Evaluation of the Offensive Approach in Information Security Education

Martin Mink, Rainer Greifeneder

Evaluation of the Offensive Approach in Information Security Education

Martin Mink[1] and Rainer Greifeneder[2]

[1] Technical University of Darmstadt, Germany
[2] University of Mannheim, Germany

Abstract. There is a tendency in information security education at universities to not only teach protection measures but also attack techniques. Increasingly more universities offer hands-on labs, where students can experience both the attackers' and the administrators' view. Getting to know the attackers' view is thought to lead to a better understanding of information security and its problems compared to teaching only strategies for defense.
This paper analyzes the situation of information security education at German and international universities. We present a method to measure knowledge in information security and – using this method in an empirical study – evaluate the offensive teaching approach. Analysis of the empirical data gathered in the study shows a tendency in favor of the offensive approach compared to the classic defensive security education.

1 Introduction

The field of academic security education today is dominated by defensive techniques like cryptography, firewalls, access control, and intrusion detection. But since some years we are observing a trend toward more offensive methods [19, 16]. In the academic literature, offensive techniques are also gaining widespread approval [2, 8, 1]. The ACM even devoted an entire special issue of their flagship publication Communications to the topic of "Hacking and Innovation" [6].

Why is this so? In his article [5], Conti argues that security academics can learn a lot from the security approach of hackers by visiting their gatherings (like DEF CON [7] or Black Hat [3]). This corresponds to the professional trend toward more offensive methods of *security testing* and its most prominent variant of *penetration testing*. This involves the use of hacking tools like network sniffers, password crackers and disassemblers as well as active penetrations of corporate networks in real time. Looking at these indications, there seems to be a substantial benefit from thinking security in an offensive way. But is there really a benefit? And if yes, can it in some way be quantified?

In this paper we show a way how to answer this question. We present an experimental setup to evaluate the offensive approach in information security education. The basic idea of the approach used in this paper has already been introduced in [14]. We conduct an empirical study that compares the offensive with the classic defensive approach in information security education. As part

of the study we designed two courses on information security. The study will show, that there is some advantage of the offensive approach but that the result is not significant. Before, we'll take a closer look on offensive education at the university degree level and give an overview of teaching methods for offensive education.

The next section introduces background knowledge: a classification of information security courses, a definition of offensive methods, an introduction into empirical research, and related work. Section 3 presents the conducted empirical study, and Sec. 4 the results of the study. We conclude in Sec. 5.

2 Background

This section introduces knowledge that is needed for this paper.

2.1 Classification of Information Security Courses

In a study, *introductory* courses on information security at German and international universities were analyzed and classified by their content [12]. The classification yielded three clusters: one large and two small ones. One of the small clusters contains courses that focus on teaching cryptography and thus was called the "conservative" cluster. The other small cluster focuses on teaching current topics of information security, thus called the "innovative" cluster. In the large cluster most topics of the two other clusters are taught in a balanced mixture, the "balanced" cluster.

2.2 The Offensive Approach in Information Security Education

What is an "offensive" method? In general, a method that is used by an attacker. But this would not be enough to differentiate between offensive and defensive techniques, since some are used both by attackers and by administrators (as network sniffing and password cracking). Therefore the results of the before mentioned classification of information security courses was used as an enumerating definition: the innovative cluster was identified with the offensive approach and the conservative cluster with the defensive approach. This way the topics of each cluster were associated with the method of the respective approach.

Types of Offensive Education *Information security labs* give students the chance to gain hands-on experience in information security. So called *Wargames* and *Capture-the-Flag* contests offer a game-like approach. These will be presented in more detail in the following paragraphs.

Several universities run a security lab, where students can experience both aspects of IT security and can get hands-on experience. In Germany, the computer science department of Darmstadt University of Technology pioneered in 1999 with the so-called *Hacker Contest* [16]. In military education, one can find similar examples of offensive lectures, for example [20].

Wargames have a long tradition among security enthusiasts. In Wargames the organizer creates a set of challenging scenarios of increasing difficulty which have to be solved by the participants [18, 9]. Challenges usually are modeled somewhat after the problems an attacker faces when attempting a system penetration. More competitive than Wargames are *Capture-the-Flag* (CTF) contests, where teams battle against each other over the control of a network. In those contests, 20 to 40 teams of educational institutions spread across the world battle against each other over the course of several hours [4, 10].

