QF01/0408-4.0E	Course Plan for Bachelor program - Study Plan Development and Updating Procedures/ Cyber Security Department

Study plan No.	**2024/2025**		**University Specialization**	**Cybersecurity**	
Course No.	**0133416**		**Course name**	**Operating Systems' Security**	
Credit Hours	**3**		**Prerequisite Co-requisite**	**0133111**	
Course type	☐ MANDATORY UNIVERSITY REQUIREMENT	☐ UNIVERSITY ELECTIVE REQUIREMENTS	☐ FACULTY MANDATORY REQUIREMENT	☐ Support course family requirements	☐ ✓ **Mandatory requirements** ☐ Elective requirements
Teaching style	☐ **Full online learning**		☐ Blended learning		☐ ✓ Traditional learning
Teaching model	☐ 2Synchronous: 1asynchronous		☐ 2 face to face : 1synchronous		☐ ✓ **3 Traditional**

Faculty member and study divisions information (to be filled in each semester by the subject instructor)

Name	Academic rank	Office No.	Phone No.	E-mail	
Division number	**Time**	**Place**	**Number of students**	**Teaching style**	**Approved model**

Brief description

This course covers both the fundamentals and advanced topics in operating system security. Memory protection and inter-process communications mechanisms will be studied. Students will learn the current state-of-the-art OS-level mechanisms and policies designed to help protect systems against sophisticated attacks. Besides, advanced persistent threats, including rootkits and malware, as well as various protection mechanisms designed to thwart these types of malicious activities, will be studied. Students will learn both hardware and software mechanisms designed to protect the O.S. The course will use virtual machines to study traditional O.S. environments on modern 64-bit systems (e.g., Windows, Linux, and macOS), as well as modern mobile operating systems (e.g., iOS and Android).

Learning resources

Course book information (Title, author, date of issue, publisher ... etc)	Operating System Security, Trent Jaeger, Morgan & Claypool Publishers, 2020
Supportive learning resources (Books, databases, periodicals, software, applications, others)	https://www.cs.columbia.edu/~smb/classes/s06-4118/l25.pdf
Supporting websites	

QF01/0408-4.0E	Course Plan for Bachelor program - Study Plan Development and Updating Procedures/ Cyber Security Department

The physical environment for teaching	☐ ✓**Class room**	☐ labs	☐ **Virtual educational platform**	☐ Others
Necessary equipment and software	**Data show, Computer**			
Supporting people with special needs				
For technical support	**E-learning and Open Educational Center. Computer Center**			

Course learning outcomes (S= *Skills, C= Competences K= Knowledge,)*

No.	Course learning outcomes	The associated program learning output code
	Knowledge	
K1	Understand the fundamentals and advanced topics in operating system security	**PLO1**
K2	Identify the current state-of-the-art OS-level mechanisms designed to help protect systems against sophisticated attacks.	**PLO3**
	Skills	
S1	Use virtual machines to study traditional O.S. environments on modern 64-bit systems (e.g., Windows, Linux, and macOS), as well as modern mobile operating systems (e.g., iOS and Android).	**PLO8**
S2	Securing file systems, including permissions, encryption, and auditing.	**PLO7**
S3	Develop, implement, and manage security policies and procedures to protect operating systems.	**PLO8**
	Competences	
C1	Independently manage tasks related to O.S hardening	**PLO11**
C2	Make constructive decisions in situations that require self-reliance	**PLO12**

Mechanisms for direct evaluation of learning outcomes

Type of assessment / learning style	Fully electronic learning	Blended learning	Traditional Learning (Theory Learning)	Traditional Learning (Practical Learning)
First exam	0	0	**0**	0
Second / midterm exam	%30	%30	**%30**	%30
Participation / practical applications	0	0	**0**	0
Asynchronous interactive activities	%30	%30	**%30**	%30
final exam	%40	%40	**%40**	%40

Note: Asynchronous interactive activities are activities, tasks, projects, assignments, research, studies, projects, work within student groups ... etc, which the student carries out on his own, through the virtual platform without a direct encounter with the subject teacher.

Schedule of simultaneous / face-to-face encounters and their topics

Week	Subject	learning style*	Reference **
1	Introduction to OS Security Part 1, Introduction to OS Security Part 2	Lectures	Lecture Notes
2	Introduction to OS Security Part 3, Introduction to OS Security Part 4	Lectures	Lecture Notes
3	Access Control Fundamental Part 1, Access Control Fundamental Part 2	Lectures	Lecture Notes
4	Access Control Fundamental Part 3, Access Control Fundamental Part 4	Lectures	Lecture Notes
5	Multics Part 1, Multics Part 2	Lectures	Lecture Notes
6	Multics Part 3, Multics Part 4	Lectures	Lecture Notes
7	Security in Operating System Part 1, Security in Operating System Part 2	Lectures	Lecture Notes
Midterm Exam (30%)			
9	Security in Operating System Part 3, Security in Operating System Part 4	Lectures	Lecture Notes
10	Variable Security Goals Part 1, Variable Security Goals Part 2	Lectures	Lecture Notes
11	Variable Security Goals Part 3, Variable Security Goals Part 4	Lectures	Lecture Notes
12	Security Kernels Part 1, Security Kernels Part 2	Lectures	Lecture Notes
13	Security Kernels Part 3, Security Kernels Part 4	Lectures	Lecture Notes
14	Securing Commercial Operating System Part 1	Lectures	Lecture Notes
15	Securing Commercial Operating System Part 2	Lectures	Lecture Notes
Final Exam (40%)			

*** Learning styles: Lecture, flipped learning, learning through projects, learning through problem solving, participatory learning ... etc.**

**** Reference: Pages in a book, database, recorded lecture, content on the e-learning platform, video, website ... etc.**

Project 1: Design and Analysis of a Secure Micro-Kernel Architecture

Task / Activity	Reference	Expected Results
Design a security kernel for a lightweight	Liedtke (1995) on	A documented kernel design

Task / Activity	Reference	Expected Results
OS. Define its reference monitor, security policy (MAC/DAC), and isolation mechanisms. Implement a proof-of-concept in a sandbox (e.g., OS simulator) and analyze its effectiveness against memory corruption attacks.	microkernels, Saltzer & Schroeder design principles, Common Criteria documentation	blueprint with security policy specifications and a test report showing successful isolation o compromised components.

Project 2: OS Security Hardening Framework with Variable Security Profiles

Task / Activity	Reference	Expected Results
Create an automated hardening tool that configures a Linux OS (e.g., Ubuntu) for different security goals: "Web Server," "Workstation," "DMZ Host." Implement configuration scripts enforcing CIS benchmarks, kernel parameter tuning, and mandatory access control (AppArmor/SELinux) profiles.	CIS Benchmarks, NIST SP 800-123, SELinux/AppArmor documentation	Three distinct, automated hardening scripts with verification checklists and a comparison report analyzing the trade-offs between security, usability, and performance for each profile.

Schedule of asynchronous interactive activities (in the case of e-learning and blended learning)

Week	Task / activity	Reference	Expected results
1			
2			
3			
4			
5			
6			
7			
8			