Supporting the Education of Information Assurance with a Laboratory Environment

Paul C. Clark

Naval Postgraduate School
833 Dyer Rd., Code CS
Monterey, CA 93943-5118
E-mail: clarkp@cs.nps.navy.mil
(831) 656-2395

Abstract

Too many students are graduating from colleges and universities without taking a single course in information assurance. The need for students to receive more and better education in information assurance is undisputed. For those educational institutions already requiring and/or teaching such courses, the educational experience can be greatly enhanced with a supportive laboratory environment where carefully chosen hands-on tutorials or exercises can be assigned to support the material being presented in the classroom. This paper describes the experiences of supporting information assurance exercises and tutorials at the Naval Postgraduate School. Recommendations are provided so that others may learn from the experience.

Submission: **General**

Biography

Paul Clark is currently working at the Naval Postgraduate School as a Research Associate, lecturing graduate students and performing research in the area of computer security, and is actively participating in the Center for Information Systems Security Studies and Research (CISR). His current area of interest revolves around the betterment of computer security education in the academic environment. He has worked in the computer security industry as an Integration Engineer, Systems Programmer and Customer Service Manager.

Supporting the Education of Information Assurance with a Laboratory Environment

Paul C. Clark

Naval Postgraduate School
833 Dyer Rd., Code CS
Monterey, CA 93943-5118
E-mail: clarkp@cs.nps.navy.mil

Abstract

Too many students are graduating from colleges and universities without taking a single course in information assurance. The need for students to receive more and better education in information assurance is undisputed. For those educational institutions already requiring and/or teaching such courses, the educational experience can be greatly enhanced with a supportive laboratory environment where carefully chosen hands-on tutorials or exercises can be assigned to support the material being presented in the classroom. This paper describes the experiences of supporting information assurance exercises and tutorials at the Naval Postgraduate School. Recommendations are provided so that others may learn from the experience.

KEYWORDS: Education, Information assurance, Computer security

Introduction

There is a critical need to improve the state of education in the field of information assurance (Spafford, 1997), with a presidential commission calling for professionals to initiate needed changes to meet the national demand for professionals in the field. (Marsh, 2000). The scope of the problem is quite large, requiring, for example, educational institutions to recognize the need to make changes, investing in the creation of related courses, having teachers/professors who are prepared to teach them, having the necessary students to be taught, and then having the students apply what they have learned.

The world is at a moment in history when the security of our systems needs to be improved, but vendors continue to sell products with major vulnerabