

Injecting Information Security in Core CS Courses: Methods, Challenges, and Impact

Mohamed R. Chouchane, *Columbus State University*

Abstract – *We argue that information security can and should be covered in the majority of core computer science courses, both at the undergraduate and the graduate level. One benefit of taking this approach is to strengthen our student's understanding of the various security problems in computing, as well as eliminating many of the security-critical computing habits that are often reported to be had by many IT professionals (especially the production of vulnerable software) early by educating our computer science students, from the very beginning, on the need to keep security in mind when using, designing, developing, and maintaining computing resources. Another benefit of taking this approach is to develop our students' interest in the area and readiness to tackle advanced information security course, and willingness to consider joining the ranks of information security researchers. Using as example the success of our own department at creating and maintaining a Center for Information Assurance Education, we describe the effect that we have seen and that we expect this instructional method to have on departments with various budget and student population levels, on the computing landscape in general, as well as on the quality of the security professionals and researchers that our departments produce.*

Index terms – Information Security, Core Courses, Undergraduate Courses, Graduate Courses

I. INTRODUCTION

Information security is undoubtedly one of the most intriguing areas in computing, both for users and for computing professionals. Many teenage hackers have made headlines by breaking into presumably “secure” networks[1]. For various reasons (such as revenge, political motivation, or financial gain), individuals and groups of individuals are joining forces to devise new ways of doing damage to remote computing infrastructures[2].

With the daily reports of local and global security incidents, and the damage that they have caused to the victim's reputations and budgets, most organizations around the globe are certainly aware of the security problem in computing. The Information Technology departments of these organizations often require their

employees to be able to ensure that the company's assets remain secure. Organizations have an economic incentive that each member of their IT personnel has a good working knowledge of how to practice secure computing, since any security breach might have the potential to cost the organization huge financial losses.

Many computer users are aware of some of the security issues in computing, at least when it comes to accessing some resource over a network, since, often times, a password-protected user account must be created in order for an individual to access a given resource, such as a bank account. Many security incidents, such as identity theft and malicious code, and the damage that they have caused are routinely reported to the general public by the media. This has raised many people's awareness of the security problem in computing, made many of them more careful users of computers and networks, and incited many to study the subject.

It is these people, who may include some active IT practitioners, that are among the ones that are most likely to someday enroll in our computer science courses, either to get a degree in computer science or to educate themselves on existing cyber-threats and cyber-defenses, and be able to devise and implement better and more efficient ways of defending against these threats.

Teaching information security to future computing professionals, as well as potential cyber-security practitioners and researchers, is a challenging endeavor, since secure design, implementation, and use of a computing resource must be practiced *at all times* and by *pretty much everyone involved* in these processes (e.g., database management systems must be designed and implemented with security in mind, software must be engineered so that the number of its security critical vulnerabilities is minimized, network managers should certainly be expert at assessing the security risk of their network and be able to implement the required countermeasures, etc.)

Information security is a vast and complex topic that requires students to have a solid grasp of mathematics and of a large number of concepts fundamental to computing, in addition to a full hands-on exposure to existing methods and tools for practicing secure computing.

*TSYS Department of Computer Science
Columbus State University
4225 University Avenue, Columbus, GA, U.S.A.*

Many computer science departments around the world now offer courses in information security. Many of these courses cover both the theory and practice of information security. Furthermore, given the urgent need that the global market has for security professionals, many colleges and universities prefer to educate their students on existing approaches and tools that have been used to secure computing systems.

In what follows, we show how information security topics can seamlessly be included in any computer science course, reducing the need for departments to develop new upper division (but still introductory) information security courses, hire new faculty to teach these courses, or revise their current curricula to make sure that their graduates are security-savvy.

We also argue that at least one *advanced* information security course should be a requirement for graduation for all upper-level and graduate computer science students.

This paper makes the following contributions.

