Rui Pereira, B.Sc. (Hons), CISSP, CISA, CPTE / CPTC
Principal Consultant, WaveFront Consulting Group
ruiper@wavefrontcg.com | 1 (604) 961-0701 | www.wavefrontcg.com

Almost every other day, we hear about a new data breach where a major company loses millions of usernames, passwords, credit card numbers and banking transactions after falling victim to a cyber-attack. In many (if not most) of those cases the attack vector is a flaw in a web application such as SQL Injection (TalkTalk, VTech, Wetherspoon, Bell Canada, Drake Intl., Archos), Cross-Site Scripting (XSS) (Wordpress, E-Bay, Linkedin, Amazon, Craigslist), Broken Access Control (Steam, MacKeeper, United Airlines, Snapchat, Citigroup, Citroen), etc. In May 2015 WhiteHat Security reported that "*86% of all websites (we) tested…had at least one serious vulnerability, and most of the time, far more than one – 56% to be precise*". And by serious they mean vulnerabilities *"…in which an attacker could take control over all, or some part, of the website, compromise user accounts…, access sensitive data, violate compliance requirements, and possibly make headline news. In short, serious vulnerabilities are those that should really be fixed!"*

Cyber criminals, hacktivists, script kiddies and other ne'er-do-wells are always searching for ways to make easy money, cause embarrassment or prove their hacking creds, and insecurely coded web applications provide easy pickings. Given that most organizations and developers are not aware of the problem, this is a disaster doomed to constant repetition – take a look at the OWASP / WASC Web Hacking Incidents Database (WHID) at https://www.owasp.org/index.php/OWASP_WASC_Web_Hacking_Incidents_Database_Project for some examples.

Course Description

This four-day course aims to provide web application developers with an understanding of application security issues and attack vectors, and the skills, tools and techniques necessary to code defensively against web attacks. We show how hackers can abuse web applications, and what developers can do to prevent this. We also show how developers and testers can test their own applications in order to determine if these are susceptible to web application attacks. The material is also of interest to software testers (QA); system, network and database administrators; software, network and system architects; Information Security personnel; and Compliance Officers (Privacy, SOX, PCI, etc.)

Although we use Java and .NET examples throughout, the material is mainly platform agnostic. The idea is to teach fundamental concepts which can be applied no matter the web application development platform in use. We concentrate on platform-specific security issues on Day 4, and spend some time on Microsoft .Net, Java, PHP, Oracle, Ajax and HTML 5. In each area, the course covers theoretical foundations, common implementation pitfalls, details on historical exploits, suggested security policies, implementation best practices, and pseudo-code examples.

The course involves both lectures and hands-on labs and exercises. We use mainly open source security tools and sample web applications, although some commercial tools are discussed and demonstrated. All necessary equipment and software is provided.

Learning Outcomes

At the end of this course the student should be able to:

- Understand and be able to implement techniques to address requirements for the confidentiality, integrity and availability of web applications;
- Understand and be able to implement techniques to address requirements for the authentication, access control and auditing of web applications;
- Understand and implement techniques for testing web applications for security issues;
- Design secure web application architectures; understand the security controls which can be implemented;

- Understand the possible types of application attacks and how to design applications to defend against them;
- Recognize the need for a secure software development life-cycle (SDLC);
- Understand threat modeling and the various web application attack and defense tools available;
- Recognize and implement techniques to deal with web client and server vulnerabilities;
- Explain potential problems with state-based attacks;
- Discuss and design ways to prevent attacks on user-supplied inputs and application outputs;
- Explain cryptography and privacy issues;
- Understand the web application security techniques available in the major development languages and platforms (ASP/.NET, Java, PHP);
- Understand security issues with Web 2.0, NoSQL, HTML 5.0, Oracle and distributed technologies;

Our Trainer

This course was developed and is presented by our Chief Trainer, Mr. Rui Pereira, B.Sc. (Hons), CISSP, CISA, CPTE / CPTC. Rui has over 30 years of experience in the IT industry, the last 20 in Information Security and Audit (with many web application penetration tests and vulnerability assessments under his belt). In addition to WaveFront's suite of IT Security courses, Rui also developed and taught several courses at the University of British Columbia (UBC) and the British Columbia Institute of Technology (BCIT). Rui's biography and resume are online at www.wavefrontcg.com/PrincipalConsultants.html.

Course Outline

Day 1 - **Introduction, Attacks and Defenses**

Introduction
- Background (Web Attacks in the Real World)
- What is Web Application Security? Why is it important?
- The Problem is...
- The Cost of Insecure Software
- Privacy, Legislation and Compliance
- Web application security standards and guidelines
- PCI DSS and Payment Application (PABP) Requirements
- The OWASP Top Ten and CWE/SANS Top 25
- Training and Certifications
- Threats, Vulnerabilities, Risk, Attacks, Exploits
- Network/system vs. web application security
- Attack demonstration - network / system attacks vs. web application attacks

Web Application Attacks and Defenses Pt. I
- SQL Injection (incl. verbose and blind SQLi)
- Cross-Site Scripting (XSS) (incl. XST, CSRF, CRLF Injection, HTTP Response Splitting and iFrame Injection)
- Cross-Site Request Forgery (CSRF)
- Logic and Control Errors
- Weak Authentication and User Management
- Access Control & Session Management (incl. Broken Access Control, Parameter Tampering, Directory Traversal, Session Hijacking and Session Fixation)
- Cookie Manipulation (incl. Insecure Cookies and Cookie Scoping)
- Open Re-direction
- Known Attacks (incl. Buffer Overflows and Website Defacement)

