A Discovery Learning Approach to Information Assurance Education

Rayford B. Vaughn, PhD
Department of Computer Science and Engineering
Mississippi State University
vaughn@cse.msstate.edu

David A. Dampier, PhD
Department of Computer Science and Engineering
Mississippi State University
dampier@cse.msstate.edu

Abstract

There is an emphasis today throughout many academic programs on developing and sustaining information assurance (IA) programs. While we often refer to such programs collectively as information assurance, they tend to be decidedly different in approach and content depending on many factors. This paper does not attempt to differentiate between such approaches but rather outlines a discovery approach to information assurance pedagogy – regardless of course content or curriculum. Such an approach is applicable whether or not the IA curriculum is a standalone degree or contained within a computer science, business information systems, or computer engineering program. The examples given in this paper were from a computer science and engineering program.

1. Introduction

Information assurance programs of instruction began to take root in the United States during the late 1990’s with the infusion of funding from the Department of Defense and the Federal government. This funding came in the form of research (primarily managed by the National Science Foundation and DOD research organizations) and pedagogy (primarily managed by the National Security Agency as a part of the National Centers of Academic Excellence program) [1]. As a result, many academic programs across the United States began to include information security subjects in their curriculum. Such programs were primarily formed in computer science, computer engineering, and management/business information systems departments. In some cases, full degree programs were developed leading to a graduate degree in information assurance and in other cases, focus areas were created within existing degree programs. As a result of the NSA Centers of Academic Excellence (CAE) program, many universities chose to develop their course content to cover subjects listed in National Training Standards (a requirement of the CAE program) [2]. As of the writing of this paper, there are now over ninety CAE’s across the country that claim to teach to the same national standards. Obviously, with so many institutions involved coupled with the wide variance in faculty backgrounds and academic disciplines, these programs vary widely. In addition, academic programs that are not CAE’s and those not claiming conformance to the National training standards will add to the variance. Since the purpose of information assurance education at the undergraduate and graduate levels is to prepare students for either careers or research endeavors in IA, one has to be concerned with what the end product (i.e., our graduates) is capable of. This paper describes a discovery learning approach to information assurance education at a major US academic institution with CAE and CAE-Research [3] credentials offering BS, MS, and PhD degrees with an IA focus. It is expressly not the intent of this paper to suggest that exact duplication of this approach is necessary or even desired at other institutions. It is suggested, however, that a discovery approach to IA education produces a better student in terms of skills learned, a more engaged student in terms of career interest, and would likely result in more uniform instruction of IA material than is occurring as a result of the NSA mandated courseware mapping process.

2. Discovery learning as an approach

The term discovery learning may not be familiar to the reader. Discovery learning has been promoted in educational literature since the early 1960’s. It takes place in problem solving situations where the learner draws on his own experience and prior knowledge and is a method of instruction through which students interact with their environment by exploring and manipulating objects, wrestling with questions and controversies, or performing experiments [4]. One of the early researchers in this technique, Jerome Bruner, argues that “Practice in discovering for oneself teaches one to acquire information in a way that makes that information more readily viable in problem solving"
3. History, courses and content

The author’s institution [note – this will be identified in the final submission] has maintained an information assurance course focus since 1998, at the BS, MS, and PhD levels. More than 700 students have completed coursework at the undergraduate level and more than 200 at the graduate level. We have produced nine IA PhD’s and have twelve PhD candidates actively engaged in research efforts. The university was designated a CAE in 2001 and a CAE-Research in 2008. In order to permit undergraduate students to focus on information assurance and to prove to prospective employers that they did so, we created an information assurance professional certificate program which is described later in this paper. To facilitate discovery learning in our students and to motivate research, we constructed four laboratories for student use. Lastly, to motivate discovery learning, we developed several individual and team exercises that we require of our students (information assurance labs, capture the flag competition, digital forensics search and seizure in a mock village, and a real courtroom experience for teaching expert witness skills).

Our curriculum weaves IA topics throughout our ABET accredited computer science and software engineering degree programs to provide students with the necessary framework from which they can build on when they take more advanced upper level IA specific courses. At the senior level (for undergraduates) or at the beginning level for graduate students, we offer three specific “split level” classes which are very focused on IA topics. We use the term “split level” to indicate a class that may be taken by upper division undergraduate (at the 4000-level) or graduate students at the (6000-level). The IA focus courses offered are:

- **CSE 4243/6243 Information and Computer Security**: This course is an introductory course to Information Assurance and is mapped to CNSS 4011. It covers the principles of IA, government processes, the DIACAP, Common Criteria, network security, data base security, operating systems security, and more. There is a lab requirement associated with this class.

