
 - Resilience Against Attacks
 - Examples of Availability Issues
 - Best Practices

THE CIA TRIAD IN WEB APPLICATION CONTEXT: BALANCING THE CIA TRIAD IN WEB APPLICATIONS



Trade-offs between CIA components



Risk Assessment



Regulatory Compliance



User Experience



Best Practices

WEB APPLICATION SECURITY LANDSCAPE: THREAT ACTORS IN WEB APPLICATION SECURITY

- Script Kiddies
 - Inexperienced hackers using pre-made tools and scripts
 - Limited technical knowledge
 - Automated scanning for common vulnerabilities
 - Generally lower risk
- Hacktivists
 - Individuals or groups who hack for political or social causes
 - Motivated by ideology rather than financial gain
 - Distributed Denial of Service (DDoS) attacks
 - Reputational damage to targeted organizations
- Organized Crime
- Nation-State Actors

WEB APPLICATION SECURITY LANDSCAPE: OWASP TOP 10 OVERVIEW

- Introduction to OWASP Top 10
 - A standard awareness document for developers and web application security
 - Identify the most critical security risks to web applications
 - Provide guidance on how to address these risks
 - Widely recognized standard in the industry
 - Updated periodically to reflect evolving threat landscape
 - Usage: As a checklist for secure development practices, to prioritize security efforts in web application projects, as a baseline for application security testing
- OWASP Top 10 - 2021 Edition
 - 1. Broken Access Control
 - 2. Cryptographic Failures
 - 3. Injection
 - 4. Insecure Design

INTRODUCTION TO SECURE DEVELOPMENT LIFECYCLE (SDLC): OVERVIEW OF SECURE SDLC

- Definition: A process that integrates security practices into every phase of the software development lifecycle.
 - Purpose: To identify and address security vulnerabilities early in the development process.
 - Benefits: Improved security posture of the final product and reduced costs associated with fixing security issues post-deployment.
 - Key Principles: Security by design, continuous security testing and improvement, and shared responsibility for security across all team members.
 - Challenges: Initial implementation can be time-consuming and requires ongoing training and culture change.

INTRODUCTION TO SECURE DEVELOPMENT LIFECYCLE (SDLC): PLANNING PHASE

- Threat Modeling: A structured approach to identifying, quantifying, and addressing security risks
 - Key components: Identifying assets, creating an architectural overview, decomposing the application, identifying threats, documenting threats, rating threats
 - Methodologies: STRIDE, DREAD, Attack trees
 - Tools: Microsoft Threat Modeling Tool, OWASP Threat Dragon
 - Benefits: Provides a systematic view of potential security threats, helps prioritize security efforts, informs security requirements for the project, promotes a security-minded approach from the start
- Security Requirements Gathering: Defines the security standards the application must meet
- Security Risk Assessment: To identify and prioritize potential security risks to the application

INTRODUCTION TO SECURE DEVELOPMENT LIFECYCLE (SDLC): DESIGN PHASE

- Definition: Design of application's structure with security in mind
- Key Elements
 - Network segmentation, DMZ, internal network isolation
 - Access control models: RBAC, ABAC
 - Encryption strategies: Data at rest, data in transit
 - Security patterns: Secure session management, secure communication protocols
- Best Practices
 - Defense in depth, principle of least privilege
 - Separation of duties, fail-safe defaults
- Considerations
 - Scalability, performance impact, compliance with standards

INTRODUCTION TO SECURE DEVELOPMENT LIFECYCLE (SDLC): IMPLEMENTATION PHASE

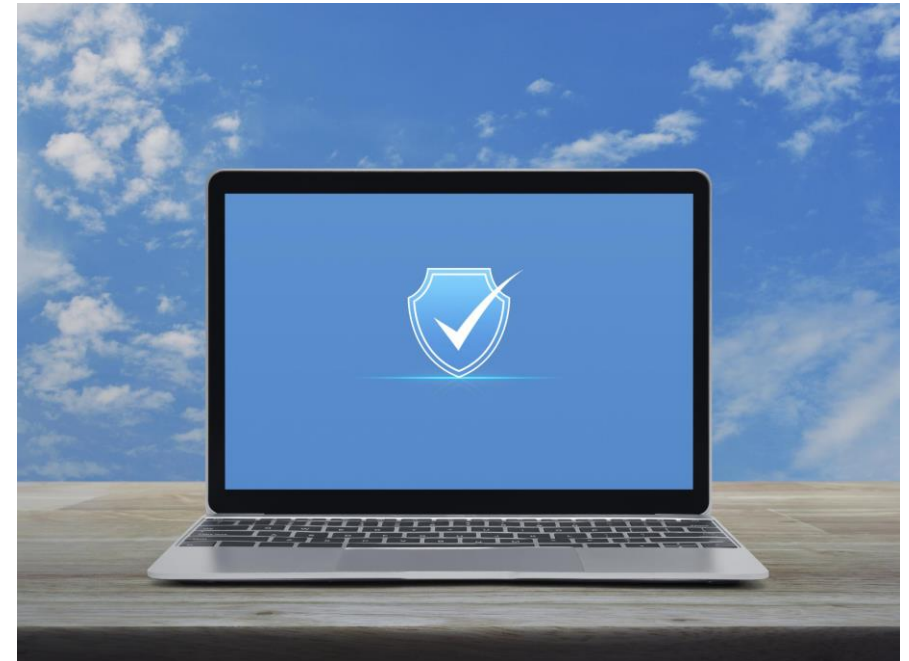
- Input validation and output encoding
 - Implementing strict input validation on all user inputs
 - Using parameterized queries to prevent SQL injection
 - Context-specific output encoding to prevent XSS
- Proper error handling and logging
 - Implementing centralized error handling
 - Avoiding sensitive information disclosure in error messages
- Secure use of cryptography
- Safe memory management (for languages like C/C++)
- Secure session management
- Language-specific security guidelines
- Tools

INTRODUCTION TO SECURE DEVELOPMENT LIFECYCLE (SDLC): TESTING PHASE

- Security Testing: Identify security vulnerabilities in the implemented application
 - Types of testing: DAST, IAST, Penetration testing, Fuzz testing
 - Key areas to test: Authentication and authorization, Input validation and output encoding, Session management, Error handling and logging, Cryptography implementation
 - Best practices: Integrating security testing into the CI/CD pipeline, Combining automated and manual testing approaches, Regular security testing
- Code Review: Security-focused

INTRODUCTION TO SECURE DEVELOPMENT LIFECYCLE (SDLC): DEPLOYMENT PHASE

- Key Areas for Secure Configuration
 - Server Hardening
 - Database Security Configuration
 - Application Server Security Settings
 - Network Security
- Best Practices for Secure Configuration
 - Use of Configuration Management Tools
 - Implementation of the Principle of Least Privilege
 - Regular Configuration Audits



INTRODUCTION TO SECURE DEVELOPMENT LIFECYCLE (SDLC): MAINTENANCE PHASE

- Patch Management
 - Process of acquiring, testing, and installing patches for the application and its components
 - Key practices include regular vulnerability scanning, prioritization of patches based on risk, testing patches in a staging environment before deployment, and maintaining a patch management policy
- Continuous Monitoring
 - Purpose is to detect security issues in the live application
 - Components include log analysis, intrusion detection/prevention systems, security information and event management (SIEM) systems, and regular vulnerability scans
- Incident Response and Recovery
 - Process of addressing and recovering from security incidents
- Regular Security Assessments



WRAP-UP AND Q&A

- Recap of key points
- Preview of next lecture
- Open floor for questions and discussions