Beyond the Security Track: Embed Security Education across Undergraduate Computing Curricula Using M-Thread Approach

Lydia Ray, Jianhua Yang

TSYS School of Computer Science, Columbus State University, Columbus, GA, USA

Summary

With the development of new computing techniques and the Internet, educating our college students to be equipped with security knowledge and skills becomes more and more important and urgent. In this paper, we propose a novel M-thread approach for providing security education to all CS major undergraduate students as appropriate for their concentration of study. Using this approach, small security modules for various courses will be designed and integrated into the existing computing curricula of Columbus State University. One significant advantage of this method is its ease of implementation. This approach can be successfully implemented into any undergraduate computing curriculum in any institution without incurring extra cost to students, faculty and the institution.

Key words:

Computer security, security education, computing curriculum, M-Thread, information assurance

1. Introduction

The importance of computer security cannot be emphasized sufficiently in the modern world where the usage of technology is continuously on rise. Almost all aspects of the national infrastructure depend upon the correct operation of computer and network systems. The security of these systems is imperative to the health and protection of our national infrastructure and information assets [6]. Protecting data and information on computer systems, on the other hand, is becoming increasingly more critical and challenging since the emergence of the Internet and the widespread of adoption of Web technology. Expertise on information assurance has become a necessity for many IT professionals working in government agencies, educational organizations, industry, and other businesses. Unfortunately, computer system security experts who are able to address the complexities of modern technology are still inadequate [4].

While the supply of IT professionals with necessary security knowledge still does not meet the demand, a great deal of research is also required to address the challenges existing in information assurance, and to further advance the technology. Meanwhile, many computer networks suffer from all kinds of security threats. Current computer networks are vulnerable to a large variety of attacks, such as denial of service, malicious hackers, malicious code, session hijacking, SQL injection, and so on. Attacks resulting from these threats, if successful, place networks at a big risk. Therefore, it is necessary that IT professionals also have basic knowledge on designing, deploying, and maintaining security mechanisms to ensure the security of computer and network systems.

To meet the above demands for IT professionals, computer science educators bear the responsibility to educate college students to be equipped with security knowledge and skills. B. Taylor and S. Azadegan pointed out in their research "most undergraduate computing students learn programming and design with little regard to security issues" [1]. Even products launched by software giants like Microsoft Corp. contain numerous vulnerabilities in spite of the efforts from highly paid security specialists. This indicates that incorporating security by a security specialist over already designed software does not provide sufficient protection. The fundamental concern is whether each designer and programmer has security awareness when designing and coding a system. From this perspective, all computing students should be required to receive security training and education while learning computing concepts and coding techniques. Unfortunately, this point has not been reflected in the computing curricula of most universities in the US.

Columbus State University (CSU) is a mid-size teaching university that offers the Bachelor of Science degree in Computer Science with four different tracks: systems, applications, games, and information technology. Like most other universities in the US, the current computing curricula at CSU rarely mention security topics. Only the information technology track includes a course on computer and network security (CPSC 5127). With limited budget and other severe constraints at CSU, starting a new security track is infeasible. Besides, a separate security track will not provide basic security education to *all* Computer Science undergraduate students.

In this paper, we propose a novel M-thread approach to establish a new educational pathway that spreads security education across undergraduate computing curricula and cultivate students to be equipped with appropriate security knowledge for every track of Computer Science. This approach will effectively integrate small security modules to a wide variety of courses in order to encompass all the tracks of undergraduate computing education without incurring extra cost or overload to faculty, students, and

Manuscript received August 5, 2011
Manuscript revised August 20, 2011

university administration. We plan to develop small and relevant security modules for all courses in all undergraduate Computer Science tracks without violating the integrity of the existing curricula. We also plan to train the computing faculty members at CSU to teach the security modules with respect to the corresponding courses. We envision enabling all future generation software professionals to be aware of potential security issues and implement appropriate security measures in all levels of software design.

In this paper, we describe our proposed approach in details along with our future research plans. We also present an analysis of our approach by comparing to three other existing approaches for promoting computer security education. The rest of this paper proceeds as follows. Related work will be presented at Section 2. M-thread approach will be discussed in details at Seciton3. Section 4 and Section 5 present our security module development, and associated with different courses. The analysis of the proposed approach will be given at Section 6. The whole paper is concluded at Section 7.

