

CAS CS 538: Fundamentals of Cryptography

Fall 2012

1 Administrative Stuff

Official Description

Basic algorithms to guarantee confidentiality and authenticity of data. Definitions and proofs of security for practical constructions. Topics include perfectly secure encryption, pseudorandom generators, RSA and ElGamal encryption, Diffie-Hellman key agreement, RSA signatures, secret sharing, block and stream ciphers.

Prerequisites

CAS CS 113 and 237 or permission of instructor. The course will also require a good comfort level with mathematical proofs and elementary probability theory. If you haven't taken CS 113 or CS 237, please talk to me in the first week of class.

Instructor

Leonid Reyzin, reyzin@cs.bu.edu, (617-35)3-3283, MCS (111 Cummington St) room 135B
Office hours (tentative): Monday 3–5, Thursday 12:30–1:30.

I encourage you to come to my office hours. If you need to talk to me but can't make the office hours, please send me email. I usually check it a few times on a weekday and at least once on a weekend.

Lectures and Notes

The lecture is in MCS (111 Cummington St) room B21, Tuesdays and Thursdays 11–12:30.

I expect you to come to lecture and encourage you to participate. This class is small enough that we can keep it interactive. Lectures are your primary source of information. Although my own lecture notes are available on-line, they will not cover all the material. They cannot substitute for coming to lecture, as they will be dry, condensed and (necessarily) non-interactive. The whole point of getting your education at a bricks-and-mortar place is the ability to interact—take advantage of it.

Textbooks

Introduction to Modern Cryptography, by Jonathan Katz and Yehuda Lindell, covers roughly the right topics at roughly the right level. I will not follow it precisely (I prefer a different order of presentation), and it is not required for this class, but it is well written and contains most of the material that we will cover and then some. The book has a web page <http://www.cs.umd.edu/~jkatz/imc.html>.

This book is closest to our class. There are many other books on cryptography and related subjects that may be of interest. For background on number theory and algebra, you may find Victor Shoup's book A Computational Introduction to Number Theory and Algebra helpful (the entire book is on-line for free at <http://shoup.net/ntb/>). For those interested in studying theoretical cryptography in more depth, I recommend Foundations of Cryptography, volumes I and II, by Oded Goldreich (the book is not on-line, but <http://www.wisdom.weizmann.ac.il/~oded/foc.html> has a lot of information). For those interested in learning more about the applied side of things, I recommend the Handbook of Applied Cryptography by Alfred Menezes, Paul van Oorschot, and Scott Vanstone (the entire book is available for free on-line at <http://www.cacr.math.uwaterloo.ca/hac/>). None of these books is required for this course.

Finally, there are many books on the history (rather than the science) of cryptography and related fields, for those of you who are interested. I enjoyed Crypto by Steven Levy. Other well-known books (which I

have not had a chance to read) include The Codebreakers: The Story of Secret Writing by David Kahn and Code Book: The Science of Secrecy from Ancient Egypt to Quantum Cryptography by Simon Singh.

Other Communication

The class has a home page: <http://www.cs.bu.edu/~reyzin/teaching/f12cs538/>. Homework Q&A and class announcements will be handled via Piazza <https://piazza.com/bu/fall2012/casc538/home>. Please sign up with Piazza ASAP with an email address that you actually check, so that you don't miss announcements ("I didn't get your email" is not an acceptable excuse). Piazza is also the place to post HW questions. Answer your fellow students' questions! Piazza has a wiki-like editing interface where you can collectively improve an answer. Instructors can edit answers, too.

The only questions that should go to my email are questions of a personal nature.

Homework, Exams, and Grading

There will be (roughly weekly) problem sets, assigned in class. Late assignments will not ordinarily be accepted. If an assignment is due at the beginning of class, I expect you to hand it in at the *beginning* of class. If, for some compelling reason, you cannot hand in an assignment on time, please contact me as far in advance as possible.

There will be a final exam, scheduled by the registrar for Monday 12/17 12:30–2:30pm. There will be no midterm. The problem sets are worth 70% of the grade. The exam is worth 30%, but you must pass it in order to pass the class. I reserve the right to deviate from this formula in unusual cases.

If you are unsure of your performance in the class, please come and talk to me. Remember that the last day to drop a class without a 'W' is Tuesday, October 9 (to add confusion, that day will have a Monday schedule of classes). The last day to drop a class with a 'W' is Friday, November 9. After that, you must receive a real grade for the course. Please talk to me if you are considering dropping the class—quite often students drop for the wrong reasons.

Collaboration Policy

Collaboration policy for this class is as follows.

