Konflikt oder Review zwei Ansätze für Labors in angewandter Informationssicherheit

Michael Näf · David Basin

Bei der Ausbildung in Informationssicherheit spielen Lehrveranstaltungen zu angewandten Aspekten eine wichige Rolle. Das in Vorlesungen angeeignete Konzeptwissen kann im Rahmen einer praktischen Lehrveranstaltung umgesetzt, vertieft und erweitert werden.

Gerade in der Informationssicherheit ist neben den theoretischen Grundlagen die Auseinandersetzung mit realitätsnahen Situationen wichtig. In der Diskussion mit anderen Studierenden oder Tutoren ergeben sich neue Probleme und Lösungsansätze, und bei der Arbeit mit der konkreten Technik zeigt sich, dass die Sicherheit eines Systems häufig von unscheinbaren Details abhängt. Diese Details können erhebliche Sicherheitsmängel verursachen, und es ist schwierig, sie alle in der Theorie zu erfassen.

Die Bedeutung von laborbasierten Lehrveranstaltungen wird auch in der Curriculum-Entwicklung betont [2, 6]. Die Vorteile von Labors sind vielfältig. Insbesondere fördern sie die Arbeit im Team, die Auseinandersetzung mit technischen Aspekten sowie das Erarbeiten von praktikablen Lösungen zu realistischen Problemstellungen unter Berücksichtigung verschiedener Randbedingungen wie Benutzbarkeit, Kosten, Effizienz oder eben Sicherheit.

Wie kann eine laborbasierte Lehrveranstaltung zu angewandter Informationssicherheit aussehen? Welche Lerninhalte bieten sich an? Und welchen Aktivitäten gehen die Studierenden nach? Mit diesen Fragen beschäftigen wir uns in den folgenden Abschnitten.

Der Konfliktansatz

Verschiedene Institutionen bieten seit langem so genannte "Hacker-Labs" an, bei denen in Echtzeit durchgeführte Angriffs- und Verteidigungsaktivitäten stark im Vordergrund stehen. Wir nennen dies den *Konfliktansatz*. Er gliedert sich gemäss der gesichteten Literatur (z. B. [3–5]) in der Regel in vier Phasen:

1. **Wissensvermittlung:** Den Studierenden werden die nötigen Kenntnisse und Fertigkeiten im offensiven und defensiven Bereich beigebracht. Sie lernen, wie man ein fremdes IT-System angreifen beziehungsweise ein eigenes System absichern und überwachen kann und wie man mit forensischen Techniken Beweismaterial sammelt. Die Wissensvermittlung geschieht mit Hilfe von Vorlesungen, Laborexperimenten oder Fachliteratur.
2. **Konzeption und Implementation:** Ausgehend von einer Übungsanlage mit Regeln und Randbedingungen erarbeiten die Studierenden in Teams ein Konzept für ihr eigenes IT-System sowie Angriffs- und Verteidigungsstrategien für die Konfliktphase. Anschliessend implementieren sie ihr System anhand des Konzepts.

DOI 10.1007/s00287-005-0020-5

Informationssicherheit
Departement Informatik
ETH Zürich
CH-8092 Zürich
Tel.: +41 44 632 72 43
E-Mail: michael.naef@inf.ethz.ch E-Mail: basin@inf.ethz.ch

3. **Konflikt:** Die Teams versuchen, ihre eigenen Systeme zu schützen und gleichzeitig fremde Systeme erfolgreich anzugreifen. In manchen Ausprägungen beschränken sich die Teams auf die Verteidigung und werden von professionellen Sicherheitsspezialisten – so genannten Red oder Tiger Teams – angegriffen.
4. **Nachbearbeitung:** Die Erfahrungen werden reflektiert und zusammengefasst, zum Beispiel in Form einer Präsentation oder eines Berichts.

Wichtigste Eigenschaft dieses Ansatzes ist der in Echtzeit ausgetragene Konflikt. Eines der ersten Labors dieser Art stammt nicht zufällig aus einer US-amerikanischen Militärakademie [4]. In Anlehnung an die Kriegsführung ist von Waffen, Kampf und Schlachtfeld die Rede. Taktiert wird basierend auf den Erkenntnissen von Militärstrategen wie Sun Tzu oder Julius Caesar [8].