2. Related Work

Integrating computer security education into undergraduate computing curricula is necessary and urgent. Many security experts and educators appeal the need and importance to embed security education into undergraduate computing curricula [4, 10, 13]. M. Bishop claimed in his research [9] that every computing undergraduate student should have the basic literacy and ability to write secure code. Security education should parallel with computing education. Integrating security into undergraduate computing curricula can result in lots benefits. Many approaches have been proposed by other researchers/educators to integrate computer security education into computing program. M. Whitman and H. Mattord proposed five academic approaches to curricular development in security education [11]: 1) add modules to existing courses, 2) add elements to capstone courses, 3) create independent information security courses, 4) create information certificates or minors, 5) create information security degree program. R. Voughn [12] discussed three models for academic instruction in security. One of them is to integrate computer security education into existing computer science program. B. Taylor and his colleagues proposed an idea “cross-site security integration” which aims to bring security education to all computing students [2]. T. Yang [13] presented a comprehensive approach to integrate computer security into an existing computing program and discussed, in particular, what should be taught and how computer security could be integrated into undergraduate education. L.F. Perrone, etc. explored approaches to undergraduate instruction in computer security [8]. They proposed three approaches to integrate computer security into undergraduate computing curricula: single-course, track, and thread. The thread approach uses security and privacy as a unifying theme across the standard core computing curricula and bridges the gap between the single-course and the track approach. L.F. Perrone, etc. also claimed in their research that the thread approach is particularly suitable to mid-size teaching institutions with insufficient number of security-trained faculty.

3. M-Thread Approach

3.1 Challenges

Embedding security education into the existing undergraduate computing curricula at CSU faces some significant challenges.

- The current curricula for different tracks are already overloaded with required courses. Inadequate number of security trained faculty, lack of space and time together with the tight budget situation make it impossible to add new security courses to the existing tracks. Therefore, security education must be embedded into the existing curricula without disrupting its integrity.
- The second challenge is to determine what security contents should be integrated into the computing curricula. Computer security education is a systematic engineering program. More than just having students trained on security techniques, it should cover the law, investigations, ethics and privacy, computer security technique, physical security, and business continuity & disaster recovery. J. Powanda [7] elaborated the basics of computer security education which includes knowing and following security policy and laws, identifying potential security problems in the context, preventing security problems proactively, reacting promptly and properly a security occurrence, getting additional help or information, making informed decisions on security matters, and speaking the language.
- At CSU, currently we do not have sufficient security-trained faculty. Spreading security education across the computing curricula would require almost every faculty to be well-trained in security techniques.
- The fourth challenge is to get strong support from CSU administration. T. Andrew Yang [13] pointed out that the integration process is a continuous process which involves faculty research, professional development, curricular design, and teaching. Institutional support is critical in terms of faculty

training, travel, reduced teaching load, and internal professional development grant.

- Security issues are omnipresent [1]. Computer security issues are pervading in every sector of computing industry: from hardware to software, from application software to system software, from design to coding, from single computer to the Internet, from protocol to data communication, from database to memory management, from computer amateur to professional. It is becoming increasingly hard to integrate a wide variety of security issues into the curricula. The last challenge is to identify appropriate security issues that encompass all aspects of computer security education.

3.2 Proposed Methodology – M-thread Approach

Considering the challenges CSU is facing and the advantages and disadvantages of the above approaches, we propose a novel cost-effective *multi-thread (M-thread) approach*. Unlike the thread approach, M-thread approach spreads the security and privacy education not only across the standard core curricula, but also across the curricula of different tracks. While different tracks share the same computer security core part, they also have security and privacy issues that are uniquely distinct. Therefore, for this M-thread approach, we plan to design a security and privacy core module, as well as different modules for different tracks and inject these modules into different courses of each track in an appropriate manner.