Criticism It is often criticized that offensive methods should not be taught to students since this only increases the population of "malicious hackers" which will not raise but rather decrease the overall level of security in the Internet. We feel that this line of argument is flawed. Any security technique can be simultaneously used and abused. The trend toward penetration testing in corporate businesses shows that offensive techniques can be used to increase the level of security of an enterprise. So students trained in offensive techniques must not necessarily become *black hats* (jargon for malicious hackers, the "bad guys"), but rather can also become *white hats* (the good guys). However, we agree that offensive techniques should not be taught in a standalone fashion. As with defensive techniques, every course in information security should be accompanied by a basic discussion of legal implications and ethics.

2.3 Empirical Methods

To have a common basis and for those not familiar with empirical research, we present a short overview of the methods used in empirical studies relevant for this case. A study starts with a *hypothesis* which expresses what lies in the interest of the researcher. The hypothesis can be a *one-tailed* (or *directional*) hypothesis, i.e. the direction of a possible difference is specified, or *two-tailed*, i.e. direction is not specified. In either case it suspects a link between at least two variables, which can be expressed as "if ..., then ...". A *variable* in a study is any characteristic that can assume multiple values or can vary in participants (e.g. variable $gender = \{male, female\}$). A hypothesis expresses the assumption that an *independent variable* (IV) affects a *dependent variable* (DV). Two concepts are critical to evaluate empirical research: *internal validity* and the *external validity*: a study has a high internal validity, if the variable the researcher intended to study is indeed the one affecting the results and not some other, unwanted variables. External validity refers to the extent to which the results of a study can be generalized or extended to others. Both concepts strongly depend on whether the investigation is conducted as a field versus laboratory experiment, and with a true experimental vs. quasi-experimental design. A *field experiment* is an experiment that is conducted in a natural environment, while a *laboratory experiment* is conducted in an environment that offers control over unwanted influences. In general, a field experiment has a lower internal validity than a laboratory experiment (since it is difficult to control confounding variables that can

substantially affect the result) but a higher external validity (since it normally is closer to real life). A study is *quasi-experimental* when study groups are determined based on a pre-existing characterisitc (e.g., gender, assignment to one or the other school class, etc.). In contrast, a study is *experimental* when participants are randomly assigned to study groups. See the book [17] for detailed information on the subject.

2.4 Related Work

Näf and Basin [15] compare two approaches for information security labs: the conflict-based and the observation-based approach as they call it. But they only do it on a conceptual level.

Vigna [19] reports on his experiences with courses where students gain hands-on experience with attack and defense. Students first work in two teams: red team (the attacker) and blue team (the defender). Next, both teams attack and defend, and in the last part both teams work in rivalry to suceed in a targeted attack first.

In 1997 Jonsson and Olovsson [11] conducted an experiment to evaluate the actions of an attacker trying to compromise a system. A number of independent teams of students worked as attackers and their actions were logged to evaluate – among other – the time that passed between two attacks, time that was spend preparing and the time spend on the attack.

Although these studies deal with attacking techniques, none of them actually assesses the *value* of the offensive approach in information security education.

3 Methods

3.1 Design of the Empirical Study

To evaluate the benefits of offensive techniques it's better to not take a one shot case study, where just the effect of this course is measured, since this offers only low internal validity. Instead, we compare the treatment to a group who received a classic defensive education in information security.

We therefore postulate:

> Students who received offensive information security education have a better understanding of IT security than those who received defensive education.

This hypothesis is a difference hypothesis and implies two treatment groups: one with offensive education, the other with defensive education, thus leading to a two-group design. The independent variable is the type of education" ("offensive" or defensive"), as dependent variable we assess "understanding of IT security". To ensure a high level of internal validity, participants in the present empirical study were randomly assigned to experimental groups in a controlled two-group treatment design.

A second independent variable was added to the study to find out if the prior knowledge in information security of the subjects is relevant. This second independent variable was chosen to be two-leveled ("high prior knowledge" and "low prior knowledge"). The result is a 2x2 factorial design, leading to four sample groups. To select the subjects according to prior knowledge, a knowledge test is used (see next section).

Tests For the study three tests were designed:

1. a test to assess the knowledge of information security at the end of the courses and that is used as the main measuring instrument,
2. a knowledge test and
3. an awareness test.