- *A way to integrate information security in core CS courses.* Courses that are traditionally offered in a typical computer science curriculum include entry-level courses such as CS0 (Computer literacy), CS1 (Introduction to programming in a high-level language), and CS2 (Advanced programming in a high level language). Upper division courses include courses in operating systems, databases, and computer networks. We argue that information security topics should and can easily be covered in these and other core courses. We provide the rationale for this claim, as well as several ways for instructors to integrate these topic in their courses, without incurring a significant cost in the amount and quality of the material typically covered in these courses.
- *An analysis of the expected impact of ensuring that all of our graduates are at least information security literate.* We examine the various potential impacts that our proposed approach could have on the security-savvy of our graduates, on the opportunities that will available to them once they graduate, as well as on the overall state of the practice of computing. In particular, we emphasize the impact that the proposed approach could have on the quality of battle-ready computer-security researchers, and on the computing landscape in general.

Section II overviews related work. Section III details our suggested information security instruction method, lists a number of core computer science courses and gives a

number of suggested information security topics that could be injected into them with little to no effect on the quality and scope of the material traditionally covered in these courses. Sections IV analyzed the impact of this method on the future of our students and on computing in general. Section V briefly describes a successful implementation of this method at Columbus State University. Section VI concludes this paper and overviews some of the challenges that computer science departments will have to address to successfully implement our proposed method.

II. RELATED WORK

Recent works on information security education have focused on the development of courses that focus exclusively on teaching information security. Some of these works argue for the need to offer at least one information security course, in a given format and in a given area (such as network security), at the undergraduate or graduate levels [3], perhaps requiring an overhaul of a department's undergraduate curriculum to implement a "Threads" model [4]. Others argue for the need to extensively use hands-on labs to improve our students understanding of the material and to better assess their performance [5]. Many proposals were made to restructure the way specific information security courses (such as database security [6], network security [7], and cryptography [8]) to better teach the material to our students. Proposals were also made to improve the online instruction of some of the most important information security courses [7]. Finally, many departments have established security concentration programs where students actually are required to take a number of information security courses to graduate with a degree (B.S., M.S., or Ph.D.) in computer science with a concentration in information security [9].

These works offer very helpful insights into how we should revise the way we teach information security concepts to meet the demand that the world has for security savvy computing professionals. Revising an entire curriculum, as well as designing new information security courses to be offered by a computer science department present a substantial *organizational* and *financial* challenge to departments and universities. In these uncertain economic times, some of our colleges and universities may find it quite difficult to implement new courses, hire new faculty to teach them, or experiment with a new curriculum that may or may not be as successful as the one currently in place.

One of the contributions made in this paper is the alleviation of the financial and organizational burdens associated with a substantial curriculum overhaul. The way we teach information security to our students could be considerably improved, with minimal budgetary

impact, by *injecting* information security topics in our regular computer science courses, perhaps requiring students to take an upper level information security course that would focus on advanced topics in computer assurance, as well as solidify the security knowledge that they have acquired over their years as freshmen, sophomores, and juniors.

III. INJECTING INFOSEC

At the undergraduate level, our students are typically taught concepts that are fundamental to computer science and its applications. The following courses are normally offered by most if not all of our departments. Course names and contents may vary somewhat, but it is generally agreed that an undergraduate computer science student should have taken these courses by the time they graduate. For each course, we list a number of information security topics that we feel should be integrated, perhaps as homework or reading assignments, into the course to achieve the expected outcomes discussed in the next section.

A. Computer Literacy

This course is typically required of most if not all students, regardless of their major. Topics that are traditionally offered in this course include basic file management, network (Internet) usage, email, and downloading and installing program. At this stage of their schooling, we may certainly educate our students about many of the basic security problems that they might have to watch out for, even this early in their path towards their degree. Information security topics that can be covered in a computer literacy course include the following.

- *Use encryption to protect your files.* Students can be asked to explore the encryption features available in most modern office software.
- *Choose secure passphrase-based passwords for your files, email and Internet accounts.* Students can be asked to test the strengths of their passwords by running them through one of the many password crackers that are readily available for download over Internet.
- *Be on the lookout for social engineering and malware.* Students could be shown examples of phishing and malicious emails that they are likely to find in their Inboxes, and sometimes in their Spam folders, depending on their email filtering configuration.