Aufgrund der Echtzeiteigenschaft stehen die Teams unter zeitlichem Druck. Sie müssen Problemursachen rasch identifizieren und geeignete Gegenmassnahmen effizient auswählen und umsetzen. So eignen sich die Studierenden in kurzer Zeit viel Wissen über technische Zusammenhänge an und lernen, sich im Team zu organisieren.

Der Review-Ansatz

Ausgehend von diesem Verständnis des Konfliktansatzes haben wir am Departement Informatik der ETH Zürich 2003 die Veranstaltung *Applied Security Laboratory* [1] konzipiert. Wir verfolgen einen ähnlichen Ansatz, setzen aber einen anderen Schwerpunkt:

1. **Wissensvermittlung:** Die Studierenden erarbeiten das notwendige Wissen in Zweiergruppen mit Hilfe eines Manuals zum Selbststudium. Gewisse konzeptionelle Vorkenntnisse in Informationssicherheit und anderen Fachgebieten werden für den Besuch der Lehrveranstaltung vorausgesetzt. (Umfang: etwa 30% des Semesters mit 14 Wochen, pro Woche 3 Laborstunden sowie zusätzliche individuelle Arbeit von etwa 3 Stunden und mehr)
2. **Konzeption und Implementation:** Die Studierenden werden in Teams aufgeteilt. Alle Teams erhalten dieselbe Aufgabenstellung und sollen ein System entwerfen und realisieren, das die gestellten Anforderungen erfüllt. (Umfang: 30%)
3. **Review:** Jedes Team führt einen sorgfältigen, technischen und konzeptionellen Review des Systems von jeweils einem anderen Team durch. (Umfang: 20%)
4. **Nachbearbeitung:** Alle Teams dokumentieren die eigenen Systeme sowie die Ergebnisse aus den Reviews. Die Berichte werden bewertet und an alle Studierenden verteilt. Besonders bemerkenswerte Ergebnisse werden im Plenum vorgestellt und diskutiert. Ausserdem geben Experten aus der Industrie in einer ergänzenden Vortragsreihe Einblick in ihre tägliche Arbeit und beleuchten ausgewählte Themen aus ihrer Perspektive. Zur Nachbearbeitung gehört ebenfalls eine geeignete Leistungskontrolle. (Umfang: 20%)

Im Vordergrund des *Review-Ansatzes* steht somit nicht die Echtzeit-Auseinandersetzung, sondern ein zeitunabhängiger Review. Die folgenden Abschnitte beschreiben den technischen und inhaltlichen Aufbau des Applied Security Laboratory im Detail.

Technische Infrastruktur

Vier grundlegende Anforderungen haben entscheidenden Einfluss auf den technischen Aufbau der Laborinfrastruktur:

- **Benutzbarkeit:** Die Studierenden sollen effizient und selbstständig arbeiten können. Sie benötigen privilegierten Zugriff auf die Experimentierumgebung.
- **Einfachheit:** Der Aufwand für die Erstimplementation und den laufenden Betrieb soll durch eine möglichst einfache Infrastruktur gering gehalten werden.
- **Sicherheit:** Drittsysteme in umliegenden Netzen und im Internet sollen durch versehentliches Fehlverhalten im Labor nicht in Mitleidenschaft gezogen werden.
- **Wiederholbarkeit und Erweiterbarkeit:** Bestehende Experimente sollen problemlos wiederholt durchgeführt, neue Experimente auf einfache Weise implementiert und bereitgestellt werden können. Es soll ausserdem eine gewisse Unabhängigkeit der Experimentierumgebung von der unterliegenden Hard- und Software erreicht werden.

Diese Ziele erreichen wir mit der in Abb. 1 dargestellten Architektur. Die zehn Laborarbeitsplätze stehen in einem separaten Raum und sind über ein eigenes Netz miteinander verbunden. Ein Infrastrukturserver stellt zentrale Dienstleistungen zur Verfügung: Benutzerverwaltung, Netzlauf-

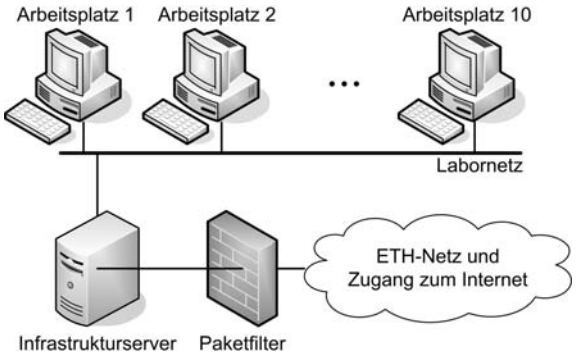

Abb. 1 Laborinfrastruktur

werk, Webserver, Internetzugang. Eine Firewall übernimmt die Paketfilterung zu externen Netzen.

