

TEACHING COMPUTER SECURITY TO UNDERGRADUATES

A Hands-On Approach

Rahul V. Tikekar

Southern Oregon University

Abstract:

Increasing awareness of the vulnerabilities of computer systems has led to the introduction of several programs in computer security. Many of these programs are meant to attract graduate students. Southern Oregon University has recently started a new undergraduate track in computer security and information assurance (CSIA). The curriculum is first of its kind in Oregon to be available at the undergraduate level. The CSIA curriculum is based on a set of core classes from computer science, mathematics, and inter-disciplinary topics. Laboratory and hands-on exercises are used extensively to augment the matter covered in class. This paper describes the laboratory and other hands-on exercises of one course in the CSIA curriculum.

Key words: CSIA curriculum, undergraduate curriculum, computer security education, lab exercises

1. INTRODUCTION

Strengthening our nation's defense against malicious IT attacks on computer systems has taken on an important role in the last few years. In this time period the Internet has grown from a static web "homepage" distribution system to a flourishing society that is engaged in transacting business all over the world. Our society and economy today are dependent on information technology and information infrastructure [1]. This dependence has introduced many security risks that have manifested in the form of attacks on computer systems.

With the ever-increasing number of threats to computer systems, employers will be looking for qualified professionals to keep their systems and information secure. It is therefore important for universities to incorporate computer and information systems security into their curriculum. The 'Federal Cyber Service: Scholarship for Service Program' [2] seeks to increase the number of qualified students entering the fields of information assurance and computer security and to increase the capacity of the United States higher education enterprise to continue to produce professionals in these fields.

SOU is a 5000-student, publicly funded, liberal arts university serving a rural population 300 miles from the nearest large city. The computer science department offers four undergraduate tracks: computer science and programming, computer information systems, computer multimedia systems, and computer security and information assurance (CSIA). A masters degree is also offered that is aimed at preparing students for the current job market. A major department goal is to enhance its reputation and ability to recruit nationally in order to expand the pool of potential students. It is the hope of the department that the CSIA curriculum will attract students to the school and the department.

The complete course requirements and description of the CSIA curriculum can be found at the department's web site [3]. One of the focuses of the curriculum is to provide students with hands-on experience in real-world problems. This experience is provided through lab exercises, class projects, and capstone projects. In the summer of 2003, the author attended the Workshop on Education in Computer Security (WECS5) and the World Conference on Information Security Education (WISE3). This paper describes some of the lab and hands-on exercises, and projects used by the author – some a result of knowledge gained by the materials and papers presented at the workshop and conference.

The rest of the paper is organized as follows. Section 2 briefly describes three important courses in the curriculum and the lessons learned from WECS5 and WISE3. Section 3 describes the lab – the equipment and layout, and describes some of the exercises used; they are still evolving. Finally section 4 presents conclusions.

2. THREE IMPORTANT COURSES

The CSIA curriculum [3] is built around a three-course sequence: CS457 Computer Security I, CS458 Computer Security II, and CS459 Computer Security III.

The course Computer Security I covers the many facets of computer security and information assurance. It explores the security organization and infrastructure within an organization along with its policies, standards and procedures. Cryptographic protocols and algorithms (like DES, AES, RSA, Kerberos, digital signatures, etc.) are covered. This course helps lay the foundation of computer security and information assurance.

The course Computer Security II covers techniques and principles of design and configuration of secure workstations, servers and LANs. Topics like user and password management, system logging, intrusion detection techniques, etc. are presented. The course also covers the basics of virtual private networks, routers, firewalls and their implementation.

The course Computer Security III is a hands-on study of the threats to computer systems connected to the Internet. The overall objective is to understand how crackers find a system, find vulnerabilities in that system, and use a vulnerability to compromise the system, including the use of viruses. Various tools to attack and defend a system are studied. This is the course that the author teaches and the students spend a large percentage of the term in the lab. It is therefore necessary to design appropriate lab and hands-on exercises using tools that will help make this course an interesting experience.

2.1 Useful lessons from WECS5 and WISE3

The presentations from WECS5 and WISE3 provided very useful materials for those intending to offer courses in CSIA. In the case of the CSIA curriculum at SOU, many of the topics presented were already included in the CS457 and CS458 courses (e.g., cryptography, common criteria, firewalls, IDSs, etc.). This section lists those topics included in the course that the author teaches – CS459 Computer Security III. First, the topics from WECS5 included in the course are listed along with the approximate number of lecture and lab hours devoted to that

topic. Many of these topics use the lab to gain hands-on experience. Detailed lab exercises are described in a later section.

- a. Passwords and Password Cracking (2): Students are presented with the parameters of good password – use of special characters and longer lengths. These strengthen the passwords and make them less vulnerable to the cracking algorithms. A lab on using the password cracker, John the Ripper, helps illustrate the points.
- b. Digital certificates and key exchange (1): Students are shown how a digital certificate facilitates in the exchange of keys to accomplish symmetric encryption. The case of a web browser and a secure server at an online store are taken to illustrate these concepts.
- c. Covert Channels (1): Students are introduced to the technique of using a high level program to modulate a resource that a low level program can detect thereby circumventing the access rules and effecting a transfer of classified or other protected information. Students are required to write programs that conduct such “covert” communications.
- d. Malicious Software and Attacks (4): Students are introduced to the world of viruses – their creation, release, and detection. A lab exercise on creating a simple malicious program is used to illustrate the concept.
- e. Packet Sniffing and Spoofing (4): Students learn how packet sniffers are used to gather information passing between computers and how spoofing can help attackers gain access or even hijack entire sessions. IP spoofing, ARP spoofing, and DNS spoofing are discussed.
- f. Steganography (2): Students learn how information can be hidden in other information, for example images and web sites. Student work on exercises using the OutGuess, Stegdetect, and Stegbreak [8].
- g. Vulnerability Assessment and Penetration via the WWW (4): A series of lab exercises devoted toward understanding ways of exploiting vulnerabilities in Internet applications to gain access to a system. Some of the vulnerabilities covered are buffer overflows and WWW server and application-based attacks.

Next, the papers presented at WISE3 [4] that provided useful material for inclusion in the CS459 course are listed. Not all of these have been incorporated into the curriculum as yet but it is the author’s strong desire to do so in the next year.

- a. Teaching Network Security Through Live Exercises by G. Vigna. This paper presented an interesting tool that can be used in a class on computer security to test the students’ understanding of the concepts and their being to use them in an exercise against other teams. Such an exercise immediately generates excitement and enthusiasm in students.
- b. Design of a Laboratory for Information Security Education by V. Anantpadmanbhan, et al. This paper provides details on how a laboratory may be designed that will aid in the instruction of CSIA topics. They present some very interesting ideas for building a reconfigurable laboratory.
- c. Information Security Fundamentals by P. Oscarson. This paper provides graphical representations to illustrate security concepts like threats, incidents, security mechanisms, vulnerability, and risk. Such a graphical tool can be very useful when explaining how the fundamental pillars of security – confidentiality, integrity, and availability, can become compromised.

3. LAB DESCRIPTION AND EXERCISES

3.1 The Computer Security and Network Lab

The computer security lab, whose figure is shown in [9], is built using PCs running Windows XP and Linux, and equipment donated by Cisco (firewalls, routers, and switches). The objective behind the design of the lab was to create one that would model a real enterprise network – subnets, routers, firewalls, DNS server, Internet access, etc. would be a part of such a network. The lab therefore serves the needs of the CS459 course as well as the Unix systems administration, and introductory and advanced computer networking courses. The machines have both Windows and Linux operating systems installed on them. In addition to the equipment described, there are machines that serve classes on computer forensics and high performance computing.

Such an arrangement presents some challenges:

- Many of the classes have more students than computers and that means students have to share workstations.
- Some exercises (like password cracking, DNS cache setup, etc.) require system level access to the machines. Hence the system password needs to be constantly updated to prevent malicious use of the system. Also, many systems go into a “broken” state after the completion of certain exercises and so they have to be re-imaged after every lab.
- Some exercises run programs (like recompiling a kernel) that take more than the class time to complete. It is necessary to warn the students in advance of such an exercise so some of them can make time to stay back and finish the exercise.
- Staffing is necessary for the maintenance of the lab. Such staffing is currently provided through volunteer students eager for experience in system administration.

3.2 Lab Exercises and Projects

Lab exercises serve two objectives: to set aside one class period to give students a chance to experiment with the matter covered while in class, and to extend the exercise as an assignment. Here is a description of some of the lab exercises used in CS459 course, not already described before.

Using the whois databases to gather information about systems: The objective of this exercise is to show students how and where the .com, .org, .net, etc. domain names are registered. The exercise shows students how to query these databases, look for network administrators and find the IP addresses assigned to an organization. Attackers use these databases as a starting point for gaining access to systems by gathering information about them. Students realized how easy it was to get information about users and organizations on the Internet. Once this information is obtained, it is possible to run port-scanning tools on those systems to gather which ports are open. This scanning exercise was carried out on computers inside the lab.

Using password analysis tools like John the Ripper to identify bad passwords: Bad or default passwords are the basis of many methods used to gain access to a system. The objective of this exercise is to show students how easy it is to crack bad passwords and to motivate them to use better passwords. It also helps students wanting to be system administrators understand the importance of formulating policies on good passwords. The challenge is to use systems without real users and yet have a sufficient number of passwords as in an actual system. The program takes a very long time to run. There were no real accounts on the systems and so a few test accounts were created. The tool was able to crack the passwords of the test accounts in a relatively short time.

