COSC/MATH 314 Cryptography

**FALL 2017**

**Syllabus**

**Instructor:**  Marius Zimand

**Room:** YR466

**Email:**  [mzimand@towson.edu](mailto:mzimand@towson.edu) (preferred way to reach me)

**Phone**: 410 704 4967

**Class Hours:** Tu, Thurs. 5:30 – 6:45 pm

**Room:** YR 202

**Office Hours:** Tu 6:45 – 7:45 pm and Wed. 9:30-10:30 am. Appointments outside these time slots can be arranged by email.

**Web page**:

The class will have a web page accessible from <http://orion.towson.edu/~mzimand>. Visit this page often for assignments, grades, notes, last-minute announcements.

**Text Book:** *Wade Trappe and Lawrence Washington, Introduction to Cryptography with Coding Theory,* Prentice Hall, second edition.

**Course Objective:**  The course provides a broad overview of the mathematical basis of modern cryptography and presents the main cryptosystems currently in use. Students are exposed to relevant chapters of number theory and computational number theory (modular arithmetic, finite fields, primality testing, quadratic residues, discrete logarithms, and others) at the undergraduate level. The course covers the most important cryptosystems (DES, AES, RSA) and the basic tools used in building security mechanisms (one-way functions, hash functions, message authentication codes, pseudo-random generators, bit commitment, hash functions, etc.). Some basic principles of cryptanalysis are presented as well. At the end of the course, students will have a good understanding of the theoretical foundations of cryptography and of the basic techniques for achieving different cryptographic services.

**Grading Policy:** Grading will be based on two tests, assignments, quizzes, and one project.

**A**: 93-100, **A-:** 90- 92.99

**B+**: 87-89.99, **B**: 83-86.99, **B-**: 80-82.99

**C+**: 75-79.99, **C**: 70-74.99

**D+**: 67-69.99, **D**: 63-66.99, **D-:** 60-62.99

**F**: <60.

**Evaluation**

If the average on the 2 exams is < 60, the grade is F. If the average is >= 60, then the grade is calculated as follows:

Midterm and Final: 30% + 35%

Assignments and Quizzes: 20%

Project : 15%

**Notes:**

1. Roll will be taken each class meeting. Class attendance is expected. If you miss a class, you are responsible for the material presented in the lecture and for obtaining information about assignments. No makeup test/quiz will be given.
2. Laptops are permitted only if you use them to take notes. Cell phones must be closed.
3. Modern cryptography relies on mathematics and computational complexity. The mathematical concepts will be covered from the basics, however the class requires some mathematical maturity.
4. Most assignments will be group assignments (to make the grading process feasible). You should form teams of 2-3 students. Each assignment will be submitted by the team (so it will be one single submission per team) and will include the name of the team members. To do well on quizzes and exams it is important that each team member participates actively in solving the assignment questions. Ideally each student will prepare her/his answers alone and then the team meets (in person or on the internet) to discuss individual answers and prepare the form that is submitted.
5. The assignments and the project are to be handed in on the due date. Each day of delay will be penalized with 15%, so write down the date you have turned in the assignment.
6. The assignments should be carefully done and typed (with the possible exception of long mathematical formulas and diagrams). Sloppiness will cost you lost points.
7. The quizzes will not be announced. They contain exercises from the last assignment and easy questions based on the material covered in the previous four lectures (however, since new concepts are built on top of older concepts, you should not count anything out, even if it was presented earlier). They are meant to guarantee that the student has actively participated in doing the assignment and that she/he is current with the last concepts explained in class.
8. Your project will be of one of the following two types: (1) a programming project in which you implement some crypto protocol, and (2) an analytical project which has a more theoretical flavor and does not involve programming. Each student will pick the type of project that he/she finds more suitable. The programming will be done in Java, C, C++ or another programming language that we can agree upon. I will provide more details about projects, including project topics soon. The project write-up should be typed nicely. Oral presentations should be carefully prepared. Sloppiness can cost you points.
9. Copying and cheating will be harshly reprimanded (the minimum penalty is zero for the assignment, quiz, or test). See the Student Handbook for the Academic Integrity Policy. See also the document on plagiarism on the web site.
10. The Student Handbook specifies that students may not repeat a course more than once without prior permission of the Academics Standards Committee.
11. The final exam is scheduled by the University for Dec. 14, 5:15 – 7:15 pm. The midterm exam will be scheduled by us (probably in the 7th or 8th week). The exam dates will not be changed and there will be no make up tests. Exceptions will be made only for emergencies that are properly documented.

**Prerequisite:**

COSC 236 and (MATH 263 or MATH 267) and junior’s standing or instructor’s permission.

**Topics (timing is tentative);**

1. Basic concepts of cryptology (2 weeks)

* Historical ciphers (Caesar, Vigenere, rotor machines, etc.)
* Cryptanalysis of historical ciphers.
* One-time pads

1. Modern Symmetric Cryptographic Systems (4 weeks)

* DES
* Differential Cryptanalysis
* Triple DES
* Modes of Operation (ECB, CBC, CFB, OFB)
* Stream ciphers – basic notions and RC4
* Advanced Encryption Standard - Rijndael

1. Basic Number Theory (2 weeks)

* Euclidean Algorithm
* Modular Arithmetic
* Chinese Remainder Theorem
* Fermat’s Little Theorem and Euler’s Theorem
* Primitive Roots
* Square Roots Mod n
* Finite Fields

1. Public Key Cryptography (3 weeks)

* RSA
* Attacks on RSA
* Factoring and Primality Testing
* Discrete Logarithms
* ElGamal Public Key Cryptosystem

1. Data integrity and authentication. (2 weeks)

* Digital signature schemes (RSA Signatures and ElGamal Signature Scheme)
* Hash Functions
* Security of Hash Functions and MACs

1. Basic cryptographic primitives and applications (1 week)

* One-way functions
* Pseudo-random generators
* Secret Sharing Schemes (if time allows)
* Bit Commitment Schemes (if time allows)
* 2-party and multy party protocols for private distributed computation (if time allows)

1. Zero-Knowledge Techniques (if time allows)

* Basic Schemes
* Feige-Fiat-Shamir Identification Scheme.