Test no. 1 is used for measuring the main dependent variable (DV) by confronting the students with a computer system that is configured with a number of security holes and vulnerabilities. They are each asked to identify and fix the vulnerabilities and to configure the system in a secure way. To assess their understanding of information security, the number of vulnerabilities found, the time used (relative to the result achieved), and the strategy used are measured (see Fig. 1). The test is identical for both groups and does not test for offensive/defensive knowledge but for the understanding of system security.

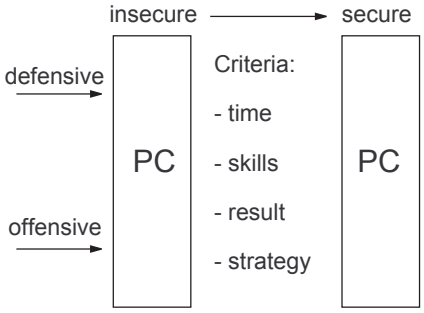

Fig. 1. Experimental setup

Tests no. 2 and 3 are paper-pencil-questionnaires and applied at different time points during the course. The purpose of the knowledge test is to measure the *increase* in knowledge as a function of the two treatment groups. The items (questions) of the test were chosen to cover a wide range of aspects of IT security so as to get a representative cross section of the field. Based on individual results in the knowledge test, participants were divided into two groups: those with high prior knowledge and those with low prior knowledge. The awareness test was included for a purpose unrelated to the present hypothesis. Both tests were tested on sample groups.

3.2 Design of the Information Security Courses

For the empirical study two courses were designed: one offensive, and one defensive. To increase experimental control and reduce common threats to internal validity, three day crash courses –instead of semester long courses – were chosen. The courses were designed to have nine modules (three each day), each lasting two hours. In order to have a common basis, the first module is an introduction into programming, networks and system administration. The last module is the test for the evaluation of the offensive approach. For the remaining seven modules we chose the topics as shown in Table 1.

Day 1	Day 2	Day 3
1. Introduction	**4. Network security 1**	**7. Web security**
Ethics	Network sniffing	SQL injection
Working with Linux	Port scanning	XSS
Programming in C		Vulnerabilities
Networks		
2. Unix security	**5. Network security 2**	**8. Malware**
Password security	Spoofing	Viruses
Access control	TCP Hijacking	Worms
Detecting intrusions	DoS attacks	Trojan horses
	SSH	Rootkits
3. Software security	**6. Firewalls**	**Test**
Buffer Overflows	Concept	
Format Strings	Architecture	
Race Conditions	Configuration	

Table 1. Overview of the course

Each module consists of an introduction into the topic by means of a presentation lasting around 30 minutes, followed by about 60 minutes of hands-on work using exercises on a hand-out sheet. Each module ends with a plenary discussion of the exercises. The distincion between the offensive and the defensive approach is only made in the practial part, the theoretical part is identical for both groups.

For the exercises Linux is used since it is freeley available, is well documented and offers the possibility to configure the system and software using configuration files (i.e. without the need to use graphical interfaces). The last point is a basic concept of the course and thought to lead to a better understanding – and also independent of the Linux distribution resp. OS used. Each module will be presented in more detail in the following sections.

Module 1: Introduction The introduction presents working with the Linux operating system, basics of networks and programming in C. In the practical part no difference is made between the offensive and the defensive course.

Module 2: OS Security This module deals with security relevant aspects of UNIX systems. It is about the problems of passwords (and what secure passwords are), about access control, including the problem of SUID root executables, and about detecting signs of intrusions. The offensive group concentrates on working with password cracking tools and how to conver tracks of an intrusions while the defensive group deals with password policies, how to create a password that is difficult to break and the search for signs of intrusions.

Module 3: Software Security After an introduction into the relevant parts of memory management and layout of processes on Intel architecture, the most common security problems in software are presented: buffer overflows, format strings and race conditions. In the exercises these problems can be experienced, the offensive group concentration on detecting vulnerabilities and exploiting them, the defensive group detecting and fixing them.

Module 4: Network Security 1 This first part of network security introduces network sniffing and port and vulnerability scanning. Students learn about sniffing protocols of the TCP/IP stack with and without encryption as well as the basics about port scanning. The offensive group learns that sniffing is also possible in switched networks by means of ARP spoofing whereras the defensive groups learns how to detect an active sniffer in a network.

Module 5: Network Security 2 The second part of network security deals with spoofing (ARP, IP, Mail, DNS), with configuring an SSH server, and denial-of-service (DoS) attacks. In the hands-on part the offensive group learns about MAC and ARP spoofing and advanced forms of sniffing and scanning. The defensive group configures an SSH server, applies the vulnerability scanner Nessus and configures an e-mail client to send encrypted e-mails.