Using a packet sniffer (like ettercap, snort, etc.) to analyze network traffic: Attackers use packet sniffers to look at network traffic; this way they gather, among other things, userIDs and passwords as they are being passed between systems. Packet sniffers are also used for legitimate reasons to diagnose network problems. The objective of this exercise is to show students how computers exchange packets (e.g., TCP handshake), the structure of a packet and how information is encased in them. This assignment also helps in explaining spoofing and session hijacking concepts. Students realized how unsafe telnet and ftp utilities were when they saw their user names and passwords being transferred unencrypted over the network.

Recreating a virus: The objective of this exercise is to understand the anatomy of a virus attack. A simple virus (like Melissa) is created in the controlled environment and propagated. The students learn how viruses are written, how they are propagated via mediums like email, how they access system resources once in a system, and how to disinfect them. This exercise provides students with an excellent hands-on look at the way a virus works. The matter learned at WECS5 is augmented by material from the book on Viruses Revealed [5].

Web-based intrusions: Students are presented with a sample web site like the one given in the book on Web Hacking [6]. This exercise is an attempt to help students understand the vulnerabilities in web servers and applications. One of the exercises is modeled after a hands-on exercise at WECS5 in which the participants were able to deface a web site by exploiting vulnerability in the IIS web server. The exercises are used to introduce such concepts as buffer overflows, understanding web URLs and use of hidden variables, SQL injection attacks, Java remote executions, and session hijacking. Various small web applications are used to demonstrate techniques used to break into a web application.

Students also work on a project as a requirement for the course. The project requires students to perform research in two parts: known exploits and cyber laws. The first part requires students to lookup some recent examples of intrusions and viruses (in the case of this term, the Sasser worm had gained popularity) and then to document that exploit or virus – how it worked and what can be done to patch the vulnerability. Such an exercise will prepare students who will go on to become system administrators to handle such situations. The second part is an attempt to understand current and pending state and federal laws that address black hat activities and crimes.

In the last week of classes a contest is organized that has students capture one or more flags by breaking into a given target system. The class is divided into teams and the team that captures the most flags wins. This project is based on the CTF contest described in [7]. A target server is created that runs 6 services with known vulnerabilities. Since the students in the class are undergraduates they are told about the services and the vulnerabilities so that they may be able to actually perform the break-ins. Also, an entire day is given for this exercise.

4. CONCLUSIONS AND OBSERVATIONS

Designing a hands-on course that covers the important aspects of computer security is a challenging proposition and this paper has attempted to show one approach. In general, undergraduates require more handholding than graduate students. Hence a hands-on approach will require more instructor involvement. As a result some exercises will take longer than the class period to complete. Having an administrator to help with setting up the lab environment is crucial to the success of a hands-on approach. Most programs for the Unix platform are available as source distributions containing makefiles and configuration file that often need to be edited. Hence it is also important that students be familiar with downloading, configuring, building, and installing software.

Also, in the newer operating systems, many of the famous weaknesses have been strengthened. As a result recreating, implementing, or experimenting with those vulnerabilities will not succeed. For example, implementing a stack overflow as described in [10] will not work

since current operating systems have implemented a stack guard. As another example, most new operating systems do not provide a telnet daemon. As a result, demonstrating session hijacking becomes difficult. As yet another example, some tools like Dsniff, will not install on new operating systems, as they need an older library that is no longer available. The conclusion to be drawn from these experiences is to use an older release of an operating system (e.g., RedHat Linux 6.2) to teach such a class.

Another issue that the author has faced in this class is bimodal students; there are almost always a few students who are very knowledgeable in this field and can overwhelm the other students. In the past the author has given such students the option to test out of the course. On the upside student enthusiasm is very high in a course like this since it involves hands-on work with material that is very current and relevant. Many students use the knowledge to test their own or their relatives' systems, only to discover the many holes that are present in them.

5. REFERENCES

1. White house paper on securing cyberspace: <http://www.whitehouse.gov/pcipb/cyberstrategy-draft.html>
2. SFS program: <http://www.ehr.nsf.gov/ehr/DUE/programs/sfs/>
3. SOU's CSIA website: <http://www.csia.sou.edu>
4. C. Irvine and H. Armstrong (editors), Security Educational and Critical Infrastructures, Kluwer Academic Publishers, 2003
5. C.D. Harley et al. (authors), Viruses Revealed, McGraw-Hill Osborne, 2001
6. S. McClure et al. (authors), Web Hacking: Attacks and Defense, Addison-Wesley, 2002
7. Capture the Flag: <http://www.cs.ucsb.edu/~vigna/CTF/>
8. Steganography resources: <http://www.outguess.org/>
9. Computer network and security lab: <http://www.priscilla.com/cis336/lab.htm>
10. Aleph One (author), Smashing the Stack for Fun and Profit: <http://www.phrack.org/show.php?p=49&a=14>