A DEDICATED UNDERGRADUATE TRACK IN COMPUTER SECURITY EDUCATION

S.Azadegan, M. Lavine, M. O'Leary, A.Wijesinha, M. Zimand
Towson University

Abstract: To better prepare our graduates to face the challenges in computer and information security, in Fall 2002, the authors launched an undergraduate track in computer security for the computer science majors at Towson University. This paper describes the motivation for this track and discusses its structure and requirements.

Keywords: Undergraduate Computer Security Education, Cryptography, Network Security, Operating Systems Security, Application Software Security, Computer Security Case Studies

1. INTRODUCTION

Our computer security track addresses the need for skilled personnel in the Computer Security field and provides an opportunity for our undergraduate students to be educated in this field. What is unique about this track is that at the time of its development, summer 2001, it was the only undergraduate track in computer security in the State of Maryland. Also, based on a preliminary search done on the Web, Towson University is one of the few Centers of Academic Excellence in Information Assurance Education with an undergraduate program in computer security. Another unique feature of this track is our capstone course, titled "Computer Security Case Studies". Students take this course as the last course in the track and collaborate with their classmates on real-world projects using their knowledge and skills gained in other courses of the track.

Students graduating with this track will have a strong background in the fundamental principles of computer security and its applications, plus hands-on experience with security tools commonly used in industry. This will better prepare them to join the 21st century workforce and to protect our national infrastructure and information assets, which is in keeping with the influential "Call to Action" document [1] that identified the lack of security skills as one of the top ten trends impacting security.

2. GOALS AND OBJECTIVES

The objective was to develop a high-quality computer security track that uses and builds upon the courses that are already part of the computer science curriculum and allows students to finish their degree in four years. The courses in this track expose students to a wide range of security problems and vulnerabilities that exist in operating systems, application systems and networking protocols, and shows students how these vulnerabilities might be exploited by potential adversaries. The students will be given opportunities for hands-on experience with the tools and software used to secure systems.

At the present, the developed track is only available to our computer science majors and computer science and mathematics double majors. Both programs are accredited by Computer Science Accreditation Commission (CSAC) and the track allows students to join the program at any time before the start of their third year. We designed the track with the expectation that it will be accredited by the CSAC at their next visit scheduled for the AY 2003-2004.

The development of the track started over two years ago and the track was approved and made available to students as of Fall 2002. In spite of this short period of time, the track has become very popular among the students, to the extent that our computer information systems majors have been requesting the creation of a similar track for their program. Other institutions offering an accredited program in computer science can easily adopt this track.

3. THE TRACK

Our computer Security track is our standard Computer Science program where upper level computer science electives are replaced by security courses. Most of our upper-level computer sciences courses run with fewer than 25 students. The table below contains the Computer Science program and also illustrates the changes that were made to incorporate the security track.

Computer Science with a Track in Computer Security	**Computer Science**
Required Core courses Computer Science I and II, Computer Architecture, Data & File Structure, Computer Organization, Operating Systems, Programming Languages: Design & Implementation, Database Systems, **Computer Ethics**	The same
Required Math courses Calculus I, Calculus II Discrete Math, and Statistics	The same
Introduction to Cryptography	One additional upper level mathematics course
Science Requirement (12 credits)	The same
Introduction to Information Security, Network Security, Application Software Security, Operating Systems Security, Case Studies in Computer Security	Elective Computer Science courses (12-14 credits)

The following seven courses are the key components of this track:

1) Computer Ethics
2) Introduction to Information Security
3) Introduction to Cryptography
4) Network Security
5) Application Software Security
6) Operating Systems Security
7) Case Studies in Computer Security

Although, there are other courses that could be included in a computer security track, e.g. computer forensics, biometrics, and communications law, we selected these courses focusing on our existing program and utilizing our expertise and strengths. Below we describe each of the courses that students in the computer security track have to complete.

3.1 Computer Ethics

This is a traditional computer ethics course and it is a required course for all students majoring in Computer Science. This course has been taught for many years and prepares students to deal as professionals with ethical questions and societal concerns related to the widespread uses of computers and resulting responsibilities of computer scientists. In this course, topics such as: intellectual property rights, privacy issues, computer crimes, codes of ethics and legislation regarding computer technology are covered. The textbooks for this course are *Readings in CyberEthics* by Richard A. Spinello and Herman T. Tavani [2] and *Morality and Machines: Perspective on Computer Ethics,* by Stacey L. Edgar [3].