Upper-level information security courses would then not have to teach these topics from the ground up.

B. CS01 and CS02

Now we are probably dealing with a student population the majority of whom are computer science majors. CS01 and CS02 courses teach students basic programming concepts such as variables, functions, object oriented programming, and elementary data structures, perhaps by having students write programs in some high-level language such as C++ or Java.

Here again is our opportunity to teach our students how to avoid the poor development practices of programs that plague software these days and contribute to some of the largest financial losses experienced by organizations all over the world. Information security topics that can be covered in CS01 and CS02 courses include the following.

- *Practice strong input validation.* Students can be asked to circumvent the input validation phase of an existing small program in order to get access to a “subscriber only” feature of that program. Good input validation should be made part of the students final scores on their programming assignments.
- *Always perform array boundary checks before indexing into an array.* Students can be shown an example where poor array boundary checking can either crash a program or give an attacker access to a protected resource.

One day these students may be the ones writing our next generation of word processors and other application programs. We certainly *must* make sure that they do not acquire the “bad programming habits” that make software an easy target for attackers, and that a number of our current programmers have been accused (rightly or wrongly) of having developed.

C. Computer Networks

It is the author's opinion that a networking course is one of the most ideal places where security should be emphasized as a major design, implementation and management goal of computer networks. After all, many security threats to a computer come from the outside. It is attackers (or attack machines) sitting at a remote location that often perform the work of doing damage to computing resource. Information security topics that can be covered in a networking course include the following.

- *IP spoofing.* Students can be asked to modify the “sender” field of IP datagrams sent by one computer to another and analyze the implications of this modification. Students will realize how easy it for an attacker to cover its tracks and that they should not completely trust that a given

packet really originates from where the sender field says it does.

- *Denial of Service.* Students can apply their recently acquired knowledge of congestion, traffic, and the client-server model of computing to fully appreciate the mechanics and implications of denial of service attacks.

These are just a two of the basic network security concepts that a student in a networking class can quickly grasp while the information about networks is still fresh in their minds. Students will quickly learn, and hopefully commit to memory, those basic security countermeasures that would make a network in a given environment less vulnerable to these attacks.

D. Algorithms

In most universities, a course on algorithms teaches students how to design and evaluate the performance of procedures that systematically solve a given problem. These often include data structure searches and manipulations, as well the study of basic complexity theory. We feel that this course could certainly cover basic concepts of cryptography. By the end (or perhaps in the middle) of this course, students will be well equipped to appreciate and evaluate the *power* and *limitations* of some of the existing ciphers. RSA and one-time-pads are good candidates for inclusion in a core algorithms course. Information security topics that can be covered in an algorithms course include the following.

- *RSA.* Covering hard problems, such as factoring, is essential in any algorithms course. Student's understanding of the hardness of factoring can be leveraged by exposing them to RSA, and having them complete hands-on exercises and observe for themselves that RSA is indeed hard to brute-force.
- *One-Time-Pads.* Time and space complexity of algorithms could certainly be illustrated and experimented with by covering the one time pad cipher in an Algorithms course. The many examples where one time pad ciphers were broken (e.g., when a pad was reused) are entertaining and would further convince the students of the (sometimes vital) importance of knowing the complexity of an algorithm.

Students will be aware of the time and space complexity issues that any attempt to brute-force break these ciphers must face.

E. Assembly Language

Malware analysts would think of a number of security-relevant topic that students taking an assembly language course should be able to quickly grasp. In order to thwart reverse-engineering efforts, malware writers tend to release their malware in an executable form. Making disassembly one of the first steps that are taken by malware analysts to analyze malware. A working knowledge of the basic structure of an assembly language program will allow the students to understand how parasitic malware infects files, and what exactly all the fuss is about virus definitions (or signatures), which often are sequences of assembly language instructions that a malware detector may use to identify a newly disassembled instance of malware. Information security topics that can be covered in an assembly language course include the following.