Module 6: Firewalls In the module firewalls, the different types of firewall architectures are presented. By example of the packet filter firewall iptables the participants learn how to configure a personal firewall for a given scenario. The defensive group tests the firewall and the offensive group tries to evade the firewall.

Module 7: Web Security The participants are introduced to a selection of common attacks on web applications (e.g. SQL injection, XSS, path traversal). In the practical part both goups work on exploiting a sample web application.

Module 8: Malware This module deals with the different types of malware (virus, worm, trojan horse) and a detailed look on rootkits. The exercise is differentiated by the offensive group taking a closer look on the workings of rootkits and the defensive group searching for the presence of rootkits.

Module 9: Test The last module of the courses was the main measuring instrument of the study. It was designed as a practical test and as an application of what the subjects had learned during the course. The same Linux system as the one used in the first eight modules was set up with (1) ten security holes and misconfigurations. Additionally, the subjects were asked to work on (2) two tasks.

(1) The security holes and misconfigurations were as follows (from now on called "vulnerabilities"):

1. A rootkit (that turns out to be a fake)
2. An additional root user
3. Unwanted entries in the sudo configuration file
4. Security relevant SUID root executables (the text editor *nano* and the command *chmod*)
5. World readable and writeable log file and root user directories (/etc/logs and /root)
6. Weak passwords and password policy
7. More active services than allowed
8. An active network sniffer
9. Network interface in promiscuous mode
10. Errors in the firewall configuration

(2) The tasks were as follows:

1. Configuration of SSH server (public key authentication)
2. Configure sudo for a user

Differentiation In some modules it was rather difficult to make a distinction between the offensive and the defensive course since a lot of methods are applied by both an attacker and an administrator (e.g. password cracking). The definition for "offensive method" derived by the study presented in Sec. 2 in some cases was not fine-grained enough to allow a distinction. In the second run of the study we made some changes to achieve a better differentiation between offensive and defensive, e.g. password cracking only in the offensive course. Still, not in all cases we were able to make a clear distinction.

3.3 Conducting the Empirical Study

The study was conducted at two German universities: in 2007 at RWTH Aachen University and in 2008 at RWTH Aachen and at University of Mannheim. In 2008 69 subjects were included for analysis (out of about 80 registered for the courses) and 39 subjects in 2007. This paper will focus on and present the results of the 2008 study; see the PhD thesis by Mink [13] for the 2007 study.

The courses were anounced on a web page of the university and on mailing lists. To register for the courses, students were asked to online create an account on a web application by entering their data (including name, age, semester, and

a statement of motivation). The same system and account was to be used by the participants during the course to access the course ressources (i.e. down- and upload of material). The number of participants was limited by the size of the room used for the couses. Registrations were admitted until the maximum number of 24 per course was reached.

Since the director of the study and the instructor of the crash courses were the same person there is the danger of the so called Rosenthal effect. This effect describes the influence a biased director of studies might have on the way he conducts the study. But since the instructor actively only gave the theoretical part of the courses – which was identical for both groups – this effect should be negligible.

Knowledge and Awareness Test These two test were conducted in the form of an online questionnaire at the beginning and at the end of the course. About six weeks later the subjects were contacted by e-mail to fill in the questionnaire again (a so called retest). Because we wanted to link tests of three different time points while preventing social-desirable (instead of honest) responses, participants were asked to code each questionnaire with the same unique code.

Main Dependent Variable Test A maximum of 60 points were possible; for each vulnerability and task five points for an accurate solution, down to one for a less accurate one and zero for none at all. The participants were given 45 minutes for the test. They received an instruction sheet explaining that they were taking over the administration of a computer in an enterprise with only some knowledge about its configuration. To complete the test they were allowed to use the course material and the WWW.

Logging Different methods were used to log data for tracing the actions of the subjects: the content of the system logs (logged via network), screen capturing, contents of the /etc and the user directory, changes to files (using an automated script) and the command history.

4 Results

The empirical data gathered in the tests was analyzed using mainly SPSS. First, we present the results of the main DV test – the main measuring instrument of the study – and then the results of the knowledge and awareness tests.

