SYLLABUS

1. Data about the program of study

1.1 Institution	The Technical University of Cluj-Napoca
1.2 Faculty	Faculty of Automation and Computer Science
1.3 Department	Computer Science
1.4 Field of study	Computer Science and Information Technology
1.5 Cycle of study	Masters
1.6 Program of study / Qualification	Cybersecurity Engineering / Masters
1.7 Form of education	Full time

2. Data about the subject

2.1 Subject name	***Windows Internals and Kernel Driver Development***			Subject code	**6**
2.2 Course responsible / lecturer	Eng. PhD. Portase Radu-Marian - radu.portase@cs.utcluj.ro				
2.3 Teachers in charge of seminars / laboratory / project	Eng. PhD. Portase Radu-Marian - radu.portase@cs.utcluj.ro				
2.4 Year of study	I	2.5 Semester	2	2.6 Type of assessment (E - exam, C - colloquium, V – verification)	E
2.7 Subject category	Formative category: DF – fundamental, DS – speciality, DC – complementary				DF
	Optionality: DOB – mandatory, DOP – optional (alternative), DFA – optional (free choice)				DOB

3. Estimated total time

3.1 Number of hours per week	4	of which:	Course	1	Seminars	-	Laboratory	3	Project	-
3.2 Number of hours per semester	56	of which:	Course	14	Seminars	-	Laboratory	42	Project	-
3.3 Individual study:										
(a) Manual, lecture material and notes, bibliography										12
(b) Supplementary study in the library, online and in the field										12
(c) Preparation for seminars/laboratory works, homework, reports, portfolios, essays										43
(d) Tutoring										-
(e) Exams and tests										2
(f) Other activities:										-

3.4 Total hours of individual study (suma (3.3(a)…3.3(f)))	69
3.5 Total hours per semester (3.2+3.4)	125
3.6 Number of credit points	5

4. Pre-requisites (where appropriate)

4.1 Curriculum	Operating systems
4.2 Competence	Computer architecture, knowledge about operating systems concepts and internals, programming in C and x86 or amd64 assembler, basic understanding of computer networks and protocols

5. Requirements (where appropriate)

5.1. For the course	Blackboard, Video Projector
5.2. For the applications	Blackboard, Video Projector, Laptop with internet access

6. Specific competence

6.1 Professional competences	• develop information security strategy • perform ICT security testing • execute ICT audits • manage system security • manage IT security compliances • identify ICT security risks • perform risk analysis • provide ICT consulting advice • perform data analysis • ensure compliance with legal requirements • ensure information privacy • monitor developments in field of expertise • keep up with the latest information systems solutions
6.2 Cross competences	• develop an analytical approach • taking a proactive approach • developing strategies to solve problems • being open minded

7. Expected Learning Outcomes

Knowledge	operating systems; security engineering; software anomalies; ICT infrastructure; cyber security; attack vectors; ICT network security risks; cyber attack counter-measures; ICT encryption; ICT security standards; risk management; information security strategy; ICT safety technologies; cloud security and compliance
Skills	manage system security; implement ICT recovery system; implement ICT security policies; develop contingency plans for emergencies; identify ICT security risks; perform ICT security testing; perform data analysis; provide ICT consulting advice; ensure compliance with legal requirements; manage IT security compliances
Responsibilities and autonomy	develop an analytical approach; developing strategies to solve problems; taking a proactive approach; being open minded; coordinate engineering teams

8. Discipline objective (as results from the *key competences gained*)

8.1 General objective	Understand Windows internals and how Windows drivers may be created. Gain practical experience with various Windows kernel technologies. Understand how Windows kernel drivers may be used for security.
8.2 Specific objectives	1. Understand Windows architecture and various executive subsystems 2. Know about the various types of kernel drivers that can be built 3. Learn how to build and debug kernel drivers 4. Learn how user mode applications interact with kernel drivers 5. Understand the security implications of Windows kernel drivers 6. Learn how kernel drivers can be used to increase security of an information system

9. Contents

9.1 Lectures	Hours	Teaching methods	Notes
Windows architecture. Using the kernel debugger to learn about various Windows structures and to debug the kernel	1		

