Universidad Zaragoza

Year : 2020/21

62240 - Exploiting Software Vulnerabilities

Syllabus Information

Academic Year: 2020/21
Subject: 62240 - Exploiting Software Vulnerabilities
Faculty / School: 110 - Escuela de Ingeniería y Arquitectura
Degree: 534 - Master's in Computer Engineering
ECTS: 3.0
Year: 1
Semester: First semester
Subject Type: Optional
Module: ---

1.General information

1.1.Aims of the course

1.2.Context and importance of this course in the degree

1.3.Recommendations to take this course

2.Learning goals

2.1.Competences

2.2.Learning goals

2.3.Importance of learning goals

3.Assessment (1st and 2nd call)

3.1.Assessment tasks (description of tasks, marking system and assessment criteria)

4.Methodology, learning tasks, syllabus and resources

4.1.Methodological overview

The methodology followed in this course is oriented towards achievement of the learning objectives. A wide range ofteaching and learning tasks are implemented, such as:

- Lectures. The instructor presents and explains the class contents, including illustrative examples.
- Laboratory sessions. Activities with specialized equipment (in the laboratory, computer room).
- Oral presentations. Preassigned problems will be presented on the classroom
- Assignments. Preparation of seminars, readings, small research projects, documents to be presented on the classroom or handed in to the teacher.

4.2.Learning tasks

The course (75 hours) includes the following learning tasks:

- 30 hours, approximately, of classroom activities: lectures, laboratory sessions, and problem-solving tasks.
- 25 hours, approximately, of assignments and research projects.
- 5 hours, approximately, of tutorials.
- 10 hours, approximately, of autonomous work and study.
- 5 hours, approximately, of the exam and defense of the course project.

4.3.Syllabus

- Introduction: vulnerability management, types of vulnerabilities, tools and analysis lab
- Program binary analysis: static analysis, dynamic analysis
- Software vulnerabilities and exploitation techniques: memory errors, integers, format strings, concurrency issues; ROP attacks, shellcode design
- Software defenses
- Malware analysis: methodology, writing and reading technical reports

4.4.Course planning and calendar

The teaching planning of this course is organized as follows:

- Lectures and problem-solving tasks
- Laboratory sessions

The exact hours of lectures and laboratory sessions will be announced beforehand in the Center's and course's websites.

Further details concerning the timetable, classroom, office hours, assessment dates and other details regarding this course, will be provided on the first day of class and announced beforehand in the Center's and course's websites.

4.5.Bibliography and recommended resources

- Reverse Engineering for Beginners, Dennis Yurichev (free book)
- Practical Reverse Engineering: X86, X64, ARM, Windows Kernel, Reversing Tools, and Obfuscation. Bruce Dang, Alexandre Gazet, Elias Bachaalany. John Wiley & Sons, Feb 17, 2014
- Reversing: Secrets of Reverse Engineering. Eldad Eilam. John Wiley & Sons, Dec 12, 2011
- A Bug Hunter's Diary. T. Klein, No Starch Press, 2011
- Hacking, 2nd Edition: The Art of Exploitation. Jon Erickson. No Starch Press, 2008
- The Shellcoder's Handbook: Discovering and Exploiting Security Holes, Jack Koziol, Wiley, Apr 2, 2004
- Gray Hat Python: Python Programming for Hackers and Reverse Engineers, Justin Seitz, No Starch Press, 2009
- Writing Security Tools and Exploits. James C. Foster, Vincent Liu. Syngress, 2006
- Buffer Overflow Attacks: Detect, Exploit, Prevent. Jason Deckard. Syngress, Jan 29, 2005