3.2 Introduction to Information Security

This course provides students with a very broad understanding of the major technical and human components of information security. It focuses on information systems security threats, vulnerabilities, technologies and business requirements. Emphasis is placed on the human and technological aspects of IT security problems and issues relevant to the risks in which information systems are exposed and methods of dealing with such risks. The prerequisite for this course is junior standing and students are strongly encouraged to take this as the first course for the track. This course has been taught three times and is also available to undergraduate students in our computer information systems major as an elective.

The following is a summary of the major topics covered in this course:

- Identification, Authentication and Access Control;
- Security Threats and Vulnerabilities;
- Security Models, Security Requirements and Standards;
- The Security Kernel;
- Network and Distributed Systems Security;
- Internet Security and Cryptography;
- Operating System Security;
- Database Security; and
- Legal and Ethical Issues

We selected *Computer Security* by Gollmann [4], and *Security in Computing* by Pfleeger [5] as the required textbooks for this course. Both books have their own specific advantages from a instructional/teaching perspective and a diversity of subject matter point of view. The combination of these books seems to be highly effective for this type of course. A significant amount of extra course materials are distributed throughout the duration of the course. In addition, a considerable amount of recently published articles are circulated by the instructor. Furthermore, students are also encouraged to bring in related articles from web sites, practitioner journals, magazines and newspapers.

This is a rigorous course and includes a diverse set of coursework. In total, there are two examinations, two small applied/hands-on projects, various written homework assignments, a group research paper and a group presentation. The examinations are based on the topics and sub-topics of the course and are usually a combination of objective, multiple choice questions and short essay questions. One of the applied projects for this course deals with a comparison of SSH and Telnet. This project is effective for student learning about utility programs and control techniques, and also provides a useful comparison of the historical and current developments in information security. The other hands-on project is based on PC vulnerability scanning. To our students, this appears to be the more interesting and intriguing of the two applied projects., since it highlights a number of the common issues with protecting individual personal computers such as: primary password controls, unpatched operating systems, configuration issues, and user and administrator privileges.

The homework assignments tend to deal with current topics in Information Security such as Information Warfare, Cyberterrorism and Critical Infrastructure Protection. For the group research project, small groups of four to five students are required to submit a project proposal outlining the specific area of research that they would like to conduct. After instructor review and approval, these projects proceed as ‘traditional’ research papers of twenty to twenty five pages in length. These papers allow students to pursue an area of specific interest and have included a diverse set of topics as: wireless security, encryption techniques, PKI, firewall architecture and employee monitoring. The group presentation is included as a thirty-minute overview of the research paper to encourage teamwork and information sharing.

Another key component of this course are guest lecturers that primarily come from private industry and consulting organizations. In the most recent semester, two guest lecturers from a major consulting organization provided excellent individual lectures on Windows and UNIX security. Another example is where an Information Security Officer from a major IT corporation gave a guest lecture on Network and Internet Security. These guest lectures were very successful and the students in the course felt that this was one of the most important features of the course. In its totality, this course provides students with a strong understanding of the fundamental principles of computer and security and lays the foundation for the other courses in the academic track.

3.3 Introduction to Cryptography

The course gives a broad overview of the mathematical basis of modern cryptography and of the main cryptosystems currently in use. Students taking the course are exposed to relevant chapters of number theory and computational number theory at a level appropriate for undergraduates. The course covers the most important cryptosystems (e.g. DES, Rijndael and some other AES finalists, RSA, Diffie-Hellman key exchange. etc.) and the basic tools used in building security mechanisms. Some important methods for cryptanalysis are presented as well.

The course also provides an overview of some important protocols having a strong cryptographic flavor. At the end of the course, students should have a good understanding of the theoretical foundations of cryptography and of the basic techniques in achieving different cryptographic services. Discrete math and junior standing are the prerequisites for this course. The course was taught for the first time in Fall 2002.

The following is a summary of the major topics covered in this course:

- **Basic concepts of cryptology:** (Historical ciphers, cryptanalysis of historical ciphers, one-time pad)
- **Modern Symmetric Cryptographic Systems**: (DES, differential cryptanalysis, triple DES, modes of operation (ECB, CBC, CFB, OFB), Advanced Encryption Standard - Rijndael, and brief coverage of the other finalists)
- **Basic Number Theory:** (Euclidean algorithm, modular arithmetic, Chinese Remainder Theorem, Fermat's Little Theorem and Euler's Theorem, primitive roots, quadratic residues, finite fields)
- **Public Key Cryptography:** (RSA , attacks on RSA, factoring and primality testing, implementation issues of RSA, discrete logarithms, ElGamal public key cryptosystem)
- **Data integrity and authentication:** (Diffie-Hellman key exchange protocol, digital signature schemes, hash functions, pseudo-random generators, security of hash functions and MACs, public-key infrastructure)
- **Protocols**: (secret sharing schemes, bit commitment schemes, 2-party and multi-party protocols for private distributed computation,)
- **Zero-Knowledge Techniques**: (Basic scheme, Feige-Fiat-Shamir identification scheme)