Kernel programming fundamentals	2	Blackboard illustrations and explanations, beamer presentations, discussions, short challenges.	
Windows I/O subsystem (I/O Manager, IRP processing). Memory management	2		
Monitoring application behavior with Windows Drivers	2		
Unscheduled execution (Interrupts, DPC queue and APCs)	1		
Creating device drivers created. Building drivers with KMDF and UMDF	1		
Programming drivers for USB devices	1		
Windows network filters. The Windows Filtering Platform	1		
Security of kernel modules. Specific vulnerabilities and exploitation techniques.	1		
Analysis of other Winows Internal Structures	1		
Concepts review	1		

Bibliography

1. Windows Kernel Programming (Yosifovich, Pavel – 2019 - CreateSpace Independent Publishing Platform)
2. Windows Internals (Russinovich, Mark – 2012 – Microsoft Press) (6th ed)
3. Windows NT File System Internals (Nagar, Rajeev – 2006 – OSR Reprint)
4. Windows Driver Kit (WDK) (Microsoft – 2010-2014 – electronic)
5. Windows Operating System Internals Curriculum Resource Kit (CRK) (Microsoft – 2006 – electronic)
6. Windows Research Kernel 1.2 (Microsoft – 2006 – electronic)

9.2 Applications - Seminars/Laboratory/Project	Hours	Teaching methods	Notes
Learning about the development environment and debugging tools.	3	Brief reviews, blackboard illustrations and explanations, tutorials, roadmaps, short live demos and guidance of code development, discussions, homework.	
Development of a NT legacy driver and a console application that is connected to it.	9		
Development of an anti-malware oriented driver. Understand the basic structure and architecture for a minisystem driver.	3		
Antimalware Driver: Interception of Windows filesystem activity	3		
Antimalware Driver:Interception of registry activity	3		
Antimalware Driver:Interception of process, threads, modules and object activity	6		
Antimalware Driver: Interception of network activity	6		
Device drivers created with KMDF. Keyboard interception Concepts review	3		

Bibliography

1. Windows Internals (Russinovich, Mark – 2012 – Microsoft Press) (6th ed)
2. Windows NT File System Internals (Nagar, Rajeev – 2006 – OSR Reprint)
3. Windows Driver Kit (WDK) (Microsoft – 2010-2014 – electronic)
4. Windows Operating System Internals Curriculum Resource Kit (CRK) (Microsoft – 2006 – electronic)
5. Windows Research Kernel 1.2 (Microsoft – 2006 – electronic)

9. Bridging course contents with the expectations of the representatives of the community, professional associations and employers in the field

Course was designed together with security profesionals from companies that have relevant activity in the field (e.g. Bitdefender) and is alligned with subjects evaluated for various pentesting certifications.

Related courses from other universities:

- *ECE 446 – Device Driver Development*, George Mason University, Fairfax, USA http://catalog.gmu.edu/preview_course_nopop.php?catoid=19&coid=226124
- *COP 5641 – Linux Kernel & Device Driver Programming,* Florida State University, USA http://www.cs.uni.edu/~diesburg/courses/dd/syllabus.html
- Part of the details from the lecture sylabus are also part of various operating systems courses.

10. Evaluation

Activity type	Assessment criteria	Assessment methods	Weight in the final grade
Course	Ability to solve domain-specific problems Presence, (inter)activity during class hours	Written exam, including online quiz tests (e.g. on Moodle platform) and presentation(s) of different subjects / paper in the course's field during semester time. *(summative assessment)*	40%
Laboratory	Ability to solve domain-specific problems Presence, (inter)activity during class hours	We will evaluate the ability to create various security oriented drivers by reviewing code submissions. *(continuous assessment)* Evaluate solutions of problems given in a final seminar exam. *(summative assessment)*	60%

Minimum standard of performance

***Lecture*.** Attending **minimum 50%** of lecture classes, to be allowed to take the final examination.
Minimum grade for exam: 5
Desired competences: Understand Windows Internals at an intermediate level. Know how to develop Windows kernel drivers.
***Lab*.** Attending **all lab classes** (one lab could be recovered during the semester, and one more during re-examination sessions).
Minimum grade on all laboratory submissions: 5
Desired competences: Demonstrate ability write kernel level code. Understand basic security of kernel drivers. Understand architectures for anti-malware drivers.

Date of filling in: 29.04.2026	**Responsible**	**Title First name Last name**	**Signature**
	Course	Eng. PhD. Radu-Marian PORTASE	
	Applications	Eng. PhD. Radu-Marian PORTASE	

Date of approval in the department	Head of department, Prof.dr.eng. Rodica Potolea
Date of approval in the Faculty Council	Dean, Prof.dr.eng. Vlad Mureșan