- *Disassemble a program and extract a malicious signature from it.* Free tools such as OllyDbg [10], will make it very easy for students to disassemble a program given to them by the instructor. The student can then follow the control flow and JMP statements for that program for any sequence of instructions that might signal a given malicious intent.
- *Investigate ways by which disassembly can be thwarted.* It is important to be able to differentiate between data and code, as well identify the targets of indirect jumps, in order to correctly disassemble a program. Students can be asked to analyze a binary obfuscated in such a way as to harden disassembly, and hence signature extraction if the binary is actually that of a malicious program.

F. Operating Systems

Kernel execution, interprocess communication, as well as file systems are commonly covered in operating systems courses. These topics are fundamental to the student's understanding of the fundamental concepts of operating systems and what really happens behind the scenes. A good understanding of these topic is also capitalized upon by malicious hackers to carry out some of the most insidious and hard-to-detect attacks on computing systems. Our student's grasp of these fundamental concepts is critical and can be improved by exposing them to some of the most fascinating security issues surrounding the design and implementation of interprocess communication algorithms and file systems. Information security topics that can be covered in an operating systems course include the following.

- *Boot sector infection.* Students can be asked to analyze how boot sector infection works and why it is hard for anti-virus detectors to catch them, since the detectors are loaded after the operating system loads itself into memory.
- *Race conditions.* Race conditions are traditionally taught in operating systems courses as a vulnerability that poor mutual exclusion algorithms could have. The author is of the opinion that it should not take too much effort on the part of the instructor to cover security-critical consequences of race conditions, such as a malicious attacker submitting a benign batch file to the operating system and then overwriting the contents (but not the name) of the file with malicious commands while the operating system is checking whether the benign set of command is actually benign.

IV. IMPACT

We expect the impact of this approach (i.e., the injection of information security topics in most, if not all, core CS courses) to teaching information security to our students to have the following impact on our students' quality as users and future designers and implementors of our computing resources.

- *Security-aware practitioners and users that never took a information security course.* Computing professionals, too, may be categorized as computer users. After all, many of them spend a large amount of their time using a computer to design, build, and maintain all or parts of computing systems. Making sure that our future professionals are well educated in the art of avoiding, detecting, and preventing malicious phenomena certainly is a step in the right direction for the profession.
- *Stronger information security courses at the graduate levels.* Government, the economy, and of course "popular demand," have encouraged many computer science departments to offer graduate programs in Information Security. By the time they reach graduate school, many of our students either have already developed certain hard-to-break programming habits (such as poor input validation) that it has taken many of them quite a while to adapt to a graduate information security curriculum. Many of our students lack the basic knowledge needed to tackle advanced issues in information security. This has forced educators to spend quite a bit of their class time introducing students to these basic issues, which the students can certainly grasp, but it will

probably take them a while to commit the information to permanent memory and be able to detect nuances and propose improvements to existing security techniques. The author is of the opinion that, should information security be injected progressively into our student's minds over their undergraduate studies, these students will be ready to immediately start and do well in an advanced graduate course on information security.

- *Fresh supply of better qualified security researchers.* Many laboratories have been created that exclusively do research in information security. For national security reasons, funding for information security research can certainly be expected to hold steady, if not increase overtime. A fresh supply of new graduate students that already are well versed in the art of information security, and who should have studied enough aspects of the topics, would certainly smooth the learning curve for these graduate as they are asked to contribute the existing body of knowledge.

V. A SUCCESSFUL IMPLEMENTATION

The computer science department at Columbus State University has been experimenting with this approach to injecting information security topics in core computer science classes for several semesters. The department runs an NSTISSI-4011 and NSTISSI-4014-compliant (as of 2007) Center for Information Assurance Education [11]. In 2003, the department awarded its first Master of Science degree in Applied Computer Science with a concentration in Information Assurance.

