YALE UNIVERSITY
DEPARTMENT OF COMPUTER SCIENCE

CPSC 467b: Cryptography and Computer Security *Handout #1*
Professor M. J. Fischer *January 11, 2013*

Syllabus (Spring 2013)

1 Official Yale course listing

CPSC 467 01 (22569) /CPSC 567 01 (20784)

Cryptography and Computer Security

Michael Fischer

TTh 1.00–2.15 AKW 000

Final exam scheduled (Group 26)

05/07/2013 T 9.00

Skills QR

A survey of such private and public key cryptographic techniques as DES, RSA, and zero-knowledge proofs, and their application to problems of maintaining privacy and security in computer networks. Focus on technology, with consideration of such societal issues as balancing individual privacy concerns against the needs of law enforcement, vulnerability of societal institutions to electronic attack, export regulations and international competitiveness, and development of secure information systems.
Some programming may be required. After CPSC 202 and 223.

Course Website: http://zoo.cs.yale.edu/classes/cs467/2013s/index.html

2 Course Description

This course is about cryptography and its applications to information and computer security. Privacy and security are central to our emerging "information society", and cryptography is a key technology for achieving them. It is also a fascinating field of study in its own right.

Information security, broadly defined, involves controlling the dissemination of information. It includes issues of privacy, data integrity, authenticity, and authority. Privacy refers to preventing information flow to unintended recipients. Data integrity properties insure that information is correct and undamaged. Authenticity identifies information with a source. Authority describes what actions are permitted by whom. Because of the ease with which information can be copied and transmitted, traditional physical means of control are of limited efficacy. Cryptography gives a way to build logical controls on the flow of information that are largely independent of the physical properties of the devices used to transmit and store information.

Computer security relies on cryptography for access control and protection of sensitive data, but computer security also includes topics such as physical security, access restrictions, activity monitoring, and control of software defects that go way beyond what will be covered in this course.

Cryptography lies at the center of this course, but we will be approaching the subject broadly. On the one end, we'll look at problems of computer and information security and see how cryptographic tools can be used to solve them. We'll also touch on some social issues surrounding the use of cryptography. At the other end, we'll explore the mathematical structures from which cryptographic primitives are built.

Security properties cannot be verified through testing since there is no way to test all possible attacks. Instead, they must be verified analytically through security modeling, or empirically through the test of time. Analytic verification means establishing plausible mathematical models in which security properties can be formally stated and proved.

3 Tentative Schedule

The lectures will generally follow the outline below but are subject to revision as the term progresses. The exam dates are firm, so you should avoid scheduling other commitments on those days.

Lecture 1 (01/15). Course overview. Symmetric cryptography.

Lecture 2 (01/17). Security of symmetric cryptography. Probability theory. Perfect secrecy.

Lecture 3 (01/22). Perfect secrecy cont. Classical cryptography. Block ciphers.

Lecture 4 (01/24). Cryptanalysis. Building and using block ciphers. DES.

Lecture 5 (01/29). AES.

Lecture 6 (01/31). Stream ciphers. Steganography. Active adversaries.

Lecture 7 (02/05). Public-key cryptography. RSA. Computing with big numbers. Number Theory: division, modular arithmetic, fast exponentiation algorithms.

Lecture 8 (02/07). Number Theory cont.

Lecture 9 (02/12). Primality tests. RSA security.

Lecture 10 (02/14). Diffie-Hellman key exchange. ElGamal key agreement.

Lecture 11 (02/19). Message integrity and authenticity. Digital signature algorithms. Security of digital signatures.

Lecture 12 (02/21). DSA. Hash functions: MD5, SHA family. Digital signatures with special properties.

Lecture 13 (02/26). Elliptic curve cryptography.

Midterm Exam [02/28].

Lecture 14 (03/05). Quadratic residues, squares, and square roots. QR probabilistic cryptosystem. The Legendre and Jacobi symbols. Useful tests of compositeness.