Auf den Arbeitsplätzen kommt VMware zum Einsatz [7]. VMware virtualisiert eine x86-Architektur. Auf diese Weise können auf einem einzigen Rechner mehrere – möglicherweise unterschiedliche – Betriebssysteme in jeweils eigenen virtuellen Maschinen laufen. Die Maschinen lassen sich auch virtuell untereinander vernetzen (Abb. 2).

Im Applied Security Laboratory werden vorkonfigurierte virtuelle Maschinen bereitgestellt. Die Studierenden laden diese Maschinen auf ihren Arbeitsplatz und erhalten so eine wohldefinierte, eigenständige Experimentierumgebung mit mehreren vernetzten Rechnern. Dieser Ansatz hat verschiedene Vorteile:

- Die Studierenden erhalten volle Administrationsrechte auf den virtuellen Maschinen, ohne dadurch die Konfiguration des Arbeitsplatzrechners zu gefährden.
- Die vernetzten Systeme laufen lokal. Die Studierenden stören sich nicht gegenseitig, z. B. durch interferierenden Netzverkehr. VMware und die Arbeitsplatzrechner sind im Labor deshalb so konfiguriert, dass die virtuellen Maschinen nicht „nach aussen" kommunizieren können.
- Der Zustand einer virtuellen Maschine lässt sich abspeichern und wieder herstellen. Bedienungsfehler und durch Experimente verursachte Veränderungen können einfach rückgängig gemacht werden.
- Der Bedarf an Hardware und physischem Raum wird reduziert. Gleichzeitig wird die Portabilität der Experimentierumgebung auch bei gewissen Änderungen an der Trägerplattform gewährleistet.

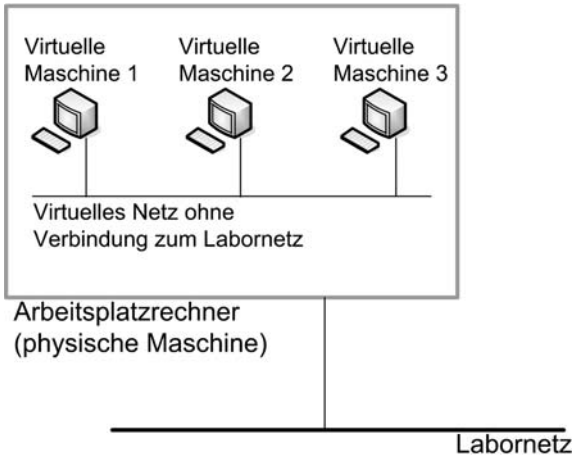

Abb. 2 Virtuelle Experimentierumgebung innerhalb eines Arbeitsplatzes

Wissensvermittlung

Die Wissensvermittlung erfolgt mit Hilfe eines ausführlichen Manuals, das die Studierenden im Selbststudium durcharbeiten. Das Manual konzentriert sich zur Zeit auf die Schwerpunkte Betriebssystemsicherheit und Webapplikationssicherheit. Der Inhalt ist eine Mischung aus Hintergrundinformationen (oder Referenzen auf weiterführendes Informationsmaterial), praktischen Experimenten und Fragen, mit deren Hilfe das Beobachtete reflektiert wird. Abb. 3 zeigt einen kurzen Ausschnitt. Einige wenige Vorträge ergänzen den im Manual behandelten Stoff und gehen zum Beispiel auf verschiedene Standards im Bereich Informationssicherheit, das Verfassen von Risikoanalysen sowie ethische und rechtliche Aspekte ein.

