Is Attack Better Than Defense? Teaching Information Security the Right Way

Martin Mink
RWTH Aachen University, Germany
Ahornstraße 55
52056 Aachen
mink@cs.rwth-aachen.de

Felix C. Freiling
University of Mannheim, Germany
A5, 6
68131 Mannheim
freiling@informatik.uni-mannheim.de

ABSTRACT
A recent trend in security education is towards teaching offensive techniques which were originally developed by hackers. This reflects tendencies in the professional world where offensive security testing (penetration testing) is quickly gathering widespread acceptance. We report on good experiences with a security curriculum at a university degree level which emphasizes offensive techniques over defensive ones. Our claim is that teaching offensive methods yields better security professionals than teaching defensive techniques alone. The paper presents an experimental setup with which we plan to investigate this claim further. The experimental setup uses concepts from psychology and pedagogical sciences to empirically assess the benefit of offensive teaching.

Categories and Subject Descriptors
K.3.2 [**Computers And Education**] - Computer and Information Science Education – *Curriculum, Information systems education.*

General Terms
Measurement, Security

Keywords
Security education, empirical measurement, offensive techniques, security lab, computer forensics, summerschool

1.INTRODUCTION
Motivation
The threats to the security of our information technology infrastructure are constantly changing. A recent report from the security front of the Internet [20] identifies a trend towards more sophistication and professionalism in the attacks. Hackers identify their targets with much more care than before and use the complete arsenal of weapons like trojan horses equipped with keyloggers to intrude corporate or government networks. On the defensive side, the trend towards more holistic approaches also continues. This means that technical solutions like firewalls and intrusion detection systems must be integrated into a security and risk management perspective. But rather surprisingly, Peter G. Neumann, the founder and long-year moderator of the well-known *Risks Digest* [15], does not include management as one of the eight most important challenges in IT security today [16]: Next to system development practice and privacy for example the issue of *security education* is also listed.

The field of academic security education today is dominated by defensive techniques like cryptography, firewalls, access control, and intrusion detection. But also here we are observing a recent trend towards more offensive methods [22,19]. In the academic literature, offensive techniques are also gaining widespread approval [4,12,3]. The ACM even devoted an entire special issue of their flagship publication *Communications* to the topic of "Hacking and Innovation" [7].

Why is this so? In a recent article, Conti [6] argues that security academics can learn a lot from the security approach of hackers by visiting their gatherings (like Defcon [2] or Black Hat [1]). This corresponds to the professional trend towards 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. Is there really a benefit? And if yes, can it in some way be quantified?

Contribution
From our experiences in teaching security to university students, we strongly feel that teaching offensive methods within the academic curriculum has a substantial advantage. Briefly spoken, we feel that more time should be devoted to attacking than to defending within university courses: attack is better than defense. However, this statement implies a basic research question: Can we quantify our hypothesis objectively?

In this paper we describe an experimental setup with which we plan to evaluate the hypothesis "attack better than defense". This implies a more precise definition of what "better" means. In our context, i.e., referring to the security education of students, "better" can be understood in multiple ways:

- a better understanding of the ways in which security systems fail,
- a faster time to solve security related tasks,
- a longer uptime of a system administrated, or
- a better ability to write secure programs.

This also implies a rigorous empirical method to evaluate the hypothesis. Here we strongly build upon widely accepted standards from the social sciences like psychology and pedagogical sciences.

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 towards 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 IT security should be accompanied by a basic discussion of legal implications and ethics.

Outline

Section 2 gives an overview on the IT security curriculum offered by the authors and lists related work. In Section 3 we present an approach to measure the effects of offensive teaching in comparison to defensive teaching. The paper concludes with Section 4.

2.Curriculum And Related Work

Curriculum

At RWTH Aachen University a two-semester university degree curriculum on IT security is offered (see Figure 1 and [10]). We briefly describe the scope of the individual courses with particular attention to the courses with offensive aspects.

- The first semester has three elements: (1) a lecture on *basic concepts of computer security*, (2) a lecture on *computer forensics*, and (3) a *research seminar* on current trends in computer security where students give a presentation.