Lecture 15 (03/07). Message digest. Cryptographic hash functions. Authentication using passwords.

Lecture 16 (03/26). Authentication while preventing impersonation. Zero knowledge interactive proofs (ZKIP). Public key infrastructure (PKI) and trust.

Lecture 17 (03/28). Formalizing zero knowledge and other kinds of interactive proofs. Non-interactive interactive proofs.

Lecture 18 (04/02). Pseudorandom sequence generation. BBS pseudorandom sequence generator. Bit-prediction.

Lecture 19 (04/04). Secret splitting. Bit commitment problem.

Lecture 20 (04/09). Formalization of bit commitment schemes. Coin-flipping. Oblivious transfer.

Lecture 21 (04/11). Encryption with special properties. Homomorphic encryption.

Lecture 22 (04/16). Oblivious transfer. Privacy-preserving multiparty computation.

Lecture 23 (04/18). Biometrics.

Lecture 24 (04/23). Kerberos, SSH, TLS. Digital rights management and trusted computing platform.

Lecture 25 (04/25). Anonymous communication. DISSENT anonymity protocol.

Final Exam [05/07, 9 am]. See official exam schedule for room (when available).

4 Course materials

Required textbooks:

1. Shafi Goldwasser and Mihir Bellare,
   Lecture Notes on Cryptography. Available online without charge.
2. Wade Trappe and Lawrence C. Washington,
   Introduction to Cryptography with Coding Theory, Second Edition, Pearson, 2006, ISBN-10: 0131862391, ISBN-13: 9780131862395. Suggested retail price: $138.67. Google product search lists a variety of sellers of new and used copies at varying prices.

Website: I maintain a course website at http://zoo.cs.yale.edu/classes/cs467/2013s/index.html. You should bookmark it in your browser and visit it often. It will grow as the term progresses and will contain announcements, handouts, lecture notes, revisions to homework assignments, programming hints, and links to documents in the course directory and elsewhere on the web. *Access to this and other Yale web sites may be restricted to machines on the Yale network.* If you find it is, you will need to configure your browser to use the Yale proxy server, or set up your machine to use a Yale VPN connection.

5 Course Mechanics

Prerequisites: This course will be taught at an advanced undergraduate/graduate level and assumes a basic computer science background. Some C/C++ programming will be required. CPSC 202a and 223b are prerequisites. Graduate students should have an equivalent background.

Requirements: Course requirements include written problem sets and programming assignments (∼30%), a midterm exam (∼25%), and a final exam (∼45%). The approximate weights of each in determining the course grade are subject to change depending on the number and difficulty of the assignments actually given. Graduate students taking the course will be expected to perform at a higher level than undergraduates and may be required to do additional work.

Assignments and other announcements: Written problem sets and programming assignments will posted on the handouts page of the course website from time to time during the course. Other course announcements will be posted on the course home page. It is your responsibility to check these pages frequently.

Help with Technical Material: The teaching assistants will be holding scheduled office hours during the term, which will be posted on the course home page. You are encouraged to meet with them with questions about the lectures, textbook, and problem sets. You may also send questions to the instructional staff by email. Please use the email address cs467help@cs.yale.edu. Your question will go to the entire instructional staff, and whoever is available at the time can decide to answer it. The response will also go to the entire staff so the others will know that it has been taken care of.

Other Questions: All questions about assessment and grading should be taken first to the TA's. If they are unable to resolve your questions to your satisfaction, or if you wish to talk to me privately about any matter, then you are always welcome to contact me, either by email to fischer-michael@cs.yale.edu or in person. Email is also the preferred way to arrange an appointment with me.

6 Policies

Late Policy: Assignments will be due at 11:59 pm on the night of the stated due date. Late work will generally be subject to a penalty of 5% per day late unless accompanied by a Dean's excuse. A 2-hour grace period following the original due date will be granted during which no late penalty will be assessed. However, there will be no grace period in counting the number of dates late for assignments turned in after the grace period. Work more than 4 days late will not be accepted, but alternative means for making up missed work may be arranged on an individual basis with a Dean's excuse.
Please contact the instructor or TA as soon as you know that you will be unable to submit work on time or to attend a scheduled exam so that suitable makeup arrangements can be made.

