

CMPS 122: Computer Security Syllabus

Computer Science Department
University of California, Santa Cruz

Winter 2011

Time: Tue & Thu, 10:00–11:45 AM
Location: Baskin Engineering 165
Instructor: Professor Ethan Miller
Office hours: Tue noon–1 PM, Thu 1:30–2:30 PM in E2-365
Prerequisites: CMPS 111 strongly recommended
Textbook: *Cryptography and Network Security: Principles and Practice, 5th Edition*,
William Stallings (ISBN-10: 0136097049)
Optional texts: *Secrets and Lies*, Schneier
Cryptography Engineering: Design Principles and Practical Applications, Ferguson & Schneier
Security Engineering, 2nd Edition, Anderson
Home page: <http://www.soe.ucsc.edu/classes/cmeps122/Winter11/>

Course objectives

The goal for students in this course is to learn the fundamentals of computer security, including:

- Principles of computer security
- Basic cryptography
- Authentication
- Secure network protocols (Kerberos, SSL)
- Program security
 - Bug exploits
 - Malicious code: viruses, worms, trojan horses, and more
- Attacks and defenses on computer systems
 - Firewalls
 - Intrusion detection
 - Countermeasures
- Trusted operating systems
- Societal issues in computer security: legal, ethical, governmental

Additional topics may be covered, depending on the interests of the students and the professor.

Where possible and appropriate, we will use examples from Linux and other modern operating systems as well as current events to illustrate concepts covered in class.

Prerequisites

The formal prerequisite for this course is CMPS 111 (Introduction to Operating Systems) before this class. Students who have not taken CMPS 111, but have a strong background in understanding computer systems, may be given permission to enroll—please contact the instructor. Students should also be familiar with basic probability and statistics, and knowledge of network protocols, though not required, is also helpful.

Texts

The required text is *Cryptography and Network Security: Principles and Practice*, by Stallings and Brown. If you want *much* more on the theory behind cryptography, please consider *Applied Cryptography*, Bruce Schneier's classic on cryptography or (a bit easier) *Cryptography Engineering: Design Principles and Practical Applications*, by Ferguson and Schneier. I also recommend that you read *Secrets and Lies*, a mostly non-technical book, because it provides an excellent overview of many of the issues in computer security that we'll explore in more depth in class.

Web pages

Most of the information for the class will be distributed via the Internet. We'll be using Moodle (<http://moodle.soe.ucsc.edu/>) for course information, assignments, etc. Most of the class web pages are available only to users taking the class.

Assignments

Homework

There will be 5–6 homework assignments, one every week or two, assigned over the course of the quarter. The assignments may require some programming, which (unless otherwise specified) may be done in any language you want. Homeworks will be graded, and will be returned as soon as possible, usually within 7–10 days. Assignments will be posted on the Moodle site, and will be accessible from anywhere on the Internet once you've logged into Moodle. Due dates for all assignments will be listed on the class schedule, as well as on the assignment itself.

Homework must be turned electronically using the Moodle submission system, and is due on the date and time listed on the assignment. **Late homework will not be accepted (the submission system shuts off at the due date), and will result in a zero for the assignment.** Homework may be submitted from anywhere with an Internet connection; having a minor illness that keeps you home for a few days is *not* a valid reason for an extension. Since unexpected events do crop up, however, each student in the class may turn in *one* assignment during the quarter up to 48 hours late with no penalty and no excuse needed, by emailing it to the instructor.

In addition to homework assignments, there are challenge problems—questions that may or may not be solvable. Solving a challenge problem gets you extra points, depending on how many people work on the solution. Typically, only the first person to solve the challenge problem gets credit, though details may vary depending on the problem. More details on challenge problems are available from the course Web pages.

Term project

There is a required term project on a topic related to computer security. The project may be a programming project or a survey of papers on a particular area of computer security. There will be checkpoints throughout the quarter to ensure that you are making progress on your project. Checkpoints will not be graded separately, though your overall report *will* be graded.

Getting help

You're strongly encouraged to seek help if you need it. You can do this by going to office hours, reading the Moodle forums, or by email. Office hours are optional, but highly recommended if you're having any difficulty understanding the material, doing the homework assignments, or working on the term project. More in-depth discussions of security-related topics are also appropriate (and encouraged) during office hours. You're welcome to use the course forum and send email at any time, but please arrange any meetings outside of office hours in advance.

We'll be using the Moodle forums for online discussions. I strongly encourage you to read the forum and post if you have *general* questions. Asking things like "how does this concept work?" or "what does this algorithm do?" are fine. Questions such as "what's the answer to Problem 3 for this week's homework?" are **not** acceptable. Please ask such questions during office hours (preferable) or via email.

Email to the instructor will be answered if possible, typically within 24 hours (potentially longer on weekends)—if you want short turnaround time, go to office hours. The best kinds of questions to ask via email are those that require short answers. Questions like "why doesn't my program work?" and "please explain this concept to me" are too difficult to answer via email, and are best asked and answered in person at office hours.