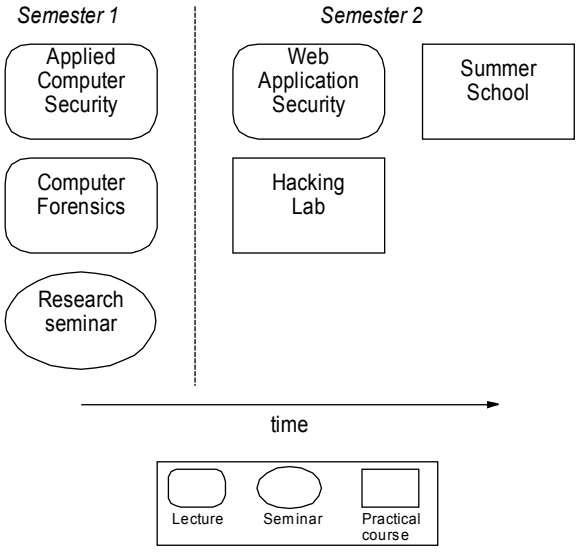

Figure 1: IT security curriculum at RWTH Aachen University

- The second semester consists of (1) a lecture on *security failures in web applications* and (2) an *extensive practical lab* in which students apply offensive and defensive techniques within an isolated test network. The final part of the semester is a *Summerschool* in which advanced attacking techniques are trained and analyzed.

Lecture on Basic Concepts of Computer Security

The lecture *Applied Computer Security* is a standard lecture (4 lecture hours per week) in which basic concepts of computer security are explained to the audience. After discussing basic terminology issues, the course first looks at the security concepts which are present in a standalone UNIX or Linux PC (authentication techniques, access control, cryptography basics, backups and physical security, among other). The second part of the course covers network and Internet security issues, e.g., securing TCP and UDP services, network based authentication systems, network file systems and secure programming techniques. Whenever suitable, common vulnerabilities and attack techniques are discussed to focus on weaknesses of current technologies. For example, dictionary attacks on passwords are discussed in the lecture on password-based authentication, or buffer-overflow attacks are discussed in the context of secure programming techniques. This lecture is meant as a "mind opener" to a broad range of aspects of computer security.

Lecture on Security Failures in Web Applications

Common Failures in Internet Applications (or "How Webservers get 0wned") is a lecture designed to teach how insecure Web applications can be broken. In a case-study approach problems are shown that arise from using web applications such as web server (i.e. HTTP) or mail server (i.e. SMTP), and helpers as PHP, HTTP cookies and authentication (e.g. SQL injection, cross site scripting, session hijacking, tampering of hidden HTML fields). Finding vulnerable services by using search machines (e.g. database or printer administration web pages accessible via the Internet) and available tools (e.g. application proxies) are demonstrated to the audience.

Lecture on Computer Forensics

Classic computer forensics is the gathering, interpretation and presentation of evidence found on computers, supplying services to the legal system. In our course on *Computer Forensics* we broaden the definition of computer forensics to a more computer science like definition: We understand computer forensics not primarily as a tool for the legal system, but also as a tool for understanding security. Sound engineering principles dictate a thorough analysis of failures to learn the workings of a system and avoid subsequent failures of the same kind in the future. We define computer forensics as "the attempt to reconstruct the events which lead to a security policy violation in an information security system". Thus computer forensics also includes the analysis of security incidents to learn the tools, tactics and techniques of the attackers and to gather facts needed to improve security in the future. The computer forensics lecture and accompanying exercises aim at providing students with the necessary knowledge to understand evidence on computers at a very deep level. A big part of the lecture consists of deepening the topics from operating system and systems programming courses in areas of specific interest to forensics. These are namely networking, process management and file systems with a strong bias to the latter. Based on a deep understanding of how relevant parts of information systems work, students learn how to extract and

interpret evidence from such systems and to evaluate the validity of that information gathered. We aim at giving the students a fundamental view on how the extraction of evidence from IT systems works, enabling them to conduct forensic analysis without anything but the most basic tools. Our students should have the ability to develop tools they need to make their forensic analysis more swift whenever they are in need of such tools. Ready made software tools are only covered briefly in the lectures since we assume that the fundamental knowledge acquired during the class should enable students to quickly understand the forensic tools available on the market.