Policy on Working Together: This course follows the Yale College policy on Cheating, Plagiarism, and Documentation, with which you should familiarize yourself. Briefly, if you use someone else's work, you must acknowledge it. If it's a piece of code, place the acknowledgement in your source file and explain clearly what parts are not your own. Similarly, if it's in a paper, the acknowledgement belongs in the paper itself. All work not so acknowledged must be your own.

You may of course discuss the lectures and readings with your classmates in order to improve your understanding of the subject matter. Helping each other learn to use the tools in the Zoo is also okay. However, the design and implementation of all programs and all written work must be your own except where other sources are explicitly noted.
You are always free (and encouraged) to come in and ask the TA or instructor for help about anything concerning the course. Please talk to the instructor if you have any questions about this policy.

Avoiding Plagiarism: You may neither copy from another student nor permit your own work to be copied, unless explicit permission is given for such collaborations. To avoid unintended involvement in plagiarism, your work should never be in the possession of another student. Do not ask someone else to deliver or pick up your work. Do not let another student "borrow" your code to compare with theirs. Keep your files protected so that others cannot read them and carefully guard your password. Do not leave printed work in public areas such as the Zoo or in accessible wastebaskets. If you think your password may have been compromised, you should change it immediately.

Policy on Computer Problems: The Yale College policy on "Use of Computers and Postponement of Work" in the Yale College Programs of Study applies to this course. It is reproduced below.

> "Problems that may arise from the use of computers, software, and printers normally are not considered legitimate reasons for the postponement of work. A student who uses computers is responsible for operating them properly and completing work on time. (It is expected that a student will exercise reasonable prudence to safeguard materials, including saving data on removable disks at frequent intervals and making duplicate copies of work files.) Any computer work should be completed well in advance of the deadline in order to avoid last-minute technical problems as well as delays caused by heavy demand on shared computer resources in Yale College."

Particularly relevant for this course are the cautions against leaving a programming assignment to the last minute when machines might be busy, printers broken, and so forth, and about safeguarding your data.

7 Computing Facilities

The Zoo: This course will use the Computer Science Department's educational computing facility, affectionately known as the Zoo. This facility contains modern workstations SuSE Linux. You will need to use these machines to prepare and submit coursework. Look at

`http://zoo.cs.yale.edu/help/`

for information on getting started if you are new to the Zoo.
These days, most of you have your own laptops and may be wondering why you should be bothered with using a new computer system. The answer is because code development software is still not completely compatible across multiple platforms. If it works on your

Mac or Windows PC but fails when the graders run it on the Zoo, you will lose points. If you ask for help with compiler errors on your personal machine, we won't be in a position to answer your questions. In short, develop your code on the Zoo! Regardless of where the code is developed, *your assignments must be submitted from your Zoo course account*, and they will be graded according to how well they work on the Zoo.
The Zoo machines support remote access via the SSH and VNC protocols. These enable you to do your work remotely when it is inconvenient to go in person to the Zoo. Instructions on how to configure your machine for remote access will be posted to the course web site.

Your course account: You *must* request a course account for this course *even if you already have a Zoo account*. You will be unable to submit your assignments without it. To obtain your account, go to

```
http://zoo.cs.yale.edu/help/accessing-zoo.shtml
```

and follow the instructions there. Do not wait. Do it now. I will be unsympathetic for late submissions due your not having followed this instruction.

Course directory: The shared course directory, `/c/cs467`, is located on the Zoo server. You can access it from your Zoo course account. It will contain any software needed for this course and miscellaneous documentation and files. It will also contain software to allow you to submit assignments electronically. Public files there can be accessed via the web as well as from a Zoo node. Your class account home directories will also be located there.