Die Arbeit mit den Manualen findet an den Rechnern im Labor mit Hilfe der vorbereiteten virtuellen Maschinen statt. Die Studierenden arbeiten in Zweiergruppen. Hauptgrund dafür ist die limitierte Zahl von Laborrechnern. Ausserdem bietet die Arbeit zu zweit die Gelegenheit zur Diskussion und gegenseitigen Hilfestellung. Zur weiteren Unterstützung sind Tutoren zu festgelegten Zeiten anwesend.

Konzeption, Implementation und Review

In der Phase „Konzeption und Implementation" werden die Studierenden mit einem realitätsnahen, aber überschaubaren Projekt konfrontiert. Eine ausführliche Aufgabenstellung definiert das Projektziel sowie funktionale und Sicherheitsanforderungen (Abb. 4). Die Arbeit erfolgt in Vierergruppen. Auf diese Weise hält sich der Aufwand für den Einzelnen in Grenzen und Entscheidungen oder Probleme können – wie bei Teamarbeit üblich – gemeinsam diskutiert werden. Die Realisierung des Projekts

Integritätsprüfung

Um Manipulationen an Dateien festzustellen, werden Dateiattribute des Ist-Zustands mit einem früher aufgezeichneten Soll-Zustand verglichen. Eine sehr simple Möglichkeit können Sie manuell ausprobieren:

```
# find / -xdev -type f -print0 | xargs -0 md5sum > /tmp/new.md5
# diff -1 -u /etc/old.md5 /tmp/new.md5
```

Frage 12: Welche Manipulationen decken reine Dateiprüfsummen, wie sie in diesem Beispiel benützt werden, nicht auf?

Es gibt verschiedene Werkzeuge wie Tripwire oder AIDE, die komfortabler und umfassender sind als die gezeigte Einfachstlösung. Wir werden uns mit dem Open Source Tool AIDE näher auseinander setzen.
[…]

Abb. 3 Auszug aus dem Selbststudienmaterial

„Home-Grown" Certification Authority
Konzeption, Implementation und Review

Die fiktive Firma iMovies.ch will erste Schritte in Richtung PKI-basierte Dienstleistungen unternehmen. Zu diesem Zweck soll eine einfache Certification Authority (CA) aufgebaut werden, mit deren Hilfe die internen Mitarbeitenden mit digitalen Zertifikaten versorgt werden können.

Aufgabe

Konzipieren und realisieren Sie die CA gemäss den unten stehenden Anforderungen. Führen Sie anschliessend einen Review bei einer anderen CA durch.

Funktionale Anforderungen

[…]

Sicherheitsanforderungen

[…]

Im Detail leiten Sie die notwendigen Sicherheitsmassnahmen anhand einer Risikoanalyse ab. Eine sinnvolle Vorgehensweise wird im Risk Management Guide for Information Technology Systems des National Institute of Standards and Technology vorgestellt (http://csrc.nist.gov/publications/nistpubs/800-30/sp800-30.pdf).

Review

[…]

Termine

[…]

Schriftlicher Bericht

[…]

Abb. 4 Auszug aus der Projektaufgabe

hilft den Studierenden, das im Selbststudium angeeignete Wissen zu vertiefen, zu festigen und zu erweitern.

Es folgen die Reviews. Dank des Einsatzes von virtuellen Maschinen können die Studierendenteams die fertigen Systeme inklusive aller benötigten Zugriffsinformationen einfach austauschen. Jedes Team erhält genau ein fremdes System zum Review. Aufgabe der Studierenden ist es, das fremde System zu begutachten, möglichst viele Schwachstellen zu identifizieren und geeignete Gegenmassnahmen aufzuzeigen. Diese Analyse basiert sowohl auf praktischen Experimenten und Untersuchungen am laufenden System als auch auf der vermehrt gedanklichen Auseinandersetzung mit der Architektur, der Applikationslogik oder der konkreten Implementation (Stichwort Code Review) des Fremdsystems. Dabei werden die Teilnehmer auch dazu angehalten,

Vergleiche zum eigenen System zu ziehen und besonders gelungene Aspekte im Fremdsystem hervorzuheben.

Zum Schluss reichen die Teams einen schriftlichen Bericht ein, der sowohl die Entwicklung des eigenen Systems inklusive Risikoanalyse als auch die Review-Ergebnisse dokumentiert. Der Bericht wird bewertet. Ausserdem erhalten die Studierenden Zugriff auf alle Berichte, sodass eine umfassende Dokumentation der Projektaktivitäten entsteht.