We chose *Introduction to Cryptography* by Wade Trappe and Larry Washington [6], as the textbook for this course. This book is quite comprehensive in its choice of topics and achieves a carefully crafted and intelligent balance between mathematical rigor and accessibility to students that have little background in number theory. Numerous handouts containing up-to-date information from a variety of sources are distributed throughout the course. Moreover, the students are encouraged to seek such information that can be shared with the whole class.

The coursework is diverse. There are two exams and a number of short quizzes, written homework assignments, lab activities, and a team project. The exam and the quiz questions are of the short-essay type and try to assess the level to which the students have grasped the main concepts.

A key component of the course is its laboratory exercises. A lab consists of a set of activities followed by a series of questions. The Labs are supported by software packages that have been written by us (partially inspired by similar lab software written by Kris Gaj [7] from George Mason University). Currently, such labs exist for DES and RSA, and some other labs are in different stages of design and implementation for Number Theory Concepts and Rijndael. For illustration, we reproduce one question from our RSA lab: "*Using the Prime Number Theorem, estimate the number of odd numbers that one has to check to find a prime number having 256 bits. Execute the demonstration program (option Demo-Find a Prime Number). Enter the number of bits (256 in this case). You will see two numbers. One is the randomly generated initial odd number and the other is the prime number that was found. You will see how many odd numbers were tested before the program obtained the prime number. Have you been lucky or unlucky in your search for prime numbers*?"

In addition to homework assignments, the students are asked to develop either a software project or an analytical project based on cryptographic techniques. Software projects typically involve writing a program in a high-level language (C, C++, Java, etc.). Analytical projects involve comparative analysis of competing algorithms, protocols, or implementations. They also may involve reviewing/surveying issues related to cryptology and some other field such as number theory, physics, or law.

Though the topic is open, students are offered a list of suggestions for their projects, and they need to first submit a proposal that has to be approved by the instructor. Typically, students are asked to research a cryptography application and to implement it in as realistic a setting as possible. The usual steps in realizing the project are: (1) survey the literature relevant to the protocol (the instructor provides a starting point), (2) implement the protocol in a distributed environment (multiple PCs for example), (3) provide a nice GUI and (4) write a report describing the problem, solutions from literature, current solution, and implementation issues.

3.4 Network Security

The course provides students with a thorough understanding of the concepts underlying all aspects of network security with an emphasis on applications. It covers network security principles and applications. Topics include authentication applications, IP security, Web security, network management security, wireless security, and system security. The prerequisites for this course are computer networks, a required course for our computer science majors, and cryptography. We have chosen *Network Security Essentials* [8] by William Stallings, as the textbook for this course. This course will be taught for the first time in Spring 2003.

The following is a summary of the major topics covered in this course:

- **Introduction to Network Security:** Security Architecture, Attacks, Services and Models; Recent Developments
- **Review of Cryptography**: Encryption, Public-key Cryptography and Message Authentication
- **Authentication Applications**: Kerberos, X.509 Authentication Service
- **E-mail Security**: PGP, S/MIME
- **IP Security**: IPSec, Virtual Private Networks, IPv6 Security, Mobile IP Security
- **Web Security:** Secure Sockets Layer and Transport Layer Security, Secure Electronic Transaction
- **Network Management Security**: SNMP Security
- **Wireless Security**: Wireless LAN, Bluetooth, and GSM Security
- **System Security Overview**: Intrusion and Intrusion Detection, Viruses and Worms, Firewalls, Denial of Service, Honeypots

The students in this course, working in small groups, will be given 4 or 5 computer assignments (some will involve programming) that deal with network security applications and tools. A sample assignment involves the use of the Snort package [9]. The objective is to introduce students to a popular security tool and its major features. In particular, students will gain experience with Snort's capabilities as a sniffer, a logger, and an intrusion detection system. The assignment requires students to become familiar with Snort commands and alerts, and to write their own Snort rules. At the end, students will study the effect of modifying the rules, examine packets in the log files, and analyze the results.

3.5 Operating System Security

This course allows students to gain an in-depth knowledge of security threats, different types of malicious codes, and access control problems in the context of operating systems. We will discuss intrusion detection techniques and the design of trusted operating systems along with their cost and performance analysis. The prerequisite for this course is operating systems, a required course for our computer science majors. The prerequisite provides the theoretical foundations and this course will be more applied and topics will be discussed in the context of Linux and Windows NT operating systems. We have selected *Maximum Linux Security* [10], Anonymous, and *Window NT Security* [11] by Michael Mclnerney, as the textbooks for this course. This course is under development and will be offered in Fall 2003.

The following is a summary of the major topics covered in this course:

- **Overview of the Linux Operating System**
- **Linux Security Basics**: (User accounts, Discretionary Access Control, Network Access Control, Intrusion Detection)
- **Linux User Security**: (Password Attacks, Data Attacks)
- **Linux Network Security**: (Malicious Code, Sniffers and electronic eavesdropping, Scanners, Spoofing)
- **Overview of the Windows NT Operating Systems**
- **Windows NT Security Architecture Overview:** (Layered approach to securing your network, Modules of NT Security Architecture, Security implementation overview)
- **File and Directory Security**: (Disk Partition, File & directory permission, File & directory security, Share permission)
- **User Profile**: (Overview, Profile permissions, Default profile)
- **Registry:** (Registry Structure, Registry Tree permission, Registry editing tools)

Similar to the Network Security course, students will work on small bi-weekly programming projects to gain hands-on experience with the security issues covered in class. As mentioned earlier we are using VMWare products that allow running of multiple operating systems on the same machine. Moreover, students can get root access to the virtual operating system without compromising the security of the underlying machine. In this setting, students can become familiar with the privileges and responsibilities of both an administrator and a user.

3.6 Application Software Security

This course studies the security concepts in developing software applications. It discusses design principles for secure software development, and some of the security issues in current programming and scripting languages, database systems and web servers. The *Survey of Programming Languages* course is the prerequisite and the *Database Systems* course is the co-requisite for this course. We have chosen "*Building Secure Software: How to avoid Security Problems the Right Way* [12], by John Viega and Gary McGraw as the textbook for this course. This course is under development and will be offered in Spring 2004.

The following is a summary of the major topics covered in this course:

- **Software Security:** (Security Goals, Common Software Security Pitfalls, Overview of Software Risk Management for Security, Software Security Principles, Auditing Software, Selecting a language)
- **Java Security:** (Java Virtual Machine, Byte code Verifier, Java Sandbox, Java Language security constructs, The Class loader, Class accessibility, Java Cryptography architecture)
- **Secure CGI/API Programming**
- **Buffer Overflows:** (Overview, Defending against Buffer Overflow, Internal Buffer Overflows, Heap Overflows, Stack Overflows, Attack Code)
- **Database Security: (**Security Problems in Databases, Secure DBMS Design, Security Controls, Using Views for Access Control, Field Protection, Statistical Database Protection, Statistics Concepts and Definitions, Security against Statistical Attacks)
- **Client-side Security: (**Traditional Threats, Using SSL, Browser as a security hole)
- **Server-side Security:** (Current Major Host Security Problems, Minimizing Risk by Minimizing Services, Secure Content Updating, Physical Security, Access Control Strategies)
- **Firewall:** (Basic Architecture, Client Proxies, Server Proxies)

As apparent by the topic selection, the students in this course will get a chance to explore the security issues relevant to web-based technologies. In this course, in addition to small projects, students will work on a team project, which allows them to integrate and use their knowledge about Java security, database security, client-side and server-side security in a real-world E-Commerce project.

3.7 Computer Security Case Studies

This is a capstone course that allows students to work on comprehensive security-related projects. Currently in development, it will provide students with an in-depth study of the practical aspects of computer security vulnerabilities in a hands-on laboratory setting. The prerequisites for this course are Network Security and Operating Systems Security. Course work will consist primarily of computer laboratory projects. There will be no exams. Course assignments will consist of 6 to 8 projects, some smaller homework assignments, and a final paper. Towards the end of the semester, speakers from industry and government will be invited to present topics of particular interest to them in the areas of computer and network security, computer ethics, public policy for computing and security. This interaction between students and industry and government representatives would allow them to get first-hand knowledge about real projects and problems. It will provide students the opportunity to establish connections with their potential future employers. This course will be first offered in Spring 2004.

Case Studies:

- **Case Study 1: Services** (FTP, Mail, Telnet / SSH, Web Servers, File Sharing, Finger, WebDAV)
- **Case Study 2: Hardening a Server** (Linux / UNIX, NT / 2000 / XP)
- **Case Study 3: Sniffers & Spoofing** (Hubs vs. Switches, Detecting Sniffers, IP Spoofing, ARP Spoofing, DNS Spoofing, EMail Spoofing, Web Spoofing)
- **Case Study 4: Session Hijacking & Anonymity**
- **Case Study 5: Firewalls & Scanners**

- **Case Study 6: Intrusion Detection Systems** (SNORT, TripWire, Logging and Audit Tools, Web Server, Tools for Analyzing Log Files)
- **Case Study 7: Password Attacks & Encryption** (Password Management, NT / 2000 Password Implementation, UNIX / LINUX Password Implementation, Web Server Passwords, Application Passwords, PGP, Steganography, Password Cracking Tools)
- **Case Study 8: DoS Attacks** (Email floods, Network DoS, Network DDoS)
- **Case Study 9: Malicious Code** (Viruses, & Worms, Detection & Removal, Policies and Procedures).

4. ISOLATED COMPUTER SECURITY LABORATORY

For the track to be successful, it is necessary to provide an environment that facilitates active learning and allows maximum opportunity for hands-on experiences for the students. Moreover, the nature of the experiments and projects does not allow the use of a general-purpose computer laboratory. Computers used for computer security experiments can never be considered to be in a 'safe' configuration for general use. Further, the configuration of these systems will be constantly changed, based on the needs of a particular laboratory exercise. Therefore, such systems cannot be maintained in a consistently configured state for general use. We are creating an isolated computer security laboratory. Access to this lab will be limited to the students who are taking the course and they will be closely supervised. Security measures will be implemented to prevent any unauthorized and inappropriate access. Students taking the security track, will be given a document describing the code of conduct and general responsibilities of the students [13]. They will be also be asked to sign an agreement acknowledging that they have read and understood the code of conduct of the computer security track and will act at all times with accordance with that code. We are in process of creating the laboratory and will be able to report on its status at the conference. In this laboratory, we plan to use the VMWare™ Workstation 3.2 [14] software product to allow running of multiple operating systems and easy re-configurability of the machines.

5. CONCLUSION

In this paper, we presented a track in computer security that can be easily incorporated into any computer science program and described its courses in detail. At Towson University, the track was made available to our students, as of Fall 2002. During the fall semester, Introduction to Information Security and Introduction to Cryptography courses were offered. The Network Security course is scheduled for the Spring 2003 semester and the remaining three courses will be offered next academic year. The track has already attracted many students, and we are getting requests from our Computer Information Systems majors to either allow them to take this computer science track or to offer a similar track for them.

ACKNOWLEGMENTS

This work was supported by the National Science Foundation under grant No. DUE-0113783. We express our thanks to the academic and industry consultants and evaluators Dr. Deborah Frincke and Dr. Ronald Gove.

REFERENCES

[1] http://www.cerias.purdue.edu/events/accenture_cta_1q2001.pdf

[2] Spinello, Richard A., Tavani, Herman T., Eds., "Readings in CyberEthics," Jones & Barlett, 2001.

[3] Edgar, Stacy L., "*Morality and Machines,*" Second Edition, Jones and Barlett, 2002.

[4] Dieter Gollmann, "*Computer Security,*" John Wiley & Sons, 1999.

[5] Charles P. Pfleeger, "*Security in Computing,*" Second Edition, Prentice Hall PTR, 1997.

[6] Wade Trappe, Larry Washington , "*Introduction to Cryptography,*" Prentice Hall, 2002.

[7] Kris Gaj, Web page for Cryptography and Computer Network Security course, http://ece.gmu.edu/courses/ECE543/index.htm

[8] William Stallings, "*Network Security Essentials,*" 2nd Edition, Prentice Hall, 2003.

[9] The Open Source Network Intrusion Detection System, Web page for Snort, http://www.snort.org/

[10] Anonymous, "*Maximum Linux Security,*" Second Edition, SAMS, 2001.

[11] Michael Mclnerney, "*Window NT Security,*" Prentice Hall, 2000.

[12] John Viega and Gary McGraw, " *Building Secure Software: How to Avoid Security Problems the Right Way,*" Addison Wesley, 2002.

[13] Julie J. C. H. Ryan, Daniel J. Ryan, "Institutional and Professional Liability in Information Assurance Education, " George Washington University.

[14] VMware, Web page for VMWare products, http://www.vmware.com/products