Practical Lab Course

The *Hacking Lab* is a practical introduction to defensive as well as offensive computer security measures. We extend the students' existing theoretical knowledge base by letting them experience failures in real systems under their control in the presence of malice. Goal of this course is to give students the opportunity to learn real life security in a controlled environment making them aware of security problems and enabling them to develop advanced techniques for defense. We think the best way to do this is by getting to know both sides: as an administrator of a computer learn the defensive measures, as a hacker learn the offensive measures. Students work in teams of three to four. The computers used are inside an isolated test network, giving the students the possibility to try attacks and getting attacked without implications to the rest of the world. By executing attacks and applying other techniques (e.g. network sniffing), students get to know weaknesses and vulnerabilities of software and the network. This knowledge aids them to secure their systems. Now that they know the weaknesses, they can develop countermeasures.

We expect the students to have gained knowledge at the end of the course of how to handle a server that has to provide services in an environment where neither the software nor the users can be trusted. Additionally we force the students to analyze what actually happened during the execution of an attack and what measures could have been taken to avoid it.

The course has evolved over time: the first time there were only few instructions given and students had to install the (operating) systems themselves. Time was divided into phases (install, explore, secure, attack) and participants were expected to find relevant literature themselves. The next time systems were preinstalled and some instructions were provided (e.g. a talk presenting the basics on each topic) and participants had to work on some work sheets and hand in their results. The current lab features a C*apture-the-Flag* (CTF) contest, bi-monthly work sheets that need to be handed in and are graded. Additionally, the lab network is connected via VPN to four other German universities so that participants from these universities can explore the remote lab networks.

Summerschool

In the two or three week *Summerschool "Applied IT Security"* students are given the opportunity to induce and study failures in security systems. Other goals are knowledge transfer and to introduce students to a scientific approach to information security. A day of the Summerschool starts with a lecture on a certain topic (e.g. forensics, malware, web applications, honeynets). In the lab session during the rest of the day participants apply the techniques learned in the lectures and develop them further. In the afternoon a so called "coffee table talk" is offered where an invited speaker presents a topic of his interest. The coffee table talks are intended to broaden the view of the participants and to get a focus on problems being faced in the real world. A day concludes with a meeting where everybody presents his work of the day. [11] gives more information on the design and deployment of the Summerschool.

Related Work

The computer science department of Darmstadt University of Technology, Germany, since 1999 regularly runs a so-called *Hacker Contest* [19]. The Hacker Contest is a lab course in which students form teams that have to set up systems and then use common exploitation techniques to attack the systems of the other teams, analyze attacks to their own systems and increasingly deploy stronger defense measures. In military education, one can find similar examples of offensive lectures, for example [23].

Several projects have pioneered the use of offensive techniques as teaching concepts but none of them has treated offensive techniques in a research-oriented way, as it is done in the Summerschool of our curriculum. So called *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. Challenges usually are modeled somewhat after the problems an attacker faces when attempting a system penetration. Typical Wargames can be found at [8,14]. Slightly more competitive than Wargames are *Capture-the-Flag* (CTF) or *Deathmatch* contests where teams battle against each other over the control of a network. Most famous is probably the Root-Fu contest in the USA [13] and the CTF contest of the UCSB, in which several educational institutions spread across the United States battle against each other [22,21].

The Information Technology and Operations Center at the U.S. Military Academy West Point has a curriculum which also teaches offensive information security techniques. The center organizes a yearly *Cyber Defense Exercise* which has similarities to the Capture-the-Flag contests. U.S. authorities with an information security education branch like the United States Military Academy, the United States Air Force Academy and the Naval Postgraduate School participate in these exercises. Machines maintained by the participants are attacked by the NSA 92nd Aggressor Squadron - Land Information Warfare Activity over the course of several days and participants have to counter these attacks [9,18].