Ergebnisse

Wir haben das Applied Security Laboratory bisher zweimal durchgeführt. Die folgenden Aussagen fassen die wichtigsten Erkenntnisse zusammen. Sie basieren auf unseren Beobachtungen während der Veranstaltung sowie auf den Rückmeldungen auf eine Befragung der Teilnehmer aus der letzten Durchführung. Die Teilnehmerzahl war aufgrund der Raumgrösse auf 20 beschränkt. 16 Teilnehmer gaben eine Rückmeldung ab.

Der Gesamteindruck war sehr positiv. Besonders hervorgehoben wurden das praktische Arbeiten am Rechner, die Projektarbeit im Team, die intensiven Diskussionen sowie die Vorträge der Spezialisten aus der Industrie.

Die Projektaufgabe wurde von einer Mehrheit mit sehr gut bewertet. Ein wichtiger Grund scheint der grosse Freiraum aufgrund der in vielen Punkten offenen Aufgabenstellung zu sein. Dies führt zu intensiven Diskussionen beim Entwerfen der Lösungen. Ausserdem entstehen auf diese Weise unterschiedliche Lösungsansätze, deren Begutachtung weiteren Erkenntnisgewinn liefert. Ein Student bringt es wie folgt auf den Punkt: „Ich fand die Aufgabe sehr spannend. Insbesondere sah man gut, wie die in den Praktika recht unabhängigen Bereiche erst als Gesamtes ein sicheres System ausmachen."

Die technische Infrastruktur bewährte sich bis auf wenige, leicht zu korrigierende Details. Die Studierenden zeigten sich zufrieden und konnten zügig arbeiten. Aus Sicht verschiedener Teams ist direkter Internetzugriff aus den virtuellen Maschinen wünschenswert, um die Installation des Betriebssystems und das spätere Einspielen weiterer Komponenten oder Patches zu vereinfachen. Bisher musste Software über Sekundärspeichermedien auf die virtuellen Maschinen gebracht werden.

Gegenüberstellung

Konflikt- und Review-Ansatz ähneln sich stark (Abb. 5). Beide Ansätze motivieren die Studierenden dank der realitätsnahen Ausgangslage und dank der kompetitiven Herausforderung während der Analyse. Ausserdem fördern sie die Arbeit im Team. Daneben können wir zwei fundamentale Unterschiede identifizieren:

Zeitdruck: Beim Konfliktansatz stehen die Studierenden unter grösserem Zeitdruck. Der Schwerpunkt liegt deshalb verstärkt auf dem Einsatz von kurzfristigeren Angriffs- und Verteidigungsmechanismen. Die Schnelligkeit im Umgang mit den verschiedenen Werkzeugen, aber auch die rasche Erfassung und Eingrenzung von Problemen und Angriffsversuchen spielen eine grosse Rolle.

Beim Review-Ansatz haben die Studierenden mehr Zeit und befassen sich in Ruhe mit Sicherheitsproblemen, die auch über das im Augenblick Machbare hinausgehen können. Insofern ist die Auseinandersetzung umfassender. Ein Beispiel dafür wäre ein Brute Force Angriff (d. h. stures Durchprobieren aller Kombinationen) auf einen Verschlüsselungsmechanismus mit ungenügender Schlüssellänge, der in der Theorie durchführbar ist, in der Praxis aber genügend Zeit oder spezielle Hardware voraussetzt.

Blickwinkel: Beim Konfliktansatz erfolgt die Betrachtung des fremden Systems in der Rolle eines potenziellen Angreifers und deshalb in erster Linie von aussen. Erst nach einem erfolgreichen

Abb. 5 Gemeinsame Struktur von Konflikt- und Review-Ansatz

Angriff kann das System auch von innen analysiert werden. Gleichzeitig nimmt ein Team die Rolle eines Verteidigers und allenfalls eines Forensikers ein und analysiert das eigene System im Hinblick auf dessen Eignung zur Abwehr oder Erkennung von fremden Angriffen.