Attendance

You're expected to attend every class, though attendance won't be taken, except as needed for UCSC administration. Most of the course material, including assignments and lecture notes, will be posted on the class web pages. However, things may get said in class that aren't in the online notes. ***You're responsible for all material covered in class***, whether or not it appeared on the Web site—I suggest you ask either a fellow student or the professor (in office hours) to fill in any material you may have missed. Emails of the form "I missed today's class—what did we cover?" will get a perfunctory response.

Grades

Your grades will be determined as follows:

- Homework: 35% (all assignments weighted equally)
- Project: 20%
- Exams: 42% (15% midterm, 27% final)
- Class participation: 3%

You must take both exams and turn in a final project to pass the class. You need not turn in every homework, but a missing homework counts as a zero (0). If your homework or exam average is below 50%, you will fail the class regardless of your overall average. Grades will be available online in Moodle during the quarter.

Academic Honesty

This is a class on computer security, so ethics are of the utmost importance. You ***must*** follow these commandments. If you are caught doing any of these things, there will be very serious consequences.

- 1. Thou shalt not plagiarize.**
- 2. Thou shalt not cheat.**
- 3. Thou shalt not attempt to break into computer systems without *written* permission from the owner.**

Unacceptable Behavior

This class is about computer security, and I encourage you to experiment with computing exploits—often, the best way to learn about something is by doing it. However, **you must obtain permission IN WRITING before you experiment on any computer system that you do not personally own. THERE ARE NO EXCEPTIONS TO THIS RULE!** If I find that you have disrupted someone else’s computer or network without their *written* permission, you will immediately fail the class and I *will* refer the case to campus authorities and/or the police. I realize that this may seem excessive, but permission in writing is the only thing that will hold up in court, and experiments on others’ computing systems can easily end up there.

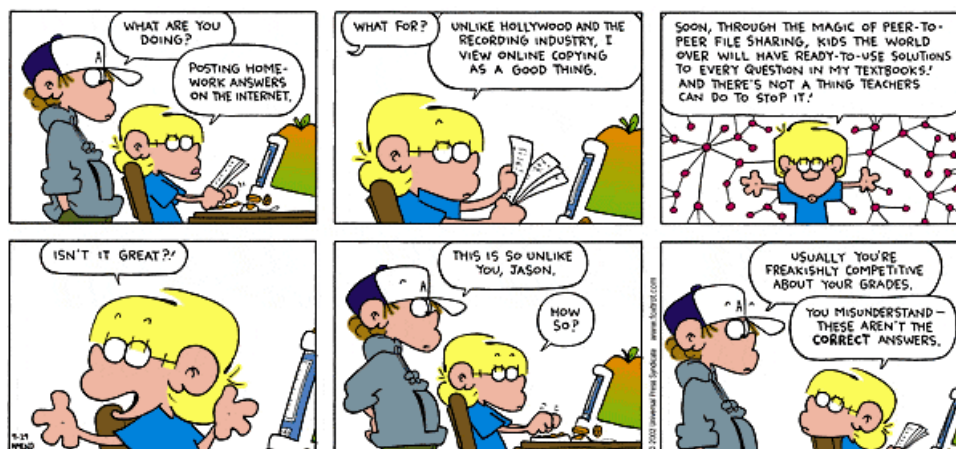
Plagiarism

Plagiarism in any form is completely unacceptable. Plagiarism is defined as “the unauthorized use or close imitation of the language and thoughts of another author and the representation of them as one’s own original work.” [source: dictionary.com] Plagiarism will be assumed, until disproved, on work that is essentially the same as that of other students. This includes identically incorrect, off-the-wall, and highly unusual duplicate answers where the probability of a sheer coincidence is extremely unlikely. All parties to this unacceptable collaboration will receive the same (zero) score. In the case of programs, reordering routines, renaming files, and simply renaming variables does *not* make two programs different. Remember—a zero score on either exam or on the term project is grounds for failing the course. Those caught cheating will, in addition to a zero score on the assignment or exam, have a letter sent to their department, the School of Engineering, and their college provost and academic preceptor. I reserve the right to take stronger action at my discretion, such as assigning a class grade of F, should the situation warrant it.

You may discuss homework with your friends, but you are expected to abide by the *Simpsons* rule—the only thing you may bring to such a discussion is you, and no written notes may be taken away from the meeting. You may discuss concepts covered in class or assigned in the homework, but you may not discuss details of the homework. Looking at, modifying, or copying each other’s files or solutions is *strictly forbidden*. If you are unsure of what is and is not allowed by this policy, please talk to me *before* doing something that might be considered cheating.

The *Simpsons* rule also states that, following any class-related discussion, you must take a break for at least half an hour before doing further class work. Watching quality TV such as *The Simpsons*, *Futurama*, or *Family Guy* qualifies, as does watching schlock like *Jerry Springer*. Reading something (inane or otherwise) *unrelated to CMPS 122* also qualifies. See me if you’d like some suggestions for non-computer science reading material.

If all of this doesn’t convince you of the folly of using others’ work, consider this Foxtrot comic¹.



¹Author: Bill Amend; publisher: Universal Uclick. Permitted use, as described at http://www.amuniversal.com/ups/permissions/reprints_edu.html