The program has grown considerably since then and has attracted students from all around the world and from a variety of backgrounds, including IT professionals who have been working in the field for quite some time, military personnel looking to get their degree in an area that is highly relevant to national security, as well as our undergraduate students who felt ready and eager to tackle advanced courses in information security. A number of advanced information security courses are offered at the graduate level and security research, as well hands-on experience, is required of all of the graduate students taking a security course. The department has three faculty members that specialize in information security and research. As of the time of this writing, four undergraduate and many of our graduate students are actively doing research in the area of information security.

Enrollment in this program and the security-savvy of our graduates (both at the B.S. and at the M.S. Level) have grown significantly over the last few years. We feel that this is in part due to the constant effort made by faculty to inject security topics at each step of our student's way towards their degrees. In addition to research and hands-on experiments in our labs, the following graduate courses in information security offered at the department have received great reviews from students and have greatly contributed to the growth of the Information Assurance track.

- CPSC 6126 Information Systems Assurance
- CPSC 6128 Network Security
- CPSC 6136 Advanced System Security
- CPSC 6159 Computer Forensics
- CPSC 6167 Network Risk Assessment
- CPSC 6178 Software Testing and Quality Assurance

VI. CONCLUSIONS AND CHALLENGES

The extra load of work needed for faculty to inject information security topics in their core courses, as well as students being exposed to these issues, perhaps for the first time in their lives, is indeed clear here. One of the main issues that must be addressed by computer science departments is to make sure that their laboratories are well equipped to provide students with the facilities needed to run experiments such as assessing the security of given remote machine, as well as running benign file infectors in a contained environment and observe the “before and after” images of infected files. Other challenges include convincing novice students of the ethical and privacy aspects of computing. After all, knowledge of how computing systems could be attacked is needed to build strong defenses against these attacks.

VII. REFERENCES

- [1] “Alarm raised on teenage hackers”, Retrieved on March 9, 2009 from <http://news.bbc.co.uk/2/hi/technology/7690126.stm>
- [2] "Ukrainian CyberCrime Boss Leads Political Party", Retrieved on March 9, 2009 from http://voices.washingtonpost.com/securityfix/2008/03/ukrainian_cybercrime_boss_leads.html
- [3] Patricia Y. Logan, "An Information Security Course: A Possible Antidote to Clueless Students", Proceedings of the 11th Colloquium for Information Systems Security Education, pp. 1 – 6 (2007).
- [4] Ju An Wang, Max North, and Sarah North, “Designing a Security Thread in Computing Curricula”,

Proceedings of the 12th Colloquium for Information Systems Security Education, pp. 40 – 46 (2008).

[5] N. Paul Schembari, “Hands-On Crypto: Experiential Learning in Cryptography”, Proceedings of the 11th Colloquium for Information Systems Security Education, pp. 7 – 13 (2007).

[6] Mario Guimaraes, "New challenges in teaching database security", Proceedings of the 3rd Annual Conference on Information Security Curriculum Development, pp. 64-67 (2006).

[7] Yan Bai, Wayne Summers, and Edward Bosworth, "Teaching network risk assessment to online graduate students", Proceedings of the 4th annual conference on Information security curriculum development, pp. 1 – 6 (2007).

[8] Jungwoo Ryoo and Tae Hwan Oh, "Teaching IP Encryption and Decryption Using the OPNET Modeling and Simulation Tool", Proceedings of the 12th Colloquium for Information Systems Security Education, pp. 113 – 118 (2008).

[9] Stephen S. Yau and Zhaoji Chen, "Information Assurance Concentration Programs: Integrating Information Assurance in Existing Computer Science Curricula", Proceedings of the 12th Colloquium for Information Systems Security Education, pp. 95 – 100 (2007).

[10] "OllyDbg", Retrieved on March 9, 2009 from <http://www.ollydbg.de/>

[11] "CSU Center for Information Assurance Education", Retrieved on March 9, 2009 from <http://csc.colstate.edu/cae-ia/>