- **CSE 4273/6273 Digital and Computer Forensics**: This course explores computer crime and the study of evidence for solving such crimes. MOST importantly, we allow students from three colleges to take this class (students from Engineering, Arts and Sciences, and Business). We attract students from all three colleges to this popular class and we involve faculty from our Criminal Justice program (Dept of Sociology) in

...
some of the lectures. There is a lab requirement with this course.

- **CSE 4383/6383 Cryptography and Network Security**: This class covers the workings of off the self cryptography, cryptosystems, algorithms, PKI, symmetric/asymmetric systems, and key management. It also covers network defense and network attack. A strong emphasis is placed on network protocol vulnerabilities. The course is open to all computer science and engineering students.

- **BIS 4113/6113: Business Information Systems Security Management**: This course is offered in the College of Business and open to Business students as well as Engineering students. Concepts, skills, tools and techniques involved in management of computer security as it applies to today's business environment are taught. This class has a security lab associated with it.

The discovery learning aspects of our program are described in the paragraphs that follow. The reader should keep in mind that each of the discovery experiences comes only after (or in conjunction with) lectures by the instructor. The end result we are after is to provide the student with enough motivation and conceptual knowledge that they can then combine with their natural curiosity, and move to a higher level of learning.

4. **Individual lab exercises**

The basic computer security laboratory used by our students is an isolated facility containing over twenty desktop machines, a compliment of hubs, routers, switches, and firewalls, and a separate machine set up to host Norton’s Ghost Server (see Figure 1). Students are not restricted on the type of security experimentation they are allowed to participate in subject to the following general guidelines given at the beginning of each semester.

- Do no harm to others
- Do no work on lab machines other than security class labs or research
- Use the card key entry system for every entry to the lab (for accountability)
- Do not touch the Ghost server in any way (electronically or otherwise)
- Experimentation outside the assigned lab exercises must be discussed with the instructor

There are a total of twelve exercises that are given to the students at the beginning of each semester. The labs are accomplished in consecutive order beginning with Lab 1 due at the beginning of the first class of week #2. Each subsequent lab is due each week in the same manner. By following this procedure, the students automatically know that they have 7 days to accomplish each lab on their own time and that the labs are due weeks 2 through 13 of the semester. There is also a Lab “0” that we include which provides specifics concerning the lab architecture, passwords, and useful information on specific operating systems commands. Each exercise is briefly described below.

**Figure 1. Computer Security Laboratory**

- **Lab 0 INTRODUCTION**: This lab is used to familiarize the student with some of the OS commands they will need to complete the lab exercises during the semester and to present an overview of the lab itself. It should be noted here that students in this class come from a variety of backgrounds to include Management Information Systems, Computer Science, Electrical Engineering, Computer Engineering, and Software Engineering – so the technical skill sets are widely varied.

- **Lab 1 PASSWORDS AND ENUMERATION**: This lab is designed to show the students the importance of selecting and maintaining strong passwords. Students are given a general set of rules on what constitutes a strong password and then are asked to create five user accounts on both a Windows machine and a Linux machine. They are given specific guidelines on how to create two of the five passwords (making them weak) and left to their own decisions on the other three. They are then instructed on how to use a password cracker “John the Ripper” and to attempt to crack the five passwords they created – plus another file of passwords provided to them. Students then learn how fast a cracker breaks a weak password and the value of a strong password. They also compare differences in timing between Linux and Windows OS.
• **Lab 2 PGP:** This lab teaches the students how to use Pretty Good Privacy (PGP) as an open source encryption tool. Students go through a tutorial on PGP in the lab readings and then install and use PGP on both a Windows and Linux/Solaris system. They not only learn how to use a valuable tool – they also become familiar with public key cryptography in the process.

• **Lab 3 Email Spoofing:** Email spoofing is a fairly common technique used today in Spam and Phishing attacks. It can also be used to trick a user into performing an action or providing sensitive information to an attacker. This exercise demonstrates to the student how easy email spoofing is accomplished, vulnerabilities in the SMTP protocol, and how to effectively read email headers. The student better understands why it is important to deny open relaying from email servers and how to reduce the amount of spoofed emails entering their corporate networks.

• **Lab 4 Steganography:** Students are taught the concepts and limits of steganography, how to use existing tools to hide data in a JPEG image, how to use statistical tools to detect such images, and how various file formats are subject to being used for steganography. Tools that are used in this exercise include Jphide, Jpseek, Stegdetect and Stegbreak.