Our proposed solution will effectively expose all CS major students at CSU to relevant computer security education while abiding by the constraints mentioned in the previous sub-section. The security-trained faculty members will be responsible for designing and developing security modules for each course. They will teach the computing security modules in different tracks and train other faculty to teach the core modules as a part of their corresponding courses. The next section presents our security module development plan in more details.

4. Security Module Development

Our ultimate goal is to expose more computing undergraduate students to security and privacy education earlier in their curricula. This requires us to instill security topics into the undergraduate curricula of the four tracks in our computing program. Based on the recommendations of the Computer Science Curriculum 2008 [2] instructions in security be presented across the core undergraduate curriculum, we propose the following development goals.

- **Security core modules** : The first goal is to design the security education module which can be integrated into the core courses of the four tracks: CPSC 1301 Computer science 1, CPSC 1302 Computer science 2, CPSC 2105 Computer organization, CPSC 2108 Data structures, CPSC 3125 Operating system, CPSC 3131 Database system 1, and CPSC 3175 Object oriented design.
- **Security modules for system track**: The second goal is to design a security module which can make the students who graduate from this track have a deep understanding about the interactions between security and systems. This module can be integrated into the following courses: CPSC 4175 Software engineering, CPSC 5115 Algorithm analysis & design, CPSC 5128 Theory of computing, CPSC 5135 Programming languages, and CPSC 5155 Computer architecture.
- **Security modules for application track**: The third goal is to design a security module to help the students in this track to have security awareness when they design and code software systems. This module can be integrated into the following courses: CPSC 2125 Internet programming, CPSC 4125 Server-side web development, CPSC 5135 Programming languages, and CPSC 5165 Web development projects.
- **Security modules for games track**: The fourth goal is to design a security module to make the students well-prepared for security concerns when developing games. This module can be integrated into the following courses: CPSC 3118 GUI, CPSC 4111/4112 Games programming I & II, CPSC 4175 Software engineering, and CPSC 5157 Computer networks.
- **Security modules for information technology**: The fifth goal is to design a security module to train the students to be keen of the Internet threats while developing a website. This module can be integrated into the following courses: CPSC 2125 Internet programming, CPSC 3118 GUI, and CPSC 5157 Computer networks.

The next section describes in details the security modules that will be integrated to different courses.

5. Security Modules to be Associated with Various Courses

In this section, we describe in details what type of security modules will be developed for various courses in the four different tracks.

5.1 Integrate Security Education into General Curricula

Courses that fall into this category are prerequisite of higher level CS courses. Thus, these courses are taken by freshman/sophomore students who have little or no idea about Computer Science. Therefore, we have carefully chosen security concepts that are relevant to these courses and are easy to grasp at the same time. Some concepts are included in more than one course in order to reinforce student learning at a deeper level.

CS Programming 1 (CPSC 1301): This is the first programming course in Java taken by CS major students. In this course, the students will gain knowledge on unreliable programming practices. They will also learn very basic defensive programming techniques that make a program more reliable and robust. The following security related topics should be discussed in this course:

- Basic programming guideline with relation to reliability of a code:
    - Documentation
    - Good choice of identifier names
    - Consistent coding style
- Importance of warnings: ignoring warnings can result in dangerous consequences
- Simple I/O validation:
    - Demonstrate how invalid I/O can result in unexpected program behavior
- Accessing variables and methods:
    - Declare variables and methods to be private whenever possible.
    - Make public access to variables only possible via accessor and mutator methods (get/set methods).

CS Programming 2 (CPSC 1302): This is the second programming course that introduces more complex concepts of programming in Java. We will introduce more complex defensive programming techniques in this course. Also we will demonstrate some dangerous programming pitfalls that make software vulnerable to attack. Below are some of the topics to be included in this course.

- Accessing variables and methods:
    - Declare variables and methods to be private whenever possible.
    - Make public access to variables only possible via accessor methods (get/set methods).
- Arrays:
    - Dangers of accessing data outside the boundaries of array
    - Basic concept of buffer overflow and its consequence (for security)
- Static variables:
    - Consequences of using non-final public static variables

Data Structures (CPSC 2108): This is the most important programming related course. Students will learn about more complex security problems associated with data structure design in programming and their solutions. Below are some example topics:

- Exception handling:
    - Security problems associated with empty catch block
    - Utility of using a “finally” block to release resources that have been claimed by “try” block
- Security concerns for various data structures:
    - Consequence of accessing memory outside the boundary
    - Stack overflow problem

Operating Systems (CPSC 3125): While the topic of security of operating systems is too broad to be covered in a general OS course, we aim at providing very basic knowledge on the security of two most important operating systems – Linux and Windows. Following topics will be included:

- Overview of the Linux Operating System
- Linux Security Basics:
- Linux User Security
- Windows NT Security Architecture Overview
- File and Directory Security
- User profile
- Registry

Database Systems 1 (CPSC 3131): This basic database course will briefly discuss the security issues associated with a database system. Solutions to these issues will be discussed at a very basic level. Some of the topics to be discussed are:

- Unauthorized access
- SQL injection
- Disclosure of configuration information
- Disclosure of sensitive application data
- Network eavesdropping problem
- Unauthorized access

Object Oriented Design (CPSC 3175): This advanced programming course will introduce important security issues involving objects and their properties.

- Inheritance:
    - Utility of making classes and methods final whenever possible
    - Consequences of calling non-final methods from constructors
- Mutable objects:
    - Problems associated with mutable objects
    - Dangers of saving references to mutable objects in some member variable, if

  - such a reference is a parameter of a public constructor or method
  - Dangers of returning references to mutable member objects from public methods

5.2 Integrate Security Education into System Track

This track focuses on the newer technologies and applications of current methods for design and engineering of software systems. Students are trained to develop an understanding of theoretical aspects of Computer Science and to analyze, design and implement a solution to real-world scientific or engineering problems. We have selected the following courses from this track to include security modules. Detailed descriptions are given below:

Software Engineering (CPSC 4175): The security module for this course will focus on the following issues of software engineering:

- Security threats to software
- Secure software requirements
- Secure software design
- Software testing for security
- Secure software tools, methods and processes
- Secure software sustainment

Senior Software Engineering Project (CPSC 4176)
In this course, students will learn how to apply their security knowledge from CPSC 4175 while building their projects.

Algorithm Analysis and Design (CPSC 5115)
In this course, instructors will discuss various applications of cryptographic hash function in ensuring system security.

Theory of Computing (CPSC 5128)
In this course, instructors will introduce the basics of cryptographic protocols.

Computer Architecture (CPSC 5155)
In this course, instructors will discuss the following topics:

- Using hardware to implement computer security
- Hardware security as an alternative to software only security
- Advantages and disadvantages of implementing security using hardware

5.3 Integrate Security Education into Games Track

This track aims to provide students with a thorough understanding of the theory, design and programming techniques required for producing games software. Users of games software may face security threats from two sides. There may be inherent vulnerabilities created by faulty programming techniques. Such buggy software when installed in a computing system may create security holes in the system. Online games software faces additional network related security risks. Vulnerabilities associated with general faulty programming techniques will be mostly covered in the general core programming courses. The game programming courses, specific to this course, will focus on specific game related security issues as described below. Also, students will learn about network security issues in Computer Networks course.

Games Programming I and II (CPSC 4111 & CPSC 4112)
In these courses, the instructor will demonstrate security threats that the players of a computer game can encounter. The following topics will be included:

- Cracking
  - How a game program can be cracked
  - How to prevent cracking through using special programming techniques
- Cheating
  - Programming techniques in multiplayer games that prevent players from cheating
- Security issues in online games
  - Network security issues with respect to the Internet applications
- Security holes and prevention in object-oriented programming
  - Security concepts introduced in Object Oriented Programming (CPSC 3175) will be reinforced with greater depth

Computer Networks (CPSC 5157)
In this course, network security will be discussed. The following topics will be included:

- Security threats and vulnerabilities in network environment
- Implementing security in different network layers
- Secure sockets layer (SSL)

5.4 Integrate Security Education into Application Track

The objective of this track is to provide training in computer programming and computer science principles as professional disciplines. Internet programming and website engineering are the main components of this track. Improper design and coding techniques of a web have made the Internet a heaven for hackers. Students of this track will be trained to identify and resolve security issues associated with the design of websites and web applications.

