KTH Datavetenskap
och kommunikation

DD2448 Foundations of Cryptography 7.5 hp
Spring 2015

Important Source of Information

All information about the course is posted on the course homepage, https://www.kth.se/social/course/DD2448/.

Goal

The goal of the course is to

- give an overview of modern cryptography

in order that students should

- know how to evaluate and, to some extent, create cryptographic constructions, and
- to be able to read and to extract useful information from research papers in cryptography.

Prerequisites

DD1352 Algorithms, data structures and complexity (or *DD2354 Algorithms and complexity* for older students). We also assume knowledge of mathematics and theory of algorithms corresponding to the required courses of the D or F-programmes at KTH.

Lecturer

Douglas Wikström is responsible for the course and he gives most lectures. The safest way to reach him is by email at `dog@csc.kth.se` (please put `Krypto15` in the subject), but he can mostly be found in his office, Room 1518, Lindstedtsvägen 3 (5th floor in the E-building).

Tentative Plan of Content

- Administration, introduction, classical cryptography.
- Symmetric ciphers, substitution-permutation networks, linear cryptanalysis, differential cryptanalysis.
- AES, Feistel networks, Luby-Rackoff, DES, modes of operations, DES-variants.
- Entropy and perfect secrecy.
- Repetition of elementary number theory: groups, fields, and rings.
- Public-key cryptography, RSA, primality testing, textbook RSA, semantic security.
- RSA in ROM, Rabin, discrete logarithms, Diffie-Hellman, El Gamal.
- Security notions of hash functions, random oracles, iterated constructions, SHA, universal hash functions.
- Message authentication codes, identification schemes, signature schemes, PKI.
- Elliptic curve cryptography.
- Pseudorandom generators.
- Guest lecture.
- Make-up time and/or special topic.

Course Material

The main course book is *Stinson: Cryptography, Theory and Practice, Chapman & Hall CRC, 3rd edition*, but this book does not cover all of the material covered in class. Pointers to additional books and other literature are provided on the course homepage. Part of the course requirement is to find the necessary resources to learn more and solve problems. Thus, no reading instructions will be given.

Course Requirements

Know the Rules. All students are expected to have read and understood the *CSC code of honor*, but additional rules apply for this course. Both can be found at the course homepage. All students are required to read and understand the meaning of these rules before starting with any of the tasks below.

Presentations. Give a 12-min oral presentation of a research paper. There is a list of proposed topics to choose from, or you can choose your own, but in the latter case you must make sure that we accept your choice before you start working on your talk. This task gives 0 or 30-80 presentation points (P-points). The approach used to grade talks and instructions are found on the course homepage.

Homework 1-4. Each homework consists of a number of assignments; both theoretical and practical. Solutions may be written in Swedish or English. Each assignment gives a number of *implementation* points (I-points) or *theory* points (T-points). Each homework satisfies $I + T \geq 50$ and $I \geq 10$. Detailed rules for how to solve and submit solutions to the homeworks are found on the course homepage.

Oral Exam. The oral exam is scheduled individually at the end of the course and gives a single oral point (O-point) if it is passed. The purpose of the oral exam is to give a more fair grade.

The starting point of the exam are the solutions to the homeworks submitted by the student and possibly also the oral presentation. A number of (positive or negative) I or T-points may be awarded for individual problems of the homeworks for which written solutions have been submitted, depending on the level of understanding displayed. No more points can be withdrawn (negative points), than was awarded for a solution.

Thus, a moderate amount of remarking may take place, so make sure that you: (1) are ready to explain your solutions in detail at the exam, and (2) exploit this opportunity to improve your grade.

Deadlines. All the deadlines of the course are announced well in advance at the homepage.

Grading

The grade requirements are cumulative, e.g., to earn a C the requirements of the grades E-C must be fulfilled. Define the sum of *all* points by $A = P + I + T + O$. The requirements are as follows:

E. $P \geq 30$, $I \geq 30$, $T \geq 40$, and $O \geq 1$.

D. $A \geq 120$.

C. $P \geq 50$ and $A \geq 140$.

B. $A \geq 170$.

A. $P \geq 60$ and $A \geq 210$.

A good presentation is important!

Kattis

Kattis is a judging server for programming competitions and for grading programming assignments, see https://kth.kattis.scrool.se. We use this for all exercises where you submit code.

By default we assume that your Kattis id is the same as your KTH user name, e.g., if your KTH email is `xyz@kth.se`, then we assume that your Kattis user name is `xyz`. If that is not the case, then please email us your kattis user name using the subject `Krypto15 Kattis`, and don't forget to put your name in there as well.

Please ask a fellow student to give you a brief introduction to Kattis if you have not used it before. If you do not have a Kattis username, then email us and we will provide one for you.

Register for course krypto15 at https://kth.kattis.scrool.se/courses/krypto15 to allow us to see your results.

Project for Motivated Students

A mix-net is a distributed cryptographic program (cryptographic protocol) executed on multiple servers that is used to implement electronic voting systems. It generates a public key used by voters to encrypt their message and later it takes a list of ciphertexts and outputs the corresponding votes in random order.

Some mix-nets output a proof of correctness at the end of the execution that leaks no knowledge of who voted for what. This makes it infeasible to change the result for an attacker even if it has corrupted and controls *all* servers. This seemingly magical property is achieved with so-called zero-knowledge proofs.

The Verificatum Mix-Net was developed at KTH and used in the real electronic municipal elections in Norway 2013. There is a document that gives a precise description of the proof output by this mix-net.

Students can work on their own or in pairs and replace some regular exercises by implementing a verifier. The work can be divided into natural steps, so this is not an all-or-nothing decision. We need to negotiate a fair number of points for each step along the way, since it is hard to predict how hard each step is.

A few students from previous years have successfully implemented more or less complete verifiers. It is a demanding exercise, but also a perfect opportunity to dig deeper into the subject of cryptography than is otherwise possible in any courses at KTH.

In short, if you want to learn as much as possible about cryptography and computer security, then you should do this!