• **Lab 5 Network Scanning and Footprinting:** For this lab, we encourage the students to work in pairs or teams. The lab gives students experience in a common problem faced by network administrators – the use of insecure rogue systems on a network. Students use NMAP for port scanning and NETCAT to determine what services are running on target systems. They are exposed to “banner grabbing” and how easy it is to footprint a system as an attacker. All exercises are run on a Linux machine.

• **Lab 6 Sniffing and Shared versus Switched Environments:** Students use this lab to gain an understanding of hub and switch technology and the vulnerabilities associated with each when faced with a sniffer threat. Students perform the exercise in both a Windows and Linux environment with Hub and Switch equipment provided in the lab. During this exercise students are exposed to Wireshark as an open source sniffer on Windows machines and ettercap as a sniffer on Linux systems. They are also introduced to the Address Resolution Protocol (ARP) and attacks against ARP tables.

• **Lab 7 Tripwire:** Tripwire is used to demonstrate basic intrusion detection. As an open source tool, it is used to determine what changes, if any, have occurred on a system. Students are shown how to install Tripwire and how to use it. The notion of using it to detect changes to Honeytokens is also suggested. While this is the most basic of IDS tools – it is used to facilitate discussion on IDS.

• **Lab 8 Using SNORT:** Network intrusion attempts are so frequent that no matter how well a network is designed, it cannot avoid intrusion attempts. Students install SNORT and review audit logs resulting from its installation. They are also taught the difference between host-based and network-based intrusion detection during this exercise. A part of the lab is designed to allow students to install a SNORT rule set and to modify that rule set.

• **Lab 9 Exploit Development:** This lab is one of the more difficult and students are asked to create a buffer overflow exploit in code. Students are provided with a program called “corkboard” that has intentional coding errors in it that render it vulnerable to a buffer overflow attack. If they exercise the lab correctly, they implement a buffer overflow attack that allows them to elevate their privileges from user to root.

• **Lab 10 Comparing SARA, STAT Scanner and Nessus:** Vulnerability scanners are a necessary tool for system administrators today and are used to find vulnerabilities and to patch them. Students use three such tools in this exercise to gain experience with scanning networks for vulnerability analysis. They employ SARA (Security Auditor’s Research Assistant) which is freely distributed; STAT Scanner, a commercial tool offered for license by Harris Corporation; and, Nessus, a security scanner available for free download.

• **Lab 11 Introduction to Digital Forensics:** We offer a full semester course on this topic so the lab is designed to acquaint students with the topic and to garner their interest in learning more. They must document their activities and search for evidence on an evidence drive. The evidence consists of a large number of photographs of “weasels” that hidden in several different ways.

• **Lab 12 Firewalls and VPN’s:** The purpose of this lab is to help the student understand the principles of firewall security, configure a firewall, understand the concept of a VPN and to configure a VPN. For this lab, a Cisco Pix 515 firewall is used. Students gain experience in configuring the Pix firewall and using it to establish a VPN. While each of these exercises in their own right are not exceptionally difficult, taken as a whole they significantly augment the class with discovery learning exercises. By assigning one exercise each week, the
students can choose their own time during a seven day period to apply a hands-on exercise to what they are being taught in lectures. They are also told that the exercises will teach those tools and techniques that will be helpful during their final Capture the Flag competitive exercise at the end of the semester in which prizes are awarded to each team. This proves to be adequate motivation for the students to complete the exercises.

5. Capture the flag contest

A full week is devoted to the Capture the Flag contest with the vast majority of the activity conducted voluntarily and outside normal class periods (see Figure 2). Students are formed into 4 to 5 person teams with one student per team designated as a team leader. Prior to the exercise, they are taught a framework used for attacking a system (reconnaissance, footprinting, enumeration, probing, and penetration); given lectures on specific tools useful for penetration testing; and, given experience in using tools in the laboratory. They are given the general network architecture that they will be penetrating and specific information on the tokens they will be looking for. The exercise is started one week prior to the actual live fire penetration so that the teams can organize, plan, and practice together. They are also allowed to use any form of social engineering that they can come up with to gain an advantage. No restrictions are placed on the students in this regard. Teams are awarded points for retrieving tokens during the full period of the exercise plus they are awarded some initial points based on a grading of their written plan of attack.