- You are encouraged to collaborate with one another in studying the notes and lecture material.
- As long as it satisfies the following conditions, collaboration on the homework assignments is permitted and will not reduce your grade:
 1. Before discussing each homework problem with anyone else, you must give it an honest half-hour of serious thought.
 2. You must write up your solutions completely on your own, without looking at other people's write-ups.
 3. In your solution to each problem, you must write the names of those with whom you discussed it.
 4. You may not consult solution manuals, other people's solutions from similar courses or prior years of this course, etc. You may not work with people outside this class (but come and talk to us if you have a tutor) or get someone else to do it for you.
- You are not permitted to collaborate on the final exam.

The last point is particularly important: if you don't make an honest effort on the homework but always get ideas from others, your exam score will reflect it.

Violations of Collaboration Policy

Violations of collaboration policy fall into two categories: ones that are *acknowledged* in your write-up and ones that are *unacknowledged*.

Acknowledged violations (e.g., reading someone else’s solution before writing your own and saying so in your own solution) will result in an appropriate reduction in the grade, but will not be considered cheating.

Unacknowledged violations of the collaboration policy—for example, not stating the names of your collaborators, or any other attempt to represent the work of another as your own—will result in an automatic failing grade and will be reported to the Academic Conduct Committee (ACC). The ACC often suspends or expels students deemed guilty of plagiarism or other forms of cheating. I have served on the ACC and have seen it happen. I will assume that you understand the CAS/GRS Academic Conduct Code; read it if you haven’t).

If you are uncertain as to whether a particular kind of interaction with someone else constitutes illegal collaboration or academic dishonesty, please ask me *before* taking any action that might violate the rules; if you can’t reach me in time, then at the very least include a clear explanation of what happened in your homework write-up to avoid being treated as a cheater. Citing your sources is usually the easiest way out of trouble.

2 Contents

What this course is about

The primary focus of this course will be on *definitions* and *constructions* of various cryptographic objects, such as pseudorandom generators, encryption schemes, digital signature schemes, message authentication codes, block ciphers, and others time permitting. We will try to understand what security properties are desirable in such objects, how to properly define these properties, and how to design objects that satisfy them.

Once we establish a good definition for a particular object, the emphasis will be on constructing examples that *provably* satisfy the definition. Thus, a main prerequisite of this course is mathematical maturity and a certain comfort level with proofs. I will be doing proofs in class, and you will be doing them on the problem sets.

Secondary topics that we will cover only briefly will be current cryptographic practice and the history of cryptography and cryptanalysis.

At the end of this course, you should be able to make sense of a good portion of current cryptography research papers and standards.

What this course is *not* about

This course will not teach you how to make your computer secure. Cryptography is only one tool in computer security. The rest of computer security has to deal with such fascinating things as buggy code, poorly managed and ever-too-curious humans, power consumption of smart cards, electromagnetic emissions from monitors, etc. We will mostly abstract all that away. I will, however, try to point out where the limitations of our models are and what else is needed for actual security.

This course will also not teach you how to implement the techniques we discuss in the most efficient manner. We will stop at cryptographic algorithms. The underlying number-theoretic algorithms will be discussed only briefly; the most advanced and efficient ones require more time to learn than we will have. For example, if you take only this class, you should be able to program RSA, but many existing implementations will probably be much more efficient than yours.

Finally, this course will not teach you how to design the next great block cipher, such as DES or AES, or the next cryptographic hash function, such as SHA. Nor will this course teach you how to “break” such designs.

Just because I will not teach these topics does not mean they are not worth your while. There are plenty of books and research papers to read and people to talk to if you are interested in pursuing any of these topics.

Topics to be Covered

Below is a rather ambitious plan for the topics I intend to cover, week by week. We will probably have to omit some as we go along. If we have to omit anything, we'll try to omit the extra constructions and keep the definitions: constructions tend to improve all the time, while definitions are here to stay.

Week	Topics
1	Perfect security: Shannon's lowerbound and the Vernam cipher (a.k.a. one-time pad); review of relevant probability theory
2	Pseudorandom generators (a.k.a. stream ciphers): definition, discrete log problem, and Blum-Micali construction
3	Indistinguishability-based definition and composability theorem for pseudorandom generators
4	Integer factorization, Chinese remainder theorem, and Blum-Blum-Shub pseudorandom generator
5	Intuition and first examples of public-key encryption: RSA, Rabin. Definition of security
6	Encrypting long messages with RSA, Blum-Goldwasser and PKCS #1
7	Brief history. Diffie-Hellman key agreement, decisional Diffie-Hellman assumption, and ElGamal encryption
8	Introduction to one-way and trapdoor functions, hardcore bits, Goldreich-Levin construction
9	Signature schemes and hash functions. Merkle trees. Random oracle model. Full-domain-hash RSA and Rabin.
10	Certificates, public-key infrastructure, brief look at simple protocols
11	Symmetric ciphers and message authentication codes. Pseudorandom functions
12	Zero-knowledge proofs
13	Secret sharing and multiparty computation

I will try to provide pointers to more advanced topics as we go along.