CS 491/591 Spring 2013 Project Proposal – 10 points out of 100 total for the class final project

This is due via an email to jedcrandall@gmail.com before 11:59pm on Sunday, 17 February 2013. You should submit a single-page PDF as your proposal.

Your final project will be to find and exploit an unknown vulnerability in a real piece of software or real system. The definition of vulnerability is somewhat flexible, for example it could be a novel inference attack that you develop for a well-known network protocol (like a new kind of idle scan) or a new IDS evasion technique.

For your proposal, you should have an introductory paragraph saying what medium you plan to work in and giving some personal background (I chose this because …, I have some experience with …, I want to learn …, *etc.*). Then you should have a few paragraphs, and perhaps a figure, detailing your plan for finding an unknown vulnerability. For example, "I'm going to use the Sulley fuzz tester to generate input files … I plan to test several non-mainstream PDF readers from download.com … I expect to find a vulnerability because PDF readers typically assume a well-formed PDF file and there are many different PDF readers from different vendors … I will use Windows XP with no service packs as the guest OS … I will script the fuzz testing using PowerShell … *etc.* You should think through what you're going to do, and discuss it with me before you start working on your proposal. All the details should be in the proposal, including exactly what software you're going to use, what versions, how you're going to set it up, *etc.*

Be sure to choose a soft target. Failure to find an unknown vulnerability is not an option. The next assignment for the final course project will require you to describe the vulnerability and give a preliminary exploit for it, and document that the vulnerability is unknown. The final assignment for the final project will be to add some kind of countermeasure and then evade it, *e.g.*, by turning on *W xor X* pages and then developing a new exploit based on return-oriented programming. In other words, it won't be possible to do the bulk of the final project until you find an actual unknown vulnerability.