Prizes are awarded to teams and tend to be a real motivator for the students (plus the obvious bragging rights that come from winning). We simply purchase “office supply” items that they students can use. For our most recent exercise, the prize table contained 2GB USB drives with snap links; wireless optical mouse; neon colored R/W CD’s; laser pointers, computer security books, and portfolios. The team placing first is allowed to choose their prizes first, second place second, and so forth. We have found it helpful to invite the news media to witness the event and to document it for local media. By doing so, we are able to further motivate the students and to advertise our program at the same time. The most interesting side effect of this exercise is the intensity seen in the students during the exercise. They prepare hard for this exercise and, judging from the comments we receive on end of course evaluations, they find it instructive and fun.

In preparation for the exercise, students are given brief instruction on a number of tools. We intentionally make the instruction brief so that students will necessarily need to work hard on their own to master the tools. With the motivation already established to win the contest – students tend to not only master the tools we teach, but also discover others on their own that they later share. Specifically we provide instruction on the following.

- **Reconnaissance tools**: We present a number of tools here that can help the student discover information about a targeted system. By exposing the student to these tools, the student can better understand how to defend against them. Tools included are p0f for OS fingerprinting; Network Stumbler for discovering wireless network characteristics; Google for search directives and cache information; and Netcraft for webserver information.

- **Enumeration tools**: Enumeration helps the attacker know more specifics about the operating environment to be exploited. For this lecture, two important tools are covered – NMAP and Nikto. NMAP provides a host of services to include OS fingerprinting, port scans, and other services. Nikto is a web scanner and assists an attacker in finding vulnerable web services.

- **Exploitation tools**: To actually exploit a system, tool support is very useful. Students are introduced to the Metasploit framework, one of the most powerful penetration tools available (essentially a point and click penetration tool); TCPReplay to replay packets on the network; hping2 to craft packets and OS fingerprinting; NetDude for packet crafting; and Cain and Able for wireless sniffing, VoIP recording, and password recovery.

- **Analysis tools**: As a security engineer, the ability to analyze network activity and diagnose problems is essential. Students are shown Microsoft Sysinternal tools [8]; ntop for analysis of network IP traffic; and VisualRoute to visualize network path data.
• **Hardening tools:** There are many automated tools available to lock down systems and make them more resistant to attack. Our students are shown a sampling of these to include Bastille as a tool to harden Linux, HP-UX, Mac OS X and other OS; IPtables as a host based static firewall; and Truecrypt as a freeware product to encrypt laptops.

6. **Mock village**

We offer a similar discovery learning exercise in our Digital and Computer Forensics class. In an effort to instill good forensically sound search and seizure practices as well as sound investigative practices – we first teach basic techniques in the classroom and emphasize to the students that they will be expected to actually apply these techniques at a crime scene and to testify in court about the evidence that they found. Students are taken on a weekend to a mock village used by a Counter-drug training academy facility about 90 miles from the University. Volunteer police are used to assist in the exercise along with teaching assistants from the University. Students spend a day arresting criminals (University staff members), seizing evidence, recording/logging evidence, imaging hard drives, and preserving evidence for courtroom presentation. As can be seen in Figures 3 and 4, the students take the exercise quite seriously.

7. **Moot court experience**

The digital and computer forensics students are given approximately ten days to examine the evidence that they captured during the mock village exercise and to prepare themselves to testify in court before a defense and prosecution attorney. They are given limited instruction in the classroom on courtroom procedure and an actual digital crime investigator is brought in to talk to the students and answer questions on courtroom experience. The key here is that the students must lead the discussion to obtain the information they need to prepare themselves – it is not lectured to them.

In preparation for the moot court experience, we partner with a law school in another state university [note: to be identified in the final paper] and they provide defense and prosecution attorney’s to argue the digital crime cases that the students prepared. We work with the local government officials and reserve the county courtroom for our use. A retired judge is obtained to sit on the bench and hear the cases. A jury is obtained by using another digital forensics class from another state university that we cooperate with [note: to be identified in the final paper]. Staff members “arrested” during the mock village exercise are used as defendants and have a defense attorney to
argue their innocence. The students “correct” testimony and legal evidence admission is paramount to obtaining a guilty verdict and for the eventual satisfaction of having accomplished a successful prosecution.

Figure 5: Student being sworn in for testimony

During this exercise students learn the difficulty of communicating technical information in a non-technical environment and they are taught the value of and need for strong communication skills in additional to technical expertise.

Figure 6. Attorney arguing case, student jury from partnering school in background

8. SCADA laboratory

In order to prepare students for new areas of information assurance practice and research exploration, we focused on critical infrastructure as an area needing attention. To facilitate discovery learning in this area, we acquired a working control system laboratory (aka, Supervisory Control and Data Acquisition – SCADA). Such control systems are widely used within the critical infrastructure to monitor and control industrial functions (e.g., flow metering, power distribution, water quality, and pressure) and to report status of industrial components (e.g., temperature, pressure, and volume). Such systems are, today, connected to the internet and may be very complex hardware/software combinations. The authors firmly believed that such a laboratory, if it was to meet our requirements for facilitating discovery learning, needed to be composed of real control system hardware and software. It also needed to be small enough such that students could observe effects of their experimentation with the laboratory.