4.1 Main Dependent Variable Test

Based on the logged protocols, we determined for each subject what was found analyzing the system and how the situation was solved. Two primary measures were assessed: the number of vulnerabilities found (see Fig. 2) and the overall score achieved (including the tasks, see Fig. 3). The diagrams show that the

participants of the offensive courses on average both found more vulnerabilities and achieved a better score than the participants of the defensive courses. Since the significance score in the ANOVA (analysis of variance) with a value of 0.03 is below 0.05 the result is significant. The error bars in the diagram show the intervals in which the result lies with a probability of 95% (confidence intervals). Although in both diagrams the error bars overlap, the result is still significant, since with 95 CI, error bars may overlap up to one fourth.

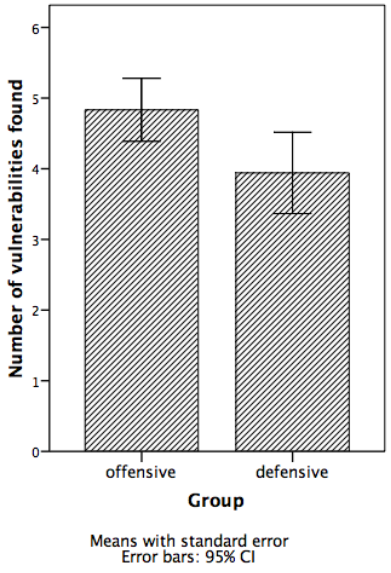

Fig. 2. Number of vulnerabilities found

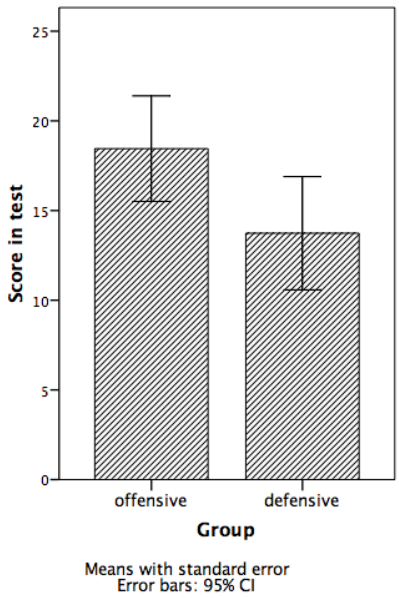

Fig. 3. Number of points in main DV test

4.2 Knowledge Test

Fig. 4 shows the result of the knowledge test at the beginning of the courses. Since the distribution of the subjects to the two groups was random, the score differs slightly between the offensive and the defensive group. The result of the knowledge test at the end of the courses is depicted by Fig. 5 (the result of the retest is not shown here). As can be observed, in both cases the defensive group achieves a slightly higher score than the offensive group, both in the low and in the high prior knowledge subgroup. This is a surprising result, because it contrasts to the hypthesis, but it is not significant. Since it also does not match prior results, most probably it is an effect at the chance level.

