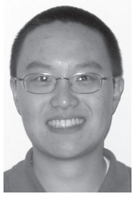
MING CHOW

teaching computer security, privacy, and politics



Ming received his bachelor's and master's degrees in computer science from Tufts University. He is a software developer, Webmaster, and instructor in Boston.

mchow@eecs.tufts.edu

IN TERMS OF PERSONAL COMPUTING, the general population is still largely clueless, both about basic issues, including how to protect themselves, and about the critical problems ahead.

At the USENIX '04 Annual Technical Conference I received tremendous motivation from the technical community about the desperate need to educate the public in computer security and privacy. The messages from the conference were clear: security is hard, complex, sensitive, and political. Very little money is spent on computer security education. Very few corporations take the initiative and responsibility to educate the public about security and privacy risks in technology products and innovations. The concluding statement from the Dan Geer–Scott Charney debate on operating system monoculture summarized the problems best: "We have dug ourselves into a deep hole, and we need to find a way out of the hole." Last December, I was offered an opportunity to teach a course entitled "Security, Privacy, and Politics in the Computer Age" at my alma mater, Tufts University, through the unique Experimental College program.

Syllabus

I wanted to cover a wide range, from high-level to low-level topics in security, privacy, and politics. I identified the topics that needed to be discussed, including file permissions, malware, firewalls, antivirus software, operating system patches, privacy-aware and privacy-enhancing technologies, and ways to protect yourself. Then I identified a small collection of advanced computer security topics, emerging technologies, and policy issues (e.g., electronic voting, DMCA, P2P, Induce Act) to be discussed. I dedicated the first week of classes to introducing students to software fundamentals: the life-cycle development process, cryptography, and the different philosophies (proprietary vs. free vs. open source software).

Assessment

Students' final grades were determined by five components: class participation (5%), portfolio (30%), position papers (30%), participation in a debate or an expert panel session (15%), and the final project (20%).

I assigned three position papers in the class. Each assignment posed a question, and the student had to respond in the affirmative or the negative, supporting their position in a one-page typed paper. Each student in the class had to participate in one of the debates or one of the expert panel sessions.

Each student maintained a portfolio of class lectures, handouts, and weekly homework assignments, designed for students to research and explain security topics (e.g., honeypots) or to use tools such as Net/MacStumbler.

For the final project, I asked the students to write a news article on a technological issue affecting society. The goal was to explain the issue to a public without much prior knowledge but very curious to find out more on the topic. I brokered a deal with the school newspaper, the *Tufts Daily,* to publish the best final project.

Course Goals

My primary goal was to inform students of the social, political, legal, privacy, and security issues in present computer technologies and innovations. During the final week of classes, the students and I put all the issues discussed during the semester into a larger framework, exploring the relationship between technology and society, and the public's need to be educated and informed on the benefits and risks in using technologies.

I urged my students to engage in constructive debates. Debates are healthy, and are essential to understanding the overall scope of sensitive and complex issues.

Student Responses

Students appreciated my talk on open source software (OSS) and were delighted when I demonstrated OSS programs such as GIMP, OpenOffice, GAIM, and even Firefox. It really struck me that most of the students had never heard of open source software, nor were they aware of alternatives to popular software packages.

Out of 23 students in my class, 13 completed a course evaluation; the results and remarks from the class were good, and surprisingly honest. On a scale of 1 to 9 (with 9 being an overall outstanding course), the course averaged an overall score of a 7.07.

All 13 students said that the course should be repeated. In general, students found that the course was quite practical and presented problems and solutions relevant to everyday life. Students found the content from the first half of the semester, which dealt with computer security and privacy, especially the in-class demonstrations (e.g., terminal exercises, screenshots, and source code), the most intriguing. Almost all students found the second half of the course, where I delved into legal and political issues, slow and boring. Many students wished there were more technical examples, especially on advanced topics such as reverse engineering of software.

The Next Time

The next time I teach this course, I will expand emphasis on the technical and hands-on topics, including data security, network profiling tools, and rootkits. I will also spread out over the semester the discussion of legal and political issues instead of consolidating them in the last four to eight weeks of class. Unlike traditional introductory courses, the content of a computer security, privacy, and politics course will certainly evolve.

Course Web Site

The Web site for "Security, Privacy, and Politics in the Computer Age" is http://www.cs.tufts.edu/~mchow/excollege, where you can find the syllabus, lectures, assignments, resources, and selected student works.