Internet Programming (CPSC 2125): The security module for this course will cover the following topics:

- Security issues related to web design
    - Consequence of improper coding that establishes connection between online database and web interface
    - Consequence of inclusion of external files for functionality
- Solutions to the above problems.

Server Side Web Development (CPSC 4125): The security module for this course will teach students how to implement security at the design level of a web server.

- Vulnerabilities in server side includes (SSI) and solutions to avoid them
- Abusing known targets: various situations and possible solutions
- Cookie abusing
- Setting up Setuid on Server to Restrict Execution Access: damaging side effects
- Extra protection for data files

Web Development Projects: In this course, students will be expected to apply their security knowledge from CPSC 2125 and CPSC 4125 while building their projects.

5.5 Integrate Security Education into IT Track

Introduction to Computer and Network Security (CPSC 5127) is a mandatory course in this track. Besides this course, IT track students will also obtain security knowledge from database systems, object-oriented programming, and professionalism in computing through courses CPSC 3131, CPSC 3175, and CPSC 3165 (as discussed in the previous subsections), respectively.

6. M-Thread Approach Analysis

There exist a few common approaches to integrate security education to undergraduate computing curricula. The simplest one is *single-course approach* that can be implemented with only security trained faculty incurring the least cost. With this approach, only one course concentrating on the most common topics of security should be offered as a required course for all undergraduate computing students. While this approach is low cost and simple, it is very difficult to ensure uniformity in prerequisites and the standard of the single course across the nation [8]. Besides, a single security course can introduce students to a few basic security concepts. It fails to provide special security training as appropriate to different types of IT professions.

A more effective approach is the *track approach*, where a separate security track is implemented for undergraduate computing students. The main benefit of this approach is that, unlike the single course approach, it presents security as a process [8]. Students get a more in-depth training in security. A security track offers a variety of security related courses. Therefore, the scope of this track can be increased as needed.

The track approach has a few significant drawbacks. While it provides a group of students an in-depth exposure to security knowledge, it fails to provide a broader coverage of computer security. This approach will generate a group of well-trained security experts. But students who choose to specialize in applied computer science, games design, information technology, or any other tracks, will not get security education as required for their future profession. Besides, track approach is expensive. A track in Information Assurance with many security related courses will require a group of security-trained faculty members.

The importance of security in computing will tend to get increase exponentially as networking technologies develop further. Security education needs to be emphasized across the core curriculum in order to obtain a broader coverage at nationwide.

The thread approach addresses the broader coverage requirement of security education. With this approach, the security education is integrated into the core part of a computing curriculum. Being a part of core curriculum, the security education is exposed to *all* computing students. Furthermore, this approach provides an early exposure to security education showcasing the importance and relevance of security in different areas of Computer Science. The early exposure to security education helps reinforce security concepts better in the minds of undergraduate students. Students learn to accept security as an integral part of any digital design.

Our proposed *M-thread (multi-thread)* approach extends the thread approach and spreads security education not only across the core curriculum, but also across different tracks. The benefits are manifold:

- *All* undergraduate computing students will obtain security education *as appropriate* to their future profession. The type of security education required by an IT specialist is definitely different from that needed by a game designer. M-thread approach addresses different security needs of different computing professions.
- The goal of this approach is to inject the relevant security concepts in a very small amount into all possible courses across different tracks. Thus, this approach does not incur extra overhead to faculty members or students. Moreover, concepts of security in similar courses get better reinforced. For example, freshmen students learn some simple techniques of defensive programming in the introductory programming courses. They learn more in-depth

defensive programming techniques in more advanced programming courses.

- This approach is very flexible. Any institution with any number of security-trained faculty members can adapt this approach without disrupting the integrity of their existing curricula.
- This approach also incurs least cost to an institution since it does not require additional faculty members or any major change to existing curricula.