Client-side Issues (incl. Client-side Validation and Comments, Password Auto-complete, Cached SSL Pages and Hidden Fields)
Insecure Configuration
Information Disclosure / Leakage (incl. Insecure Data Storage, Forceful Browsing, Directory Browsing, Verbose Error Messages, Source Code & IP Address Disclosure, Verbose Headers & Banners, E-Mail Disclosure, Sensitive Information in URL's, Username Enumeration and Google Hacking)

Day 2 – Web Application Attacks & Defenses Pt. II, Secure Application Architecture, Authentication and Access Control, Cryptography, User Management, Data Validation, Client Side Security

Web Application Attacks & Defenses Pt. II
- XML & Web Services (incl. XML External Entity Processing and Xpath Injection)
- Web Crawling / Mirroring
- Administrative Interfaces
- File Uploading (incl. Poison Null Byte)
- Framework and third-Party Components (incl. Bugs in Frameworks, Libraries, Shopping Carts, CMS Software, and Sample Applications)
- Dangerous HTTP Commands
- Denial of Service (DoS)
- Other Injection Attacks (incl. LDAP Injection, Command Injection, and Eval Injection)
- Other Attacks (incl. Serialization / De-serialization, Click-jacking, MIME Sniffing, HTTP Parameter Pollution, Cross-Domain Script Include, Race Conditions, Anti-automation, etc.)

Secure Application Architecture
- Exposure and Risk
- Security Architectures and Multi-tier Systems
- Forward and Reverse Proxies
- A Sample Web Application Architecture

Authentication
- Fundamental web application security services
- Authentication, Authorization, Accounting (AAA Services)
- Methods of authentication
- Session logout mechanisms
- Centralized authentication and Single Sign-On (SSO)
- Authentication for Web Services
- Identification and impersonation on multi-tier systems
- Distributed Authentication Technologies (OpenID, JASIG AS)
- Web authentication attacks

Authorization / Access Control
- Authorization Models
- Authorization in Multi-tier Systems
- Role Based Access Control (RBAC)
- Distributed Authorization Technologies (OAuth)
- Authorization for Web Services
- Web Authorization Attacks

Cryptography for Web Applications
- Uses of cryptography
- Cryptographic building blocks
- Symmetric and Asymmetric Cryptography
- Message Digests and Integrity
- Hybrid Cryptography
- Digital Certificates and Certificate Authorities
- Cryptography for network traffic and data at rest
- Key Management and Cryptographic attacks

User Management
- Password and Account Management
- Resetting Passwords
- Password Cracking
- Secure user management systems
- Single Sign-On (SSO) and Identity Management
- User Management Considerations

Data Validation
- Data Validation and Sanitization
- Data Validation Models and Layers
- Trust Boundaries
- Data Validation Attacks

Client Side Security
- Client attacks
- Securing the client platform
- Code obfuscation

Day 3 - Error Handling, Logging, Database Security, Secure SDLC, Session Management, Security Testing, Threat Modeling, Attack & Defense Tools

Error Handling and Exception Management
- Designing for Failure / Failing Securely
- Designing error messages

Logging and Auditing
- Logs and Monitoring
- Uses of Logs
- Logging Processes and Problems
- Logging best practices

Database Access and Security
- Connection strings
- Database access for web applications
- Standard and Integrated Security (MS) and Trusted Connections (Oracle)
- Stored Procedures
- Database Security & Access Controls
- NoSQL security

Sundry Web Application Security Issues
- Security and the SDLC (incl. application migration)
- Session Management
- Software security testing (incl. source code review and pen-testing)
- Threat Modeling
- Attack and Defense Tools

Day 4 - Secure Web Application Coding in Microsoft .NET, Java, PHP, Web 2.0, Oracle, etc.

Microsoft .Net
- Cryptographic Support
- Declarative vs. Programmatic security
- Using authentication protocols
- Implementing Authorization (RBAC)
- Impersonation and Delegation in .Net
- Trusted Sub-systems
- Built-in Validators
- Session Management and Integrity
- Model-View-Controller (MVC) in .Net
- XSS and SQL Injection in .Net
- .Net Services

Java
- The Java Security Model
- Security in the Java Language
- Cryptographic Support
- Authentication and Access Control (incl. RBAC)
- Java Web Services
- Application Architecture using Model-View-Controller (MVC)
- Java Frameworks
- Built-in Validators in JAVA, incl. the Apache Struts framework
- Session Management and Integrity
- Database Access and Logging Facilities in Java

Security for Other Development Platforms
- Secure application coding in PHP, AJAX (Web 2.0), Ruby, HTML 5.0, Oracle, etc.
- Secure Coding Principles and Best Practices

Code Review / Walkthrough's

Contact Rui Pereira at ruiper@wavefrontcg.com or 1 (604) 961-0701 to book a place on the next presentation, or for further information. We also provide one-day seminars and one-two hour presentations based on this material, and can customize the material for in-house presentation.

This document is also available online at www.wavefrontcg.com/IntroWebAppSec_2016.pdf.

Please enquire about our other training offerings, including "Ethical Hacking, Penetration Testing and Vulnerability Assessments" and "Wireless Network Security Complete".