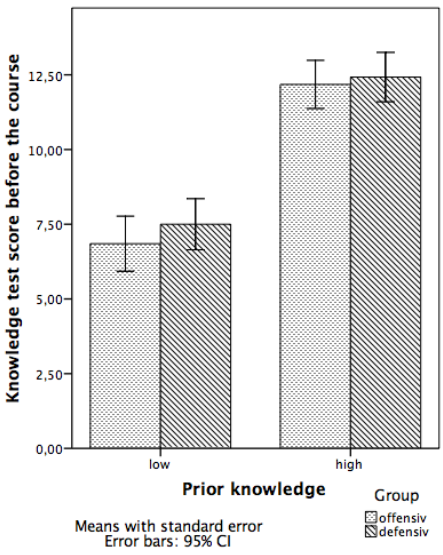

Fig. 4. Knowledge test at the beginning

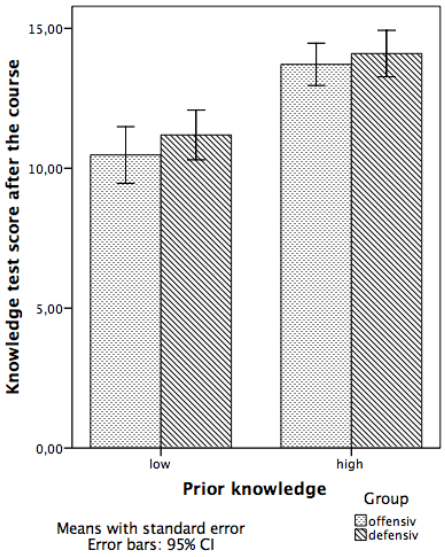

Fig. 5. Knowledge test at the end

5 Conclusion

In this paper we presented an experimental setup to assess different approaches of information security education at the university degree level. The setup was used to evaluate the offensive approach by comparing it to the classic, defensive approach. To this end, two courses in information security were designed and an empirical study was conducted. While the results of the knowledge test do not support the hypothesis, they show that the design is not flawed, since the subgroups with higher prior knowledge achieve a higher score – something to be expected. The results of the main measuring instrument, the main DV test, show an advantage of the offensive group over the defensive group that is significant. The advantages of the offensive over the defensive approach are, that it leads to a better understanding of information security and that it is more motivating. It is more motivation, because of its game-like approach (a higher fun factor), and because it is easier to discover a vulnerability than to prove that there are no vulnerabilities at all. As a consequence, information security courses should teach offensive aspects. The results of the classification of information security courses at universities (see Sec. 2.1) illustrate, that this is already a trend at universities, because the majority of the reviewed courses includes offensive techniques.

The presented setup can be used to repeat the study to gather more empirical data. In reruns of the study more subjects should be used. A still better separation of defensive and offensive methods in the crash courses might be achieved. One way could be to incorporate cryptography as a topic into the courses. The

main DV test as it is designed now is restricted to system administrator tasks and could be expanded to also include other – security relevant – topics.

The offensive approach presents itself as a valuable method in information security education. And there will always be a need for at least a small amount of experts with offensive experience, e.g. for AV industry, intelligence or law enforcement.

References

1. Iváan Arce and Gary McGraw. Why attacking systems is a good idea (Guest Editors' introduction). *IEEE Security & Privacy*, 2(4):17–19, July/August 2004.
2. Kirk P. Arnett and Mark B. Schmidt. Busting the ghost in the machine. *Communications of the ACM*, 48(8):92–95, August 2005.
3. Black Hat briefings, training and consulting. `http://www.blackhat.com`.
4. Homepage "CIPHER CTF". `http://www.cipher-ctf.org/`.
5. Gregory Conti. Why computer scientists should attend hacker conferences. *Communications of the ACM*, 48(3):23–24, March 2005.
6. Gregory Conti. Hacking and innovation (Guest Editors' introduction). *Communications of the ACM*, 49(6):33–36, June 2006.
7. DEF CON Hacking Event. `http://www.defcon.org`.
8. Dan Farmer and Wietse Venema. Improving the security of your site by breaking into it. Usenet Posting to comp.security.unix, 3. December 1993.
9. Homepage "Hack This Site". `http://www.hackthissite.org/missions/`.
10. Homepage "International Capture The Flag". `http://ictf.cs.ucsb.edu/`.
11. Erland Jonsson and Tomas Olovsson. A Quantitative Model of the Security Intrusion Process Based on Attacker Behavior. In *Transactions on Software Engineering*, volume 23, pages 235–245. IEEE, April 1997.
12. Christian Mertens. Wie lehrt man IT-Sicherheit am Besten – Übersicht, Klassifikation und Basismodule. Master's thesis, RWTH Aachen, 2007.
13. Martin Mink. *Vergleich von Lehransätzen für die Ausbildung in IT-Sicherheit*. PhD thesis, University of Mannheim, 2009.
14. Martin Mink and Felix C. Freiling. Is Attack Better Than Defense? Teaching Information Security the Right Way. In *Proceedings of the Conference on Information Security Curriculum Development (InfoSecCD)*, pages 44–48. ACM Press, 2006.
15. Michael Näf and David Basin. Conflict or review – two approaches to an information security laboratory. *Communications of the ACM*, 51(12):138–142, December 2008.
16. Markus Schumacher, Marie-Luise Moschgath, and Utz Roedig. Angewandte Informationssicherheit: Ein Hacker-Praktikum an Universitäten. *Informatik Spektrum*, 6(23), June 2000.
17. William R. Shadish, Thomas D. Cook, and Donald T. Campbell. *Experimental and Quasi-Experimental Designs for Generalized Causal Inference*. Cengage Learning, 2001.
18. Starfleet academy hackits. `http://isatcis.com/`.
19. Giovanni Vigna. Red team/blue team, capture the flag, and treasure hunt: Teaching network security through live exercises. In *World Conference on Information Security Education*, pages 3–18, 2003.
20. Gregory White and Gregory Nordstrom. Security across the curriculum: Using computer security to teach computer science principles. In *Proceedings of the 19th International Information Systems Security Conference*, pages 519–525, 1998.