

Math 445, Spring 2004
Introduction to Cryptography
Project Suggestions

Standards: Projects may be roughly divided into two types: *study projects* and *implementation projects*. Study projects will involve researching a topic and writing a paper of roughly 10 pages. An “A” study project will involve significant analysis and synthesis, and it will be well-written and professionally presented. Implementation projects will involve implementing intelligently a cryptographic protocol, computationally testing claims made in the literature, or automating the cryptanalysis of an interesting cryptosystem. An “A” implementation project will be a complete package allowing for easy compilation, demonstration, and testing of the algorithms involved, it will perform well on standard hardware, and it will adhere to high standards of coding and documentation.

The line between study and implementation is somewhat artificial and projects may mix aspects of both types. For example, an essay comparing two cryptosystems could be supplemented with code to test the real-world performance of the two systems.

Dates: A short description of your project subject and goals is due on **Friday, April 2**. The project itself is due on **Friday, April 23**. Revisions may be suggested, in which case the final version is due on **Monday, May 3**.

Plagiarism: Plagiarism is the use of someone else’s ideas, words, code, etc., without proper attribution. It is an unforgivable offense in academic settings and in this course it will be dealt with as harshly as is reasonably possible. In particular, anyone caught plagiarizing in his or her project is extremely unlikely to pass the course. If you have the slightest doubt about what is allowable, please see me as soon as possible.

Suggested topics: This list does not give any details and is far from exhaustive. Feel free to ask for more information or to propose another topic that interests you.

1. Survey of applications of cryptography in the real world.
2. Cryptography on the internet. SSL, ssh, https, ..., their strengths and weaknesses.
3. Primality testing and factoring. There are many subtopics here: quadratic and number field sieves (mathematically sophisticated), probabilistic and deterministic tests, the new AKS algorithm, elliptic curve algorithms, ...
4. Finite fields. Theory and computation.
5. A package to do linear algebra modulo N .
6. Social and political aspects of cryptography. One place to start is “Privacy on the Line” by Whitfield Diffie and Susan Landau. See also the review of it in the June 1998 issue of the *Notices of the American Mathematical Society*.
7. Pretty Good Privacy (PGP). There are interesting social, political, and technical aspects here.
8. Brute force attacks on DES.
9. Random numbers: generating them and testing for randomness. An excellent reference is Volume 2 of Knuth’s *Art of Computer Programming*.
10. Elliptic curves in cryptography. There are several subtopics here: elliptic curve primality proving, elliptic curve factorization algorithms, elliptic curve versions of Diffie-Hellman, DSA, El-Gamal, ... These topics are only practical if you have some knowledge of elliptic curves from other courses or independent studies.
11. The NTRU (lattice) cryptosystem. This requires some mathematical sophistication.
12. The Arithmetica (braid group) cryptosystem. This requires even more mathematical sophistication.
13. The Solitaire cryptosystem. This uses a deck of cards and is meant for field use.
14. Timing attacks.
15. Power analysis attacks.
16. The Merkle-Hellman (knapsack) and McEliece (coding theory) cryptosystems.
17. Rivest’s proposal of “Chaffing and Winnowing” for confidentiality without encryption.
18. Hashing.
19. Historical incidents where cryptography played a significant role. (You need to have a thesis here—recounting a couple of interesting incidents will not suffice.)
20. Survey of the AES finalists.

21. Strength or weakness of systems used in common consumer applications. E.g., WEP (Wi-Fi), GSM (a cell phone protocol common throughout the rest of the world), CDMA (another common cell phone protocol).
22. Broadcast one-time pad systems.
23. Cryptography in various resource constrained applications, such as smart cards, ATM machines, cell-phones, polling machines, ...
24. Automated cryptanalysis of substitution ciphers. (This may be pretty hard.)
25. Automated cryptanalysis of Vignère ciphers. This is in principle easier than the previous topic, so standards for success will be higher.
26. Analysis of one (or comparison of several) commonly used symmetric cryptosystem(s).
27. Kerberos.
28. Attacks on PGP: Implement and determine whether this is really practical. See <http://www.schneier.com/paper-chotext.html>.
29. Simulation of Enigma, or possibly a simplified version. A nice interface is required.
30. Pros and cons of stream ciphers vs. block ciphers.
31. Electronic voting.