3.Attack Versus Defense

None of the above mentioned related approaches tries to estimate the use of offensive methods over defensive methods in teaching information security to students. As stated in the introduction, we believe that offensive techniques are central to better understand the ways in which security systems fail. But how can we prove this in a scientific manner? We are not aware of any such attempts in the security community.

But if one takes a look at disciplines from social sciences as psychology and pedagogical sciences we see that there's a long history of and methods for measurement of knowledge. We'll use these methods to compare the effects of both approaches.

In the following section we give a short overview of methods used in empirical studies. Section 3.2 then presents the experimental setup we chose to compare the effects of teaching offensive and defensive techniques.

Empirical Research

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 might be a *one-tailed* (or *directional*) hypothesis (the direction of a possible difference is specified) or *two-tailed* (direction is not specified). In either case it suspects a link between at least two variables, which might 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* is presumed to potentially affect a *dependent variable*. Important for the validity of a study are the *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. The validity is influenced most by two concepts: field vs. laboratory experiment and 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). If in a study a naturally grown group is compared (e.g. a school class) it's called a *quasi experimental* design while in a *true experimental* design the groups to be examined are created using randomization (to help ensure that the groups are the same before treatment). See the books [5] and [17] for in-depth information.

Empirical Study

As scenario we assume a practical course in IT security for students with offensive orientation, as the Hacking Lab or the Summerschool. To show that the offensive techniques learned in such a course lead to a better understanding of IT security we rather 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 classical defensive education (this implies, that we will have to offer a defense-oriented course as well).

So we refine our hypothesis to:

> *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. Our independent variable is "offensive – defensive", as dependent variable we get "understanding of IT security".

To conduct the study there are two choices: examination of the students who took the offensive or the defensive course respectively as part of their curriculum. Since in this case the students decided which course to take there's no control of personally confounding variables and the internal validity of the study is affected negatively. The second choice, applying randomization to form the groups, makes it a true experimental design, improving internal validity: n subjects are randomly assigned to two treatment groups S_1 and S_2 with size n_1 and n_2 (ideally $n_1 = n_2$). Group S_1 receives one treatment (here: offensive techniques), group S_2 the other treatment (here: defensive techniques). Then the study is conducted on the two groups. We choose the second approach.

In our study we don't consider a control group (i.e. one without treatment, here: no security education at all). This might be taken into account in later studies, leading to a three-group design.

To find out if other independent variables are relevant, a factorial design can be used: a second independent variable is added to the study. In our case this might be "previous knowledge", "motivation", "intelligence" or "type of instructions" (guided/not guided). If we assume the second independent variable to also be two-leveled (B_1, B_2) it is a 2x2 factorial design and thus four sample groups S_{11}, S_{12}, S_{21} and S_{22} (see Table 1) to which subjects must be randomly distributed. In case of a variable we can't influence (e.g. intelligence) we need to select subjects (e.g. by help of an IQ test or questionnaire).

Table 1. Examination scheme of a two-factorial design with independent variables A and B

	B_1	B_2
A_1	S_{11}	S_{12}
A_2	S_{21}	S_{22}

As an additional measure, a questionnaire can be used at the start of the course to measure the knowledge of the participants and one at the end. The results are compared to determine the increase in knowledge. Another test some time after the course helps us determine what the subjects remember from the course.

Experiment

A practical test at the end of the course is used as the experiment: Students are each presented a computer system that is configured with a number of security holes and vulnerabilities. They are asked to identify and fix the vulnerabilities to configure the system in a secure way. For evaluation, the number of vulnerabilities found, the time used (relative to the result achieved), and the strategy used are measured (see Figure 2).

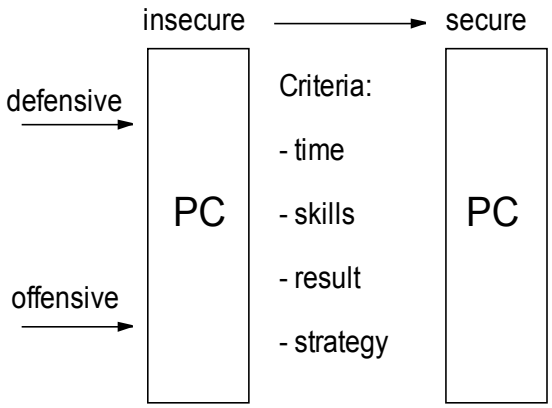

Figure 2: Experimental setup

Restrictions
The setup for this experiment has the restriction that it is limited to securing of a (single) computer.