7. Conclusion

We propose a novel M-thread approach for providing security education to all CS major undergraduate students as appropriate for their concentration of study. Using this approach, small security modules for various courses will be designed and integrated into the existing computing curricula of CSU. As a direct impact of this approach, all CS major undergraduate students at CSU will have an exposure to security education appropriate for various software related professions. This approach will strengthen security education of all Computer Science faculty members. One significant advantage of this method is its ease of implementation. This approach can be successfully implemented into any undergraduate computing curriculum in any institution without incurring extra costs to students, faculty and the institution.

In near future, we will develop security modules in detail that we proposed in this paper. And also, we will focus on evaluating the effectiveness of this approach in promoting security education to undergraduate students.

References

[1] Blair Taylor, Shiva Azadegan, Moving beyond security tracks: Integrating security in CS0 and CS1. SIG Computer Science Education (SIGCES), ACM (2008) .

[2] Blair Taylor, Harry Hochheiser, shiva Azadegan, And Michael O'Leary, Cross-site security integration: preliminary experiences across curricula and institutions, Proceedings of the 13th colloquium for information systems security education, Seattle, WA, June 1-3, 2009.

[3] Association for Computing machinery, IEEE computer society, Computer Science Curriculum 2008, December 2008.

[4] Irvine, C.E., Chin, S. and Frincke, D. 1998, Intergrating Security into the Curriculum, IEEE Computer, pp. 25-30., Dec. 1998.

[5] Information Security Job Market Overview, available at: http://www.odinjobs.com/Information-Security_job_market_overview.html, accessed at 2011

[6] Hill, J.M.D., Curtis, J., Carver, A., Humphries, J.W., and Pooch, U.W. Using an isolated network laboratory to teach advanced networks and security. In SIGCSE'01: Proceedings of the 32nd SIGCSE technical symposium on Computer Science Education, New York, NY, USA, 2001.

[7] Powanda, J. 1999: Assembling a curriculum for various security disciplines, 12th Annual FISSEA conference, NIST.

[8] Luiz Felipe Perrone, Maurice Aburdene, and Xiannong Meng, Approaches to undergraduate instruction in computer security, Proceedings of the 2005 American Society for Engineering Education Annual Conference & Exposition, 2005.

[9] Bishop, M. and Frincke, D. 2005. Teaching Secure Programming, IEEE Security and Privacy 3(5) pp. 54-56, Sep. 2005..

[10] Howard, M. and LeBlanc, D. 2003. Writing Secure code, Microsoft Press, Redmond, WA.

[11] Whitman, Michael e. and Herbert J. Mattord, A draft model curriculum for programs of study in information security and assurance. Proceedings of the 8th colloquium for Information System

[12] Vaughn, Rayford. Building a computer security emphasis in Academic programs. Proceedings of the fourth annual national colloquium for Information system security education, pp. 90-94, 2000.

[13] T. Andrew Yang, Computer Security and Impact on Computer Science Education, May 2001, ACM Journal of Computing in Small Colleges, Vol. 16, p.233.

Lydia Ray was born in India in 1975. She earned her PhD degree in Computer Science from Louisiana State University, Baton Rouge, Louisiana in 2005, M.Stat degree in Statistics from Indian Statistical Institute in 1998, and B.Sc degree in Statistics from University of Calcutta in 1996. Her major field of study at present is Computer Science. She is currently working as assistant professor at TSYS Department of Computer Science, Columbus State University. She joined here in 2006 Fall. Her research interest is in sensor networks and security.

Jianhua Yang earned his Ph.D. degree in computer science at University of Houston, Houston, TX USA in 2006, Master degree in computer engineering in 1990 and Bachelor degree in electronic engineering in 1987 at Shandong University, China. His major field of study is computer science. He is currently working at TSYS School of Computer Science, Columbus State University (CSU), Columbus, GA USA as an Associate Professor. Before joining CSU, he was an Assistant Professor at Bennett College for Women from 2006 to 2008, University of Maryland Eastern Shore from 2008 to 2009, and Associate Professor at Beijing Institute of Petro-Chemical Technology, Beijing, China from 1990 to 2000. His research interests include computer network and information security, digital topology. Dr. Yang has published more than 30 research papers in the area of stepping-stone intrusion detection since 2004. He is serving as a reviewer for IEEE Transactions on Signal Processing, Journal of Computers and Security, Elsevier, and many other professional journals.