Beim Review-Ansatz gibt es nur eine Rolle, diejenige des Reviewers. Die Betrachtung geschieht von innen. Die Teams erhalten vollen Zugriff auf alle Komponenten des zu untersuchenden Systems und können eine umfassende Analyse vornehmen. Das ist wichtig, denn es gibt verschiedene Sicherheitsprobleme, deren Identifizierung durch Systemzugriff oder Einblick in die Funktionsweise einer Applikation stark vereinfacht oder überhaupt erst ermöglicht wird. Dazu gehören beispielsweise zu wenig restriktiv gesetzte Zugriffsrechte im Filesystem, Programmierfehler in privilegierten Systemprogrammen, mangelhafte Backup-Mechanismen oder Race Conditions innerhalb einer Eigenentwicklung.

Fazit

Die beiden gegenübergestellten Ansätze konkurrieren nicht. Sie rücken schlicht unterschiedliche Tätigkeiten und Fähigkeiten in den Vordergrund. Und beide Ansätze bewähren sich in der Praxis, wie die Literaturangaben beziehungsweise die Erfahrungen an der ETH Zürich belegen.

Eine Bildungsinstitution muss sich ausgehend von den angestrebten Lernzielen für den geeigneten Ansatz entscheiden. Es ist schwierig, genaue Kriterien für diesen Entscheid zu geben. Aus unserer Sicht gilt als Tendenz: Der Konfliktansatz betont die intensive Auseinandersetzung mit Angriffs- und Verteidigungstechniken unter Zeitdruck. Dieses Wissen ist besonders für angehende Informatiker in einem betrieblichen Umfeld (beispielsweise bei einem Service Provider) sofort umsetzbar, hat aber auch als Awareness-Massnahme für zukünftige Führungskräfte grosse Bedeutung.

Der Review-Ansatz stellt den Entwurf eines sicheren Systems, die eingehende Untersuchung von Entwurfsvarianten und die Analyse der Implementationen anhand des Entwurfs in den Vordergrund. Er ist demzufolge besonders für angehende Entwickler und Architekten von IT-Systemen geeignet.

Die beiden Ansätze können sich aber auch wie folgt ergänzen: Die Strukturierung erfolgt gemäss der in Abb. 5 gezeigten vier Phasen. In der Analysephase tragen die Studierenden zunächst den Konflikt aus. Daran anschliessend tauschen die Teams die im Konflikt bereits unter Beweis gestellten Systeme aus und führen während des Reviews weitere Untersuchungen durch.

Diese Kombination der beiden Ansätze setzt natürlich voraus, dass die nötige Zeit und die Ressourcen für die Betreuung vorhanden sind. Sind diese Voraussetzungen erfüllt, lassen sich auf diese Weise die Vorteile von Konflikt- und Review-Ansatz vereinen. Aus dem „oder“ im Titel dieses Beitrags würde somit letztlich ein „und“.

Literatur

1. Applied Security Laboratory: http://www.infsec.ethz.ch/lab/appliedlab/
2. Denning, P.J., Comer, D., Gries, D., Mulder, M.C., Tucker, A.B., Turner, A.J., Young, P.R.: Computing as a discipline. Commun. ACM 32(1), 9–23 (1989)
3. Hill, J.M.D., Carver, Jr., A.C., Humphries, J.W., Pooch, U.W.: Using an isolated network laboratory to teach advanced networks and security. In: Proceedings of the 32nd SIGCSE technical symposium on computer science education (S. 36–40). Charlotte, North Carolina, USA: ACM Press 2001
4. Schafer, J., Ragsdale, D.J., Surdu, J.R.: The IWAR range: A laboratory for undergraduate information assurance education. In: Proceedings of the 6th annual CCSC northeastern conference on computing in small colleges (S. 223–232). Middlebury, Vermont, USA: CCSC 2001
5. Schumacher, M., Moschgath, M.-L., Roedig, U.: Angewandte Informationssicherheit: Ein Hacker-Praktikum an Universitäten. Informatik Spektrum 23(3), 202–211 (2000)
6. Tucker, A.B.: Computing curricula 1991. Commun. ACM 34(6), 68–84 (1991)
7. VMware, Inc.: http://www.vmware.com/
8. Welch, D., Ragsdale, D., Schepens, W.: Training for information assurance. Computer 35(4), 30–37 (2002)