CS 491/591 Computer Security and Privacy, Spring 2010

Instructor: Jed Crandall, jedcrandall@gmail.com

Never hesitate to email me directly about anything.

Office and office hours: FEC 335, Mondays from 2-5pm.

Prerequisites: None formally, having taken CS 341 (Computer Organization and Design) before this class is highly recommended. If you're uncomfortable with low-level systems programming or haven't learned about stuff like virtual paging or caches then come talk to me early in the semester.

TA: You're all TAs! Just kidding, but the labs will be challenging, so you are encouraged to help each other out through the mailing list.

Mailing lists: There are two mailing lists, one list that you are required to join that only I will post on, and I will only post important class announcements to that list. There is also a chat list, which is optional but highly encouraged, for students to share ideas and thoughts, things they see in the news, ask questions, get help on labs, etc. Discouraging other students from posting to this list will not be tolerated, if you feel someone is abusing the list let me know privately and I'll deal with it or simply remove yourself from the mailing list.

Course website: http://www.cs.unm.edu/~crandall/491591spring10/

I'll post lots of important stuff here, like the lab assignments, links to the mailing lists and Google calendar, grades, etc.

Required texts:

1. *Introduction to Computer Security* by Matt Bishop (the brown graduate version titled, *Computer Security: Art and Science* is also acceptable)

2. *Gray Hat Hacking, 2nd. Edition* by Shon Harris *et al.*

Class meeting time and place: MWF 1 to 1:50pm, in Woodward Hall room 149. Regular attendance in person (in Albuquerque or via one of the ITV centers where you can ask questions) is expected. Watching the ITV videos or watching the lecture live from home or work is not acceptable unless it is an excused absence. If you have no way to ask questions or participate in discussion then you really aren't in class. Maybe I'm old-fashioned, but I strongly believe that live, full-duplex channels are important for learning. If you miss two lectures in a row without contacting me, I reserve the right to use the instructor drop feature to remove you from the course.

Grading: The final grade will be calculated as 50% labs, 25% final, 10% midterm, and 15% reading assignments. The points for each will be added up and divided into the total possible before weighting, so a 5-point reading assignment does not necessarily contribute 5% as much to your grade as a 100 point final. The labs and reading assignments will not be curved, the midterm and final will each be curved *at most* 15 points., if at all. I also reserve the right to curve the overall grades at the end of the semester (up, never down) if I don't feel that they reflect the amount of effort students put into the class. The overall grade will be out of 100, weighted as described above. For letter grade purposes, below 60 is an F, 60 and up is a D, 65 and up is a C-, 70 and up is a C, 75 and up is a C+, 80 and up is a B-, 82 and up is a B, 85 and up is a B+, 87 and up is an A-, and 90 and up is an A. I only give A+'s in extreme circumstances.

Labs: There will be 3 labs, each worth 100 points. You may use any language to implement them. C and Perl are recommended, Python is also a good choice. Silly fairy tail languages such as LOLCODE or Java are strongly discouraged and will not be supported in the lab environment. A separate addendum to this syllabus regarding the labs and lab environment will be handed out after we complete the ethics portion of the course. Be sure to start early on the lab assignments and get the help you need to get them done.

For turning in the labs, you have 4 late days that you can use for any reason in increments of a whole day. E.g, you can turn in one lab 4 days late, one 3 days late and another 1 day late, *etc*. Once you've used your late days, though, they're gone.

Reading assignments: The reading assignments seem like a lot (3 or more chapters a week), but they're not that bad considering the lengths of the chapters. Your reading assignment answers should be emailed directly to me before the start of class (1:00pm sharp) on the due date. I will not accept reading assignments after 1:00pm going by the timestamps in my gmail account. Type your answers in the body of the email, I will not grade attachments. Please keep your answers as short as possible.

Midterm: The midterm will be on Friday, 12 March in class at the regular time. The test will be proctored in Los Alamos, but Sandia students need to be physically present on the main campus (in Woodward Hall room 149) to take the midterm.

Final: The final is optional for students that turn in all of labs. If you turn in all three of the labs on-time (using your late days counts as on-time for this purpose, of course) I'll enter a 100 in the grade book for the final whether you take it or not. The final will be very challenging. There is one other way to get a good grade on the final, besides studying like crazy or getting all the labs done to get out of it, which I'll explain early in the semester. I strongly recommend getting all the labs done.

We will not meet during finals week. The final will be on the last day of classes, *i.e.*, 7 May 2010 at 1pm, the normal time and place. Sandia students who plan to take the final must be physically present on the main campus to take the test that day. The final will be proctored in Los Alamos.

UNM statement of compliance with ADA: “Qualified students with disabilities needing appropriate academic adjustments should contact the professor as soon as possible to ensure your needs are met in a timely manner. Students must inform the professor of the disability early in the class so appropriate accommodations can be met. Handouts are available in alternative accessible formats upon request.”

Cheating and collaboration: I hope that you’ll help each other out on the projects. Looking at each other’s code on a monitor and sharing code snippets is okay. To prevent getting yourself into trouble you don’t deserve to be in, though, do not share your entire code with somebody as a file. Unless specified otherwise, all projects and homeworks are individual efforts. Part of learning the nitty-gritty details of systems programming is that you have to suffer individually. If you’d really like to help out your classmate and are not sure if it would cross the line into cheating, ask me. I'll try to make it clear at the top of each lab assignment what is okay and what is not okay with regards to cheating and collaboration specifically for that lab, but in general you should only help people with specific techniques or high-level discussions and should not explicitly tell them the answers to the big questions of how to do the lab or write their code for them.

Each test will state at the top what materials you’re allowed to use (book, notes, etc.). Not noticing, for example, that the top of the test says that it’s not open notes is not an excuse. Anything not specified as open is closed. In other words if the test instructions don’t say “open-iPod” you should assume that the test is closed-iPod, and if the test instructions don’t say “open-cheat-sheet-on-the-inside-of-your-water-bottle-label”, assume that the test is closed-cheat-sheet-on-the-inside-of-your-water-bottle-label.

All university policies regarding these matters will be strictly enforced.

A separate document titled "Lab Policies" will be handed out after the ethics portion of the class is complete. You must read this document before signing the ethics form and getting an account in the lab environment. That document should be considered an addendum to this syllabus.

Partial list of topics to be covered (email me with suggestions for other topics):

Ethics, legal issues, ethical disclosure, UNIX security, memory corruption, jails, TOCTTOU and other concurrency vulnerabilities, secure design principles, thread models, history of cryptography, symmetric cryptography, cryptanalysis, authentication, asymmetric cryptography, Shor's algorithm, quantum key distribution, access control models, confidentiality policies, integrity policies, hybrid policies, the Windows access control model, information flow, covert channels, timing channels, inference channel attacks, port scanning, availability, denial-of-service, TCP/IP security, BGP security, ARP security, web security, SYN floods, network intrusion detection, insertion and evasion, Internet censorship, privacy, anonymity, malicious code, malware detection, polymorphism and metamorphism, distributed systems security, Sybil attacks, differential privacy, and forensics.