CS4101: SOFTWARE SECURITY

Effective Term
Semester A 2025/26

Part I Course Overview

Course Title
Software Security

Subject Code
CS - Computer Science
Course Number
4101

Academic Unit
Computer Science (CS)

College/School
College of Computing (CC)

Course Duration
One Semester

Credit Units
3

Level
B1, B2, B3, B4 - Bachelor's Degree

Medium of Instruction
English

Medium of Assessment
English

Prerequisites
CS3103 - Operating Systems

Precursors
Nil

Equivalent Courses
Nil

Exclusive Courses
Nil

Part II Course Details

Abstract
This course aims to provide students with a solid understanding of a range of topics in the field of software security principles, practices, and techniques, emphasizing the identification of security threats to real-world software systems

and the appropriate countermeasures. Students are exposed to basic programming constructs (such as variables, control structures, data structures, programming syntax) as well as the basic principles of object-oriented programming languages. Upon completing the course, students will learn about common vulnerabilities, threat models, and secure coding practices. The course covers both theoretical concepts and practical applications, preparing students to design and develop secure software systems. Meanwhile, the students can also learn to specify and evaluate appropriate security measures for computer systems and software applications.

Course Intended Learning Outcomes (CILOs)

	CILOs	Weighting (if app.)	DEC-A1	DEC-A2	DEC-A3
1	Identify and analyse common threats and vulnerabilities of software, network, and system applications.	25	x	x	
2	Suggest and study major countermeasures to software, network, and system attacks.	25	x	x	
3	Explain and enquire security issues in emerging computing technology and applications.	20	x	x	x
4	Implement and evaluate countermeasures techniques to detect and localize vulnerability in real-world software.	30	x	x	x

A1: Attitude
Develop an attitude of discovery/innovation/creativity, as demonstrated by students possessing a strong sense of curiosity, asking questions actively, challenging assumptions or engaging in inquiry together with teachers.

A2: Ability
Develop the ability/skill needed to discover/innovate/create, as demonstrated by students possessing critical thinking skills to assess ideas, acquiring research skills, synthesizing knowledge across disciplines or applying academic knowledge to real-life problems.

A3: Accomplishments
Demonstrate accomplishment of discovery/innovation/creativity through producing /constructing creative works/new artefacts, effective solutions to real-life problems or new processes.

Learning and Teaching Activities (LTAs)

	LTAs	Brief Description	CILO No.	Hours/week (if applicable)
1	Lectures	Lectures introduce different types of vulnerabilities in diverse software systems, such as kernels, compilers, and web applications. Lectures demonstrate the defence principles, techniques, and technologies used for these attacks. Lectures additionally discuss selected timely security issues in emerging computing technology.	1, 2, 3, 4	3 hours/ week

2	Tutorials	Tutorials are conducted in the laboratory in the form of discussion, demonstration, and hands-on sessions. Students will work with selected security issues and attacking tools, which provide students with hands-on experience in using and configuring the tools and analyzing how the security and attacking tools work. With these exercises, students will learn how the adversary makes use of the tool to attack software and web applications. Students should be able to identify and analyze potential threats to computer systems in organizations and formulate solutions for how organizations may defend themselves.	1, 2, 3, 4	8 hours/ semester
3	Projects	Students will be asked to conduct a substantial case study or in-depth survey on selected security topics, such as thoroughly analysing the security properties of crypto techniques in state-of-the-art system kernels, network protocols, advanced access control, passwords and related usages, memory safety issues and defences, web tracking, command injection attacks and defences, cloud security, etc.	1, 2, 3, 4	2 hours/ week for 4 weeks

Assessment Tasks / Activities (ATs)

	ATs	CILO No.	Weighting (%)	Remarks ("-" for nil entry)	Allow Use of GenAI?
1	Assignment 1	1, 2	10	-	Yes
2	Assignment 2	2, 3	10	-	Yes
3	Assignment 3	3, 4	10	-	Yes
4	Project	1, 2, 3, 4	20	-	Yes

Continuous Assessment (%)

50

Examination (%)
50

Examination Duration (Hours)
2

Minimum Examination Passing Requirement (%)
30

Additional Information for ATs
For a student to pass the course, at least 30% of the maximum mark of the examination must be obtained.

Assessment Rubrics (AR)

Assessment Task
Assignments including problem set and hands-on exercises

Criterion
Ability to answer fundamental software security attacks and defences.

Excellent (A+, A, A-)
High

Good (B+, B, B-)
Significant

Fair (C+, C, C-)
Moderate

Marginal (D)
Basic

Failure (F)
Not even reaching marginal levels

Assessment Task
Project

Criterion
Ability to conduct a substantial case study in vulnerability exploitation and in-depth survey on security topics.

Excellent (A+, A, A-)
High

Good (B+, B, B-)
Significant

Fair (C+, C, C-)
Moderate

Marginal (D)
Basic

Failure (F)
Not even reaching marginal levels

Assessment Task
Examination

Criterion
The exam will include questions to assess the student's ability to explain the principles, techniques and technologies used for software security and its applications in practice.

Excellent (A+, A, A-)
High

Good (B+, B, B-)
Significant

Fair (C+, C, C-)
Moderate

Marginal (D)
Basic

Failure (F)
Not even reaching marginal levels

Part III Other Information

Keyword Syllabus
The syllabus evolves over time as security topics change. The following is a list of potential course topics:
1) Software security principle and threat model.
2) Secure software development lifecycle.
3) Common software vulnerabilities.
4) Static and dynamic analysis.
5) Secure coding practices.
6) Other emerging topics in computer security: AI-related security, cloud security, information governance, information privacy, security evaluation, legal and ethical issues, computer crime and computer forensics, new access control paradigms, mobile security, and database security.

Reading List

Compulsory Readings

	Title
1	Nil

Additional Readings

	Title
1	Gary McGraw. Software Security - Building Security
2	M. Goodrich and R. Tamassia. Introduction to Computer Security. Pearson. (2014)
3	W. Stallings and L. Brown. Computer Security: Principles and Practice. (2015)

4	Shah S. Web 2.0 security: Defending Ajax, RIA, and SOA. Thomson (2008)
5	Spitzner L. Honeypot: Tracking hackers. Addison-Wesley (2003)
6	Whittaker and Thompson. How to break software security. Addison Wesley (2004)
7	Bace R. G. Intrusion Detection. Macmillan Technical (2000)
8	Skoudis and Liston, Counter Hack Reloaded (2e). Prentice Hall (2006)
9	Jon Erickson. Hacking: The Art of Exploitation, 2nd Edition. No Starch Press, 2008
10	Andrews and Whittaker. How to break web software. Addison Wesley (2006)