COURSE DESCRIPTION CARD - SYLLABUS

Course name
Malware Analysis and Detection [S1Cybez1>AZO]

Course

Field of study
Cybersecurity

Year/Semester
3/5

Area of study (specialization)
–

Profile of study
general academic

Level of study
first-cycle

Course offered in
Polish

Form of study
full-time

Requirements
compulsory

Number of hours

Lecture
16

Laboratory classes
30

Other
0

Tutorials
0

Projects/seminars
30

Number of credit points

5,00

Coordinators

dr inż. Marek Michalski
marek.michalski@put.poznan.pl

dr hab. inż. Mariusz Żal
mariusz.zal@put.poznan.pl

Lecturers

Prerequisites

• Basic knowledge of low-level programming languages (e.g., C, Assembly). • Ability to use Linux and Windows operating systems at the administrative level. • Fundamental knowledge of computer networks and communication protocols. • Understanding of the basics of cryptography and information security.

Course objective

The aim of this course is to familiarize students with methods of malware analysis, including identification, decompilation, and both functional and behavioral analysis. Students will acquire the ability to recognize threats, characterize them, and counter their effects.

Course-related learning outcomes

Knowledge:
• Knows the basic types of malware and their characteristics. [K1_W06]
• Understands the processes of static and dynamic analysis. [K1_W06]

• Is familiar with the tools used in malware analysis. [K1_W09]

Skills:
• Can conduct both static and dynamic analysis of malware samples. [K1_U03]
• Is able to use relevant tools to identify the functionality of malicious software. [K1_U02]

Social competences:
• Appreciates the role of prevention in IT security. [K1_K05]
• Is ready to independently expand knowledge of emerging threats in the rapidly evolving field of cybersecurity.[K1_K01]

Methods for verifying learning outcomes and assessment criteria

Learning outcomes presented above are verified as follows:

• Final test in written or oral form. Includes multiple-choice and open-ended questions (requiring descriptive answers).
• Laboratory project and assessment of laboratory exercises, evaluated during the execution of the exercises.
In each form of the course assessment, the grade depends on the number of points the student earns relative to the maximum number of required points. Earning at least 50% of the possible points is a prerequisite for passing. The relationship between the grade and the number of points is defined by the Study Regulations. Additionally, the course completion rules and the exact passing thresholds will be communicated to students at the beginning of the semester through the university's electronic systems and during the first class meeting (in each form of classes).

Programme content

This course introduces students to the issues related to malware analysis. It covers basic static and dynamic analysis techniques and the tools used to identify, investigate, and counter the threats posed by malware.

Course topics

1. Introduction to Malware Analysis: Basic concepts, types of malicious software.
2. Static Analysis: Examining binary files, reading metadata, and code analysis.
3. Dynamic Analysis: Running malware in a controlled environment and monitoring its behavior.
4. Reverse Engineering: Tools for decompilation and code analysis (e.g., IDA Pro, Ghidra).
5. Analysis Environments: Creating sandboxes and secure testing environments.
6. Malware Countermeasures: Techniques for protecting against and removing malicious software.
7. Case Studies: Analysis of historical and current threats.
Laboratory sessions will focus on the practical aspects of the topics covered in lectures, including disassembly, static and dynamic analysis, memory analysis, the use of malware analysis tools, and the setup of secure environments.

Teaching methods

• Theoretical lectures illustrated with case studies.
• Laboratory sessions using malware analysis tools.

Bibliography

Basic:
• Michael Sikorski, Andrew Honig, Practical Malware Analysis, No Starch Press, 2012.
• Monnappa K A, Learning Malware Analysis, Packt Publishing, 2018.
• Skoudis, E., Zeltser, L. "Malware: Fighting Malicious Code", Pearson, 2003.

Additional:
• Documentation of analytical tools, for example Ghidra, Cuckoo Sandbox, Wireshark.
• National Institute of Standards and Technology (NIST)

Breakdown of average student's workload

	Hours	ECTS
Total workload	136	5,00
Classes requiring direct contact with the teacher	76	3,00
Student's own work (literature studies, preparation for laboratory classes/ tutorials, preparation for tests/exam, project preparation)	60	2,00