APPENDIX A

PROJECTS AND OTHER STUDENT EXERCISES FOR TEACHING COMPUTER SECURITY

Many instructors believe that research or implementation projects are crucial to the clear understanding of computer security. Without projects, it may be difficult for students to grasp some of the basic concepts and interactions among security functions. Projects reinforce the concepts introduced in the book, give the student a greater appreciation of how a cryptographic algorithm or security function works, and can motivate students and give them confidence that they are capable of not only understanding but implementing the details of a security capability.

In this text, we have tried to present the concepts of computer security as clearly as possible and have provided numerous homework problems to reinforce those concepts. However, many instructors will wish to supplement this material with projects. This appendix provides some guidance in that regard and describes support material available in the **Instructor's Resource Center (IRC)** for this book, accessible from Pearson for instructors. The support material covers 11 types of projects and other student exercises:

- Hacking projects
- Laboratory exercise
- Security education (SEED) projects
- Research projects
- Programming projects
- Practical security assessments
- Firewall projects
- Case studies
- Reading/report assignments
- Writing assignments
- Webcasts for teaching computer security

A.1 HACKING PROJECT

The aim of this project is to hack into a corporation's network through a series of steps. The corporation is named Extreme In Security Corporation. As the name indicates, the corporation has some security holes in it and a clever hacker is able to access critical information by hacking into its network. The IRC includes what is needed to set up the Website. The student's goal is to capture the secret

information about the price on the quote the corporation is placing next week to obtain a contract for a governmental project.

The student should start at the Website and find his or her way into the network. At each step, if the student succeeds, there are indications as to how to proceed on to the next step as well as the grade until that point.

The project can be attempted in three ways:

1. Without seeking any sort of help
2. Using some provided hints
3. Using exact directions

The IRC includes the files needed for this project:

1. Web Security project named extremeinsecure (extremeinsecure.zip)
2. Web Hacking exercises (XSS and Script-attacks) covering client-side and server-side vulnerability exploitations respectively (webhacking.zip)
3. Documentation for installation and use for the above (description.doc)
4. A PowerPoint file describing Web hacking (Web_Security.ppt). This file is crucial to understanding how to use the exercises, since it clearly explains the operation using screen shots.

This project was designed and implemented by Professor Sreekanth Malladi of Dakota State University.

A.2 LABORATORY EXERCISES

Professor Sanjay Rao and Ruben Torres of Purdue University have prepared a set of laboratory exercises that are part of the IRC. These are implementation projects designed to be programmed on Linux, but could be adapted for any UNIX environment. These laboratory exercises provide realistic experience in implementing security functions and applications.

A.3 SECURITY EDUCATION (SEED) PROJECTS

The SEED projects are a set of hands-on exercises, or labs, covering a wide range of security topics. They were designed by Professor Wenliang Du of Syracuse University for use by other instructors [DU11]. The SEED lab exercises are designed so no dedicated physical laboratory is needed. All SEED labs can be carried out on students' personal computers or in a general computing laboratory. The collection consists of three types of lab exercises:

- **Vulnerability and attack labs:** These 12 labs cover many common vulnerabilities and attacks. In each lab, students are given a system (or program) with hidden vulnerabilities. Based upon the hints provided, students must find these vulnerabilities, then devise strategies to exploit them. Students also need to

demonstrate ways to defend against the attacks or comment on the prevailing mitigating methods and their effectiveness.

- **Exploration labs:** The objective of these 9 labs is to enhance students' learning via observation, playing, and exploration, so they can understand what security principles feel like in a real system; and to provide students with opportunities to apply security principles in analyzing and evaluating systems.
- **Design and implementation labs:** In security education, students should also be given opportunities to apply security principles in designing and implementing systems. The challenge is to design meaningful assignments that do not require a major commitment of time. The 9 labs in this category meet this requirement.

Table A.1 maps the 30 lab exercises in the SEED repertoire to the relevant chapters in the book, together with an estimate of the number of weeks required for the typical student to complete a lab, assuming about 10 hours per week devoted to the task.

Table A.1 Mapping of SEED Labs to Textbook Chapters

Types	Labs	Time (weeks)	Chapters
Vulnerability and Attack Labs (Linux-based)	Buffer Overflow Vulnerability	1	10
	Return-to-libc Attack	1	10
	Format String Vulnerability	1	11
	Race Condition Vulnerability	1	11
	Set-UID Program Vulnerability	1	11
	Chroot Sandbox Vulnerability	1	12
	Cross-Site Request Forgery Attack	1	11
	Cross-Site Scripting Attack	1	11
	SQL Injection Attack	1	5
	Clickjacking Attack	1	6
	TCP/IP Attacks	2	7, 22
	DNS Pharming Attacks	2	22
Exploration Labs (Linux-based)	Pack Sniffing & Spoofing	1	22
	Pluggable Authentication Module	1	3
	Web Access Control	1	4, 6
	SYN Cookie	1	7, 22
	Linux Capability-Based Access Control	1	4, 12
	Secret-Key Encryption	1	20
	One-Way Hash Function	1	21
	Public-Key Infrastructure	1	21, 23
	Linux Firewall Exploration	1	9

(Continued)

Table A.1 *(Continued)*

Types	Labs	Time (weeks)	Chapters
Design and Implementation Labs	Virtual Private Network (Linux)	4	22
	IPsec (Minix)	4	22
	Firewall (Linux)	2	9
	Firewall (Minix)	2	9
	Role-Based Access Control (Minix)	4	4
	Capability-Based Access Control (Minix)	3	4
	Encrypted File System (Minix)	4	12
	Address Space Randomization (Minix)	2	12
	Set-Random UID Sandbox (Minix)	1	12

A Webpage accessible through the Companion Website at williamstallings .com/ComputerSecurity (Instructor Resources link) provides links to all the labs, organized by chapter. Each lab includes student instructions, relevant documents, and any software needed to perform the lab. In addition, a link is provided for instructors to enable them to obtain the instructor manual.