The literature reports many vulnerabilities in such systems and there are reports of attacks against critical infrastructure through control system software. Our intention was to provide the laboratory to the students, provide the motivation to research, and then to see what we got as results. The lab was funded by a grant from the National Security Agency and was architected and developed by Mr. John Gordon of Control Systems Inc. (johngordon@controlsystemsinc.net). It was constructed on seven portable carts so that it would be mobile between classrooms and labs and it contains two master control stations running GE Fanuc Proficy iFIX 4.5 control software which communicates with five Control Microsystems [9] SCADAPack controllers employed in miniature critical infrastructure settings (industrial blower application, city water system, oil storage tank, conveyor belt system, and gas pipeline). The hardware and software systems are real – the devices controlled are miniature (see figures 7 and 8).

Figure 7. Control System Laboratory
Several research efforts (and findings) were generated by students working with this laboratory. The first was from a PhD student that became interested in how one would perform forensic investigation of such systems after an attack has been discovered. The student determined that there was not any residual electronic data maintained by the system so that such data could be investigated as evidence or to help determine what was done and by who. The student then developed their own capture device (show in Figure 9) that allows for a data capture device (tap) to be installed within the control system for the capture of MODBUS or DNP3 protocol packets and to write them off to a separate data storage facility. Once the device was developed and shown to work without degrading the system response time – another PhD student noted that the use of such a capture device was very inefficient when constantly employed and only really needed to be activated when the control system was experiencing an anomalous event. This lead the second PhD student to investigate the construction of a honeypot or honeynet that would function within a control system environment and activate the capture device when it detected activity. Further experimentation by the students (including a master’s degree student) has since resulted in the discovery of serious vulnerabilities in operational control system software and within recommended procedures for installation that will introduce exploitable vulnerabilities in critical systems using this commercial system. These vulnerabilities were reported to the National Security Agency.

Additional discoveries are coming out of this effort and are being reported separately in other venues. The key point we make here is that these research efforts are all discovery based and initiated by the students who were introduced to an idea generating laboratory.

9. The information assurance professional certificate.

Students involved in information assurance research and those taking classes as undergraduates expressed a desire to have some proof that they could offer to future employers that they “focused” their study or had a strong interest in information assurance topics. For a graduate student, this was not much of a problem in that the result of their work (thesis or project) would demonstrate that focus. For an undergraduate, this was more of a concern. To accommodate this student request, the authors developed a university approved “Certificate” program that once completed, resulted in an entry on the student’s transcript and a formal paper certificate – both of which documented the student’s focus. The certificate program is open to students from several different disciplines (computer science, software engineering, computer engineering, electrical engineering, and business information systems). The certificate itself involves fifteen hours of course work – three information assurance specific courses and two other supporting courses chosen from a list of acceptable classes. More details can be found at http://www.cse.msstate.edu/ftc. To date, more than 100 such certificates have been awarded. One interesting side effect of this certificate program is that it has attracted a number of international students to the information assurance classes because they see the certificate as employment enhancing.

10. Summary and conclusions

The purpose of this paper has been to demonstrate that students, when given the proper motivations and facilities to work with will, in many cases, learn more by doing than by listening and will actually contribute to the learning experience for the instructor and other students. Students tend to be more energetic toward learning, thrive on the challenge of discovery, and appear to respond to motivations involving prizes, winning, or simply bragging rights. There is absolutely no intention on the part of the authors to claim this method as “better” or more effective than others that are being used. It is reported as empirical evidence of an approach that works and produces results. In addition to the student experiences and
research results identified in this paper, it is worth noting that these same students seem to enjoy publishing the results of their work. More than thirty student publications have resulted from peer reviewed papers written by our information assurance students—with six best paper awards being achieved (three of which were undergraduate students). We motivate student publications by telling our students that if they achieve a published paper, we will support their travel to present it at a conference or any publication fees involved in a journal publication (e.g., reprints).

It appears to the authors that a discovery learning approach is one that works and excites the students to learn and to contribute. We find that students exposed to this approach value the learning experience and report such on their end of the semester evaluations. The authors are prepared to share any of the material discussed in this paper to include specifics on lab exercises.

11. References