4.Conclusions

In this paper, we presented an approach to measure the effects of offensive teaching on university students. In order to empirically prove that offensive teaching is in some way better than pure defensive teaching, we use methods from social sciences to set up an experiment with student groups. In future work, we plan to apply the experimental setup to courses at RWTH Aachen University. Results will be available in the course of this summer.

5.REFERENCES

[1] Black Hat briefings, training and consulting. http://www.blackhat.com.

[2] Defcon hacking event, Las Vegas. http://www.defcon.org.

[3] I. Arce and G. McGraw. Guest Editors' introduction: Why attacking systems is a good idea. *IEEE Security & Privacy*, 2(4):17-19, July Aug. 2004.

[4] K. P. Arnett and M. B. Schmidt. Busting the ghost in the machine. *Communications of the ACM*, 48(8):92-95, Aug. 2005.

[5] J. Bortz and N. Döring. *Forschungsmethoden und Evaluation für Human- und Sozialwissenschaftler*. Springer, 3rd edition, 2003.

[6] G. Conti. Why computer scientists should attend hacker conferences. *Communications of the ACM*, 48(3):23-24, Mar. 2005.

[7] G. Conti. Hacking and innovation (guest editor's introduction). *Communications of the ACM*, June 2006.

[8] Digital Evolution. Homepage "Digital Evolution". http://www.dievo.org/. Accessed March 2006.

[9] R. Dodge, D. J. Ragsdale, and C. Reynolds. Organization and training of a cyber security team. In *Proceedings of the 2003 IEEE International Conference on Systems, Man & Cybernetics*, 2003.

[10] D. Farmer and W. Venema. Improving the security of your site by breaking into it. Usenet Posting to comp.security.unix, 3. Dec. 1993.

[11] M. Dornseif, F. C. Freiling, M. Mink, and L. Pimenidis. Teaching data security at university degree level. In *Proceedings of the Fourth World Conference on Information Security Education*, pages 213–222, 2005.

[12] M. Dornseif, F. C. Gärtner, T. Holz, and M. Mink. An Offensive Approach to teaching Information Security: "Aachen Summer School Applied IT Security". Technical Report AIB-2005-02, RWTH Aachen, Jan. 2005.

[13] Ghetto Hackers. Homepage "Root-Fu". http://www.ghettohackers.net/rootfu/. Accessed April 2006.

[14] Hack this page. Homepage "Hack this page". http://www.hackthispage.tk/. Accessed May 2006.

[15] P. G. Neumann. The risks-forum digest. http://catless.ncl.ac.uk/risks.

[16] P. G. Neumann. Inside risks: the big picture. *Communications of the ACM*, 47(9):112, Sept. 2004.

[17] D. Rost. *Interpretation und Bewertung pädagogisch-psychologischer Studien*. Beltz, 2005.

[18] W. Schepens and J. James. Architecture of a cyber defense competition. In *Proceedings of the 2003 IEEE International Conference on Systems, Man & Cybernetics*, 1998.

[19] M. Schumacher, M.-L. Moschgath, and U. Roedig. Angewandte Informationssicherheit: Ein Hacker-Praktikum an Universitäten. *Informatik Spektrum*, 6(23), June 2000.

[20] T. Slewe and M. Hoogenboom. Who will rob you on the digital highway? *Communications of the ACM*, 47(5):56-60, May 2004.

[21] UCSB. Homepage "UCSB Capture The Flag". http://www.cs.ucsb.edu/~vigna/CTF/. Accessed May 2006.

[22] G. Vigna. Teaching network security through live exercises. In C. E. Irvine and H. L. Armstrong, editors, *World Conference on Information Security Education*, volume 253 of *IFIP Conference Proceedings*, pages 3-18. Kluwer, 2003.

[23] G. White and G. 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.