A.4 RESEARCH PROJECTS

An effective way of reinforcing basic concepts from the course and for teaching students research skills is to assign a research project. Such a project could involve a literature search as well as an Internet search of vendor products, research lab activities, and standardization efforts. Projects could be assigned to teams or, for smaller projects, to individuals. In any case, it is best to require some sort of project proposal early in the term, giving the instructor time to evaluate the proposal for appropriate topic and appropriate level of effort. Student handouts for research projects should include:

- A format for the proposal
- A format for the final report
- A schedule with intermediate and final deadlines
- A list of possible project topics

The students can select one of the topics listed in the IRC or devise their own comparable project. The instructor's supplement includes a suggested format for the proposal and final report as well as a list of possible research topics.

The following individuals have supplied the research and programming projects suggested in the instructor's supplement: Henning Schulzrinne of Columbia University; Cetin Kaya Koc of Oregon State University; David M. Balenson of Trusted Information Systems and George Washington University; Dan Wallach of Rice University; and David Evans of the University of Virginia.

A.5 PROGRAMMING PROJECTS

The programming project is a useful pedagogical tool. There are several attractive features of stand-alone programming projects that are not part of an existing security facility:

1. The instructor can choose from a wide variety of cryptography and computer security concepts to assign projects.
2. The projects can be programmed by the students on any available computer and in any appropriate language; they are platform- and language-independent.
3. The instructor need not download, install, and configure any particular infrastructure for stand-alone projects.

There is also flexibility in the size of projects. Larger projects give students more a sense of achievement, but students with less ability or fewer organizational skills can be left behind. Larger projects usually elicit more overall effort from the best students. Smaller projects can have a higher concepts-to-code ratio, and because more of them can be assigned, the opportunity exists to address a variety of different areas.

Again, as with research projects, the students should first submit a proposal. The student handout should include the same elements listed in the preceding section. The IRC includes a set of 12 possible programming projects.

The following individuals have supplied the research and programming projects suggested in the IRC: Henning Schulzrinne of Columbia University; Cetin Kaya Koc of Oregon State University; and David M. Balenson of Trusted Information Systems and George Washington University.

A.6 PRACTICAL SECURITY ASSESSMENTS

Examining the current infrastructure and practices of an existing organization is one of the best ways of developing skills in assessing its security posture. The IRC contains a description of the tasks needed to conduct a security assessment. Students, working either individually or in small groups, select a suitable small- to medium-sized organization. They then interview some key personnel in that organization to conduct a suitable selection of security risk assessment and review tasks as it relates to the organization's IT infrastructure and practices. As a result, they can then recommend suitable changes, which can improve the organization's IT security. These activities help students develop an appreciation of current security practices, and the skills needed to review these and recommend changes.

A.7 FIREWALL PROJECTS

The implementation of network firewalls can be a difficult concept for students to grasp initially. The IRC includes Network Firewall Visualization tool to convey and teach network security and firewall configuration. This tool is intended to teach

and reinforce key concepts including the use and purpose of a perimeter firewall, the use of separated subnets, the purposes behind packet filtering, and the shortcomings of a simple packet filter firewall.

The IRC includes a .jar file that is fully portable, and a series of exercises. The tool and exercises were developed at U.S. Air Force Academy.

A.8 CASE STUDIES

Teaching with case studies engages students in active learning. The IRC includes case studies in the following areas:

- Disaster recovery
- Firewalls
- Incidence response
- Physical security
- Risk
- Security policy
- Virtualization

Each case study includes learning objectives, case description, and a series of case discussion questions. Each case study is based on real-world situations and includes papers or reports describing the case.

The case studies were developed at North Carolina A&T State University.

A.9 READING/REPORT ASSIGNMENTS

Another excellent way to reinforce concepts from the course and to give students research experience is to assign papers from the literature to be read and analyzed. The IRC includes a suggested list of papers to be assigned, organized by chapter. The Premium Content Website provides a copy of each of the papers. The IRC also includes a suggested assignment wording.

A.10 WRITING ASSIGNMENTS

Writing assignments can have a powerful multiplier effect in the learning process in a technical discipline such as computer security. Adherents of the Writing Across the Curriculum (WAC) movement (http://wac.colostate.edu/) report substantial benefits of writing assignments in facilitating learning. Writing assignments lead to more detailed and complete thinking about a particular topic. In addition, writing assignments help to overcome the tendency of students to pursue a subject with a minimum of personal engagement, just learning facts and problem-solving techniques without obtaining a deep understanding of the subject matter.

The IRC contains a number of suggested writing assignments, organized by chapter. Instructors may ultimately find that this is the most important part of their

approach to teaching the material. We would greatly appreciate any feedback on this area and any suggestions for additional writing assignments.

A.11 WEBCASTS FOR TEACHING COMPUTER SECURITY

The Companion Website at williamstallings.com/ComputerSecurity (Instructor Resources link) provides a link to a catalog of webcast sites that can be used to enhance the course. An effective way of using this catalog is to select, or allow the student to select, one or a few videos to watch, then assign the student to write a report/analysis of the video.