# **H9SPW: Secure Programming for Web**

Module Code:		H9SPW					
Long Title		Secure Programming for Web CONDITIONAL APPROVAL					
Title		Secure Programming for Web					
Module Level:		LEVEL 9					
EQF Level:		7					
EHEA Level:		Second Cycle					
Credits:							
Module Coordinator:		EL BRADFORD					
Module Author:		MICHAEL BRADFORD					
Departments:		nool of Computing					
Specifications of the qualifications and experience required of staff							
Learning Ou	itcomes						
On successfi	ul completion of this modu	ıle the learner will be able to:					
#	Learning Outcome	Description					
LO1	Analyse, compare, c	contrast and critically evaluate common vulnerabilities of web applications with a view to identifying counter-measures to prevent such being exploited.					
LO2	Critically assess the	technological challenges associated with securing web applications from a programming perspective.					
LO3	Evaluate, develop ar associated tools.	o and implement programming solutions for securing web applications using relevant programming techniques, programming languages, and					
LO4	Appraise the tools as applications.	and techniques used to attack web applications and assess the usage of such tools and techniques to strengthen the security of web					
Dependenci	es						
Module Rec	ommendations						
No recommendations listed							
Co-requisite Modules							
No Co-requisite modules listed							
Entry requirements							

# **H9SPW: Secure Programming for Web**

### **Module Content & Assessment**

## **Indicative Content**

# **Browser Security Model**

· Same-origin policy · Cookies security model · Plugin security model (e.g., Flash, ActiveX, Java) · Content Security Policy · Security-related HTTP headers · HTTP security extensions (e.g., HSTS)

### **HTML5 Security**

• HTML5 Security introduction • Web Messaging • Cross Origin Resource Sharing (CORS) • Web Sockets • Server-Sent Events • Local Storage • Client-side Databases • Web Workers · Sandboxed Frames

## Secure JavaScript

· Insecure JavaScript methods · JavaScript output encoding · Security provisions within common JavaScript frameworks

Application Vulnerabilities & Defences

• XSS (Cross Site Scripting) • CSRF (Cross Site Request Forgery) • XSS, CRSF prevention • SQL and NoSQL injection • SQL and NoSQL injection prevention • Other injection attacks (e.g., OS command injection, CRLF injection) • XML security and parsing vulnerabilities • Clickjacking • Clickjacking prevention

## **Authentication and Authorization**

· Authentication and authorization vulnerabilities · Password reset/change vulnerabilities · CAPTCHA vulnerabilities · Integration of single sign-on (SSO) into web applications (e.g., OAuth and SAML) • Authorization approaches

# Session Management

· Secure session lifecycle · Session hijacking · Session fixation · Cookie security

## Input Validation

Client side vs server side input validation
 Open redirects
 File upload validation
 Data type validation

# Server-Side Application Security

Security as a cross-cutting concern • Defence in depth • Security design patterns • Web server vulnerabilities • Interpreted vs. compiled application code • Code analysis, code coverage and code review • Exception handling and application logging • Security provisions within common server-side web application frameworks

• Process / Methodology followed towards successful attack • Common tools used to hack web applications • Automating customised attacks • Hacking for application security test purposes

Assessment Breakdown	%		
Coursework	100.00%		

### Assessments

# **Full Time**

Coursework Assessment Type: Continuous Assessment

% of total: 60 Outcome addressed: **Assessment Date:** 1,2,3

Non-Marked: No

Assessment Description:

Practical work will be conducted throughout the semester to assess the learner's evaluation skills in terms of secure design strategies and secure application development.

Project 40 Assessment Type: % of total: 1,3,4 **Assessment Date:** n/a Outcome addressed:

Non-Marked: No

# **Assessment Description:**

Learners are required to complete a project where they incorporate a number of web programming vulnerabilities into an application (inclusive of non-obvious and complex vulnerabilities) and proceed to fix those security vulnerabilities. Learners must compile an associated report and evaluate the security strength of the resulting application.

No End of Module Assessment

No Workplace Assessment

# Reassessment Requirement

# Coursework Only

This module is reassessed solely on the basis of re-submitted coursework. There is no repeat written examination

# **H9SPW: Secure Programming for Web**

Module Workload								
Module Target Workload Hours 0 Hours								
Workload: Full Time								
Workload Type	Workload Description		Hours	Frequency	Average Weekly Learner Workload			
Lecture	No Description		2	Every Week	2.00			
Tutorial	No Description		3	Every Week	3.00			
Independent Learning	No Description		7.5	Every Week	7.50			
Total Weekly Contact Hours								

# Module Resources

## Recommended Book Resources

- J. P. Mueller. (2015), Security for Web Developers: Using JavaScript, HTML, and CSS, O'Reilly Media.
- D. Stuttard, M. Pinto. (2011), The Web Application Hackers Handbook: Finding a security Flaws, 2nd. Wiley.
- M. Zalewski. (2011), The Tangled Web: A Guide to Securing Modern Web Applications, No Starch Press.

Jonathan LeBlanc, Tim Messerschmidt. (2016), Identity and Data Security for Web Development: Best Practices, 1st. O' Reilly Media, p.204, [ISBN: 1491937017].

## Supplementary Book Resources

- J. Manico. (2014), Iron-Clad Java: Building Secure Web Applications, 1st. McGraw-Hill Education.
- D. Chell, T. Erasmus, S. Colley, O. Whitehouse. (2015), The Mobile Application Hacker's Handbook, Wiley.

This module does not have any article/paper resources

# Other Resources

[website], OWASP, https://www.owasp.org

[website], Web for Pentester, https://pentesterlab.com/exercises/web\_f or\_pentester

[website], Web for Pentester 2,

https://pentesterlab.com/exercises/web\_f or\_pentester\_II

[website], Burp Suite, https://portswigger.net/burp/

Discussion Note: