SE 14-448: Information and Cyber Security
Department of Software Engineering
Achi Racov Faculty of Engineering
Kinneret College on the Sea of Galilee

Instructor: Michael J. May

Semester 2 of 5786

1 Course Details

The course meets at **8:00am-11:00am** on Wednesday (**3** hours of lecture and recitation combined) in Room 1006.

2 Prerequisites

The official prerequisite for the this course is SE 14-331: Introduction to Computer Networks. Given that the course is in the last semester of the degree program, a proficiency in programming in Java and a working knowledge of probability are presumed.

3 Overview

The course covers the following topics from both theoretical and practical perspectives:

Application Security	Buffer Overflows
Encryption and Randomness	Hash Functions
Authentication	Public/Private Key Pairs
Shared Secrets	Digital Certificates
Internet and Web Security	Transport Layer Security
SQL Injection	Symmetric Encryption

The course focuses on developing secure applications for mobile and web. We'll study the foundations of how software breaks and tools that can prevent them from doing so. We'll also study security tools such as encryption, hashing, and digital signatures that make it possible for apps to communicate securely and verify information.

The main books for the course are *Computer and Internet Security* by Du [3] and *A Graduate Course in Applied Cryptography* by Dan Boneh and Victor Shoup [2].

3.1 Course Goals

At the end of the course, students shall be able to do the following:

1. Give proper definitions for the terms “authenticated”, “encrypted”, “trustworthy”, “secret”, “complete” and explain the differences between them.
2. Properly use the following security atoms in application code: cryptographic hash functions; AES and similar symmetric ciphers; RSA asymmetric cipher; RSA digital signatures; Diffie-Hellman key establishment; and Needham-Schroeder key distribution.
3. Explain the operation of X.509 certificates and use them in Java programs.
4. Design a system that securely uses and stores passwords for authentication.
5. Perform types of buffer overflow attacks in C and defend against them.
6. Perform SQL injection attacks and defend against them.

4 Lecture Schedule

The course lectures are structured in the following way. The relevant chapters from the Du (Du) and Boneh & Shoup (BS) books are listed in the indicated column. Material not covered well in the books may be supplemented from papers or other sources as shown in the O column.

#	Subject	Du	BS	O
1	Requirements Cryptographic analysis One time pads, computational security			
2	Computational Security Stream and Block Cipher Functions Advanced Block Ciphers	24	 3.1–3.3 4.1,9.1–7	
3	Symmetric Cipher Modes	24	4.1,9.1–7	
4	Hashes and Merkle Trees	25	8.1,8.6-8	
5	Merkle Trees, Diffie-Hellman		10.3–4	[4]
6	Public/Private Key Pairs, RSA Quantum and Post-Quantum Crypto	26	10.3–4	[4] [1]
7	Passwords and Human Authentication	25.4	18.3–4	
8	Digital Signatures Key Exchange & Establishment	26	13.1–2 13.1–2	
9	PKI, Digital Certificates	27		
10	Transport Layer Security		21.10,9.8	
11	Software Security: Buffer Flow and Variable attacks	4		
12	Software Security: Advanced Buffer Overflow	5,6		
13	Secure coding techniques for SQL and injection	14		

Since this is an advanced course, students **are expected to come to class having read the material listed above in the lecture schedule**. Students who do not come prepared will find themselves at a significant disadvantage.

5 Assignments and Labs

There will be 4 programming assignments during the course of the semester. More details of the assignments will be distributed during the course of the semester.

There will also be 3–4 hands on security attack labs during the last three weeks of the semester. More details on the labs will be given during the course of the semester.

6 Recitation and Laboratory Work

There is no dedicated laboratory or recitation session for the course. Instead, lecture sessions will include a mixture of lecture and hands on experimentation.

7 Attendance

Students are responsible for all material presented in class, recitation, and laboratory sessions, all assigned readings, and all material provided for additional reading out of class. Students who miss a lecture or targil can look at the course syllabus and web page to see which material was missed.

Attendance of lectures and targil sessions is expected and required for this course. As per College policy, a student who misses 20% or more of the lectures of targil sessions may be expelled from the course. Students who miss lectures do so at their own risk and expense and will be expected to make up missed material on their own. Students who know they will be missing two or more lectures due to circumstances beyond their control should inform the instructor as soon as possible beforehand.

Attendance will not be taken directly during class, however, there will be a weekly quiz at the beginning of class beginning on week 2. There will not be an opportunity to make up missed quizzes.

8 Submissions

8.1 How to Submit Work

All work must be submitted work via private per-assignment GitHub repositories managed by the instructor. Materials sent via email, uploaded to unrelated GitHub repositories, submitted via any other method risk being ignored or ungraded without consideration of their merits.

If work is submitted by a team of students, every student on the team must make at least one significant code commit to the GitHub repository. If a student's name appears on a submission, but the student doesn't perform at least one significant code commit to the assignment repository, the student **will not** receive a grade for the assignment.

8.2 Assignment and Lab Late Submission Policy

Students are expected to be on time with their assignment submissions. Each assignment must be turned in by the date it is due.

Each student will be given 36 slip hours to use for late submissions. The slip hours can be used on a single assignment or divided up among several. Slip hours are rounded up per assignment (*i.e.* 70 minutes late = 2 slip hours, 3 minutes late = 1 slip hour). Once the slip hours are finished, a student's submissions will no longer be accepted.

36 hours after the due date of any assignment, no submissions will be accepted.

Students who are called up to Miluim duty will have their assignment deadlines extended in accordance with college policy.

9 Cheating

Cheating of any sort will not be tolerated. Student collaboration is encouraged, but within limits as set forth in the college's rules on academic integrity. Any students caught cheating will be immediately referred to the department head and the Dean and may receive a failing grade for the course.

Cheating includes:

- Copying information, content, or verbatim text from other students, internet sites, books (other than the ones listed in the bibliography), other unaffiliated individuals to answer questions, solve problems, or aid in programming projects.
- Copying or submitting source code, documentation, or other programming aids **without attribution** from other students, **web sites**, online repositories, text books, open source programs, or other unaffiliated individuals.
- Project teams which submit work which is identical or substantially identical to work submitted by other project teams, whether current or from previous years.
- Other forms of academic misconduct as described on the site: `https://catalog.upenn.edu/pennbook/code-of-academic-integrity/` or as reasonably assessed by the instructor, program head, or dean.

If you have any questions about what constitutes cheating in the above rules, contact the instructor as early as possible.

10 Exams

There will be a final exam at the end of the semester that covers all material in course.

11 Grading

Final grades will be calculated by combining grades from assignments and the exam. The grades are weighted as follows:

40%	Assignments and Labs (Required)
60%	Final Exam

The instructor will not address questions about specific individual grades during the lecture or review sessions. Students may contact the instructor *in person* during office hours or at the instructor's convenience.

12 Books

The following books are used for the class: Du [3], Boneh and Shoup [2].

The library has copies of the books listed, but students are encouraged, to purchase the books as needed. Some of the above books are available freely online as indicated in the bibliography.

13 Contact Information

Instructor: Michael J. May
Website: https://mjmay-kinneret.github.io/

References

[1] Daniel J. Bernstein and Tanja Lange. Post-quantum cryptography. *Nature*, 549(7671):188–194, 2017.

[2] Dan Boneh and Victor Shoup. *A Graduate Course in Applied Cryptography.* Online, 0.6 edition, Jan 2023. https://toc.cryptobook.us/book.pdf.

[3] Wenliang Du. *Computer & Internet Security: A Hands-on Approach.* Wenliang Du, 3rd ed. edition, May 2022.

[4] Arjen K. Lenstra, James P. Hughes, Maxime Augier, Joppe W. Bos, Thorsten Kleinjung, and Christophe Wachter. Ron was wrong, Whit is right. Cryptology ePrint Archive, Report 2012/064, 2012. https://eprint.iacr.org/2012/064.