COURSE SPECIFICATION DOCUMENT

Academic Department: Science, Innovation & Technology

Programme: Computer Science

***FHEQ* Level:** 6

Course Title: Advance Secure Programming

Course Code: COMP 6101

Student Engagement Hours: 160

Timetabled Hours: 45
Guided Learning Hours: 15
Independent Learning Hours: 100

Credits: 16 UK CATS credits
8 ECTS credits
4 US credits

Course Description:

This module delves into the advanced principles of secure programming, focusing on identifying and addressing programming errors that lead to system vulnerabilities. Students will explore various secure coding techniques and best practices across several programming languages (e.g., C, C++, python). The course emphasizes balancing security with performance, usability, and other software quality attributes, enabling students to create robust, secure code. Key topics include string security, dynamic memory management, concurrency, and file I/O security, along with mitigation strategies for common vulnerabilities such as buffer overflows and race conditions.

Prerequisites:

70 credits, COMP 5102 Cyber Security

Aims and Objectives:

The module aims to equip students with advanced knowledge and skills to identify and address programming vulnerabilities that can lead to system exploitation. It emphasizes secure coding techniques across various programming languages and encourages the adoption of secure programming as an integral part of the software development lifecycle. Students will learn to evaluate and balance trade-offs between security, performance, and usability, ensuring the development of robust and secure applications. Additionally, the course provides a critical comparison of different programming languages and their execution environments, fostering a deeper understanding of their respective security capabilities and risks.

Programme Outcomes:

L6: AI, II, BI, III, CII, DIII

A detailed list of the programme outcomes are found in the Programme Specification. This is located at the archive maintained by Registry and found at:
https://www.richmond.ac.uk/programme-and-course-specifications/

Learning Outcomes:

By the end of this course, successful students should be able to:

- Identify and analyse programming errors that can lead to system vulnerabilities.
- Develop and apply secure coding techniques and mitigation strategies to reduce or eliminate the risk of exploitation.
- Integrate secure programming practices throughout the software development lifecycle.
- Compare and contrast different programming languages (e.g., C, C++, Python) regarding their security features and vulnerabilities.

Indicative Content:

- Introduction to Secure Programming
- Systems Programming and Debugging
- String Security
- Pointer Subterfuge
- Dynamic Memory Management
- Concurrency
- File I/O
- Kernel Security

Assessment:

This course conforms to the University Assessment Norms approved at Academic Board and located at: https://www.richmond.ac.uk/university-policies/

Teaching Methodology:

This course will be delivered face to face through a combination of lectures and interactive sessions. In addition to classroom activities, there are guided learning elements that are tutor led and arranged through Blackboard. These activities can be asynchronous online sessions, flipped classrooms, set readings with discussion boards or set guest lectures for example. Set activities are monitored by the instructor to ascertain student engagement. Students are encouraged to prepare for class and to play an active part, to raise questions, following-up ideas and interact with a wide range of provided material.

Indicative Text(s):

Core Texts:

Seacord, R.C. (2020) *Secure Coding in C and C++*. 3rd edn. Boston: Addison-Wesley.

Seacord, R.C. (2024) *Effective C: An Introduction to Professional C Programming*. 2nd edn. California: No Starch Press.

Supplementary Reading:

Dowd, M., McDonald, J. and Schuh, J. (2006) *The Art of Software Security Assessment: Identifying and Preventing Software Vulnerabilities*. Boston: Addison-Wesley Professional.

Additional Text:

Johnsson, D., Deogun, D. and Sawano, D. (2019) *Secure By Design Manning Publications.* Manning Publications.

See syllabus for complete reading list.

Change Log for this CSD:

Nature of Change	Date Approved & Approval Body (School or AB)	Change Actioned by Registry Services
First Edition	Nov 2024	