STUDY GUIDE

Software Security **(2DMI20)**

2023-2024

Version 10th November 2023

Lecturer(s):
dr. S. Schäge

1. Introduction

Software is ubiquitous in our modern world. However, developing software is far from trivial and highly prone to errors on both a technical and a conceptual level. Unfortunately, many of these errors could allow attackers to sabotage the system that runs the software. This puts digital systems at risk and media coverage of successful attacks have shown devastating effects on system owners and users, and sometimes on a society in general.

The goal of the course Software Security is to show how errors in software can ultimately lead to a variety of attacks and how to use methods (countermeasures) to avoid them.

2. Information about the lectures

A detailed schedule of the lecture is available on Canvas.

3. Position in curriculum

3.1 Class planning: Day, time, place

All scheduled elements of the course take place in Quarter 2 of the academic year 2022-2023, on Tuesday mornings in 10:45-12:45 and Friday afternoons in Hours 15:30-17:30. Sessions are supposed to take place on campus.

3.2 Relation with other subjects

In the IST mastertrack, the course serves as an important foundation for later courses in Q3 like Advanced Network Security or the Intrusion Detection Laboratory. Essentially, in these courses, one often assumes that attackers have gained access to some computing device (node corruption). In practice, this can happen because of vulnerable software which in turn is the consequence of insufficient care for software security.

3.3 Relevance of the subject for the domain

Software Security is, besides cryptography, an important pillar of modern IT-Security and thus an indispensable part of our modern society.

3.4 Admission criteria

This course can be taken as a core course (IST master), an elective (e.g. in the in the embedded software stream), and, of course, as a free elective. The admission criteria vary in these cases and students should consult their study program documentation.

4. Learning objectives and content

4.1 Learning Objectives

Students familiarize themselves with basic knowledge of theories, methods and techniques of software security.
At the end of the course, students should be able to:

- recall and understand important weaknesses and countermeasures in detail.
- determine appropriate countermeasures to complex weaknesses.
- analyse software development and usage scenarios, categorize important weaknesses, and apply appropriate countermeasures .
- prioritize weaknesses and compare and justify efforts of countermeasures .
- create original solutions to new weaknesses.

Students should also be familiar with some key concepts and applications of either descriptive decision theory or information security.
Students should be able to determine, and reason about, the applicability of these concepts to real-world problems arising from stakeholders.

4.2 Content

- Introduction to Software, (In)Secure Software, Software Lifecycle, and the 7+1 Kingdoms
- Introduction to Software Development
- Memory Corruption Attacks
- Static Code Analysis
- Dynamic Code Analysis/Fuzzing
- Safe Languages
- Introduction to the Web
- Web Security
- Side-Channel Attacks and Encapsulation
- Race Condition Vulnerabilities
- API Abuse

5.Organization

5.1 Type of education

The course consists of 13 lectures.

5.2 Class planning

All material will be tested on a final exam containing questions from the entire lecture.

Moreover, students can gain bonus points up to 10% by completing challenges. In addition to that students will be required to work through additional reading material. To deepen the material further, quizzes will be made available to students as well as additional interactive toy examples.

5.3 Distribution of study load

Total study load (5 ECTS): 140 hours

6. Type of examination

6.1 Type of examination

The written exam at the end of the term counts for 100% of the final grade. It is possible to gain bonus points (up to 10%) for the written exam by completing quizzes.

FINAL_POINTS_EXAM= POINTS_WRITTEN_EXAM + 10% * MAXNUMPOINTS_EXAM * PROPORTION_IN_CHALLENGES

To pass the course, a student needs to:

score 5.5 or higher for the written exam **excluding** bonus points.

Unless permission is explicitly granted, grades for an assignment or exam achieved in prior academic years cannot be carried over to this academic year.

The exam is a written exam. No books or other texts are permitted during the exam. The exam will be held on-campus if the pandemic situation allows, and held online otherwise.

Date and time of the exam: Wednesday, 31st January 2023, 13:30-16:30.

The location will be announced later. Don’t forget to register in advance.

6.2 Course material

Textbooks:
The Cyber Security Body of Knowledge
(https://www.cybok.org/media/downloads/CyBOK_v1.1.0).

Gary McGraw: Software Security: Building Security In, 2006

Robert C. Seacord: Secure Coding In C and C+, 2013

Michael Howard, David LeBlanc, John Viega: 24 Deadly Sins of Software Security, 2010

6.3 Planning of examination

The examination will be a normal on-campus (paper) examination unless the administration announces that this is not possible.

6.4 Date feedback and/or inspection

Exam grades will be available within 15 working days of exam completion

6.5 Definition of final grade

See 6.1

6.6 Distribution exam questions on course material

The final exam will cover the reading and material for the entirety of the course.

7. Anti-plagiarism

When you submit your work under your own name you are asserting ownership of that work. When using ideas of another person, you must give that person appropriate credit through referencing. Referencing serves multiple purposes: (i) it allows readers to further explore sources you have consulted, (ii) it shows the depth of your own thinking and process of inquiry, (iii) it allows you and your readers to compare and contrast your position with other people's positions, agreeing with some, disagreeing with others, and (iv) it gives proper credit to the hard work that many people have done before you. Submitted work will be screened for plagiarism using software licensed by the university for this purpose.