CS 491/491 Computer and Network Security, Spring 2013

Instructor: Jed Crandall, jedcrandall@gmail.com

Never hesitate to email me directly about anything.

Office and office hours: Tuesdays and Thursdays 3:15pm to 4:45pm. Please come to office hours frequently. I have candy!

Reasons to drop this class right now:

- ***CS 444/544 is a prerequisite.*** I have already waived this prerequisite for some of you, if you haven't taken 444/544 and haven't been told by me that the prerequisite is waived specifically for you please come talk to me.
- ***This class is going to be very challenging.*** Students will be expected to understand advanced computer and network security concepts at a detailed technical level. Because this is an advanced class that assumes that you've taken 444/544 and have a strong background in computer and network security already, I won't be accepting excuses such as, "I don't have much experience with assembly language" or "I couldn't get Scapy to work with fragmented packets."
- ***The class is taught by me.*** I expect students to take responsibility for their own learning and help each other out. I'll probably be giving lectures less than half the time we're in the classroom, the rest of the time you guys will be presenting something or you'll be working on your project or lab assignments. Students have described my teaching style as "having to teach themselves." If you're uncomfortable with this approach it would be better if you dropped the class now.

TAs: None.

Mailing lists: There is a required mailing list, see the course website to join it.

Course website: http://www.cs.unm.edu/~crandall/491591spring13/

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

Required text: None. See the course website for readings (all of which will be freely available on the web from any campus computer).

Required materials: A decent set of colored pencils (you'll want at least a dozen different colors, the bookstore sells a set of a dozen that are erasable for just over $7 after taxes), and some blank 8.5" by 11" paper to draw on (I can't endorse that you steal the blank paper out of printer trays, but that's what I do).

Class meeting time and place: Tue/Thur 9:30am to 10:45am, in CEC B146 (the lab in the basement of the Centennial Engineering Center). Attendance is required, but will not affect your grade. If you miss two class periods in a row without notifying me why you're missing class I reserve the right to drop you

from the course.

Grading: The final grade will be calculated as 60% final project, 40% labs. The points for each will be added up and divided into the total possible before weighting, so a 100-point lab does not necessarily contribute the same amount to your grade as a 100 point final project. I 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 about 5 lab assignments. If you've taken a class from me in the past, the labs in this class will be much smaller in scope—more like homework assignments. Typically, you'll turn in about a paragraph of writing describing what you did and a colored pencil drawing illustrating what insight you gained, then I'll project your colored pencil drawing for the class and you can tell us about it. Sometimes you'll need to turn in tar balls of source code. Your colored pencil drawings need to be completely hand-drawn. The idea is to get your insights down onto paper without too many constraints.

Late assignments will be accepted only in special circumstances (medical, *etc.*).

Final project: The meat of the class will center around the question, "What is a vulnerability?" Each student will choose a medium to work in throughout the semester (weak DACLs in Windows, memory corruption, SQL injection, cross-site scripting, network inference, or whatever you're comfortable working with). Each student will be expected to identify a vulnerability in a real piece of software (or maybe hardware) that is unknown to the public. I'll assign portions of the final project throughout the semester. The first part will probably be a proposal where you specify what medium you intend to work in. In the second part (which will be due about mid-semester) you will be expected to use fuzz testing, reverse engineering, or something similar to identify an unknown, real vulnerability. Then you'll develop an exploit about it and do some further analysis in the second half of the semester.

Midterm and Final: There will be no midterm or final, or tests of any kind, in this course.

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, personal statements:

For the final project, you can seek out help and use existing code as much as you like as long as the primary effort to discover the vulnerability and develop an exploit for it is your own.

For lab assignments, each lab assignment will have specific instructions about what is acceptable in terms of cheating and collaboration. Be sure to read it, and if you don't understand it ask me questions. In general, unless otherwise specified, you are not allowed to discuss lab assignments before the due date with anybody but me (*i.e.*, you shouldn't discuss any lab assignment with your classmates until after they are due, not even at a high level). Also, I won't assign lab assignments for which code is available on the web, so use any code you want to from the web or other sources, *but not from your classmates.* There is to be no sharing of code between classmates for labs in the class, unless the specific lab assignment says clearly otherwise.

All university policies regarding these matters will be strictly enforced. Typically I'll give the cheating parties a 0 on the assignment, but I may pursue further action in some cases.

My expectations of you as students in this advanced class

- **Be studious:** I'm fairly old-fashioned, I expect students to come to class, to come on time, to stay on task, to take the time to make sure they understand things well, *etc.*
- **Take responsibility for your own learning:** We'll cover some basic background in the science of hacking, so to speak, but for the most part the knowledge I want us all to take away from this class hasn't been created yet. I've written papers and book chapters about the question "What is a vulnerability?," but I don't know the answer to that question. We'll try to answer it this semester as best we can. A good philosophical approach for you to take in this class is to "teach the teacher."
- **Do only excellent work:** Anything worth doing is worth doing well. In terms of your grade on labs and the final project, you'll be much better off doing solid work on something that is very simple than to try to do complicated things. Keep your projects and writing simple and make sure everything you do is excellent and technically sound.
- **Show leadership and be a mentor:** I expect students to do lab assignments on their own, but other than that in general I expect students to help each other as much as possible.

Material to be covered...

Ethical disclosure, legal issues, and University policy

Readings: UNM Policy 2500 and UNM Policy 2520

Vulnerability studies, what is the nature of a vulnerability?

Readings: Daniela's and my NSPW paper, weird machines, Once upon a free(), Advanced Doug Lea's malloc exploits

Different media for vulnerabilities/exploits

Readings: Idle scans, SQL injection, buffer overflows, Gray Hat chapter on Windows DACLs, physical frame injection, cache timing channels, X.509 attacks, man-in-the-middle attacks, Ptacek and Newsham, format strings, Zalewski on draining the entropy pool, weak keys, voting machine security, car security

Finding vulnerabilities

Readings: Fuzz testing, EXE: Automatically generating inputs of death, Static detection of cross-site scripting vulnerabilities

Advanced evasion techniques

Readings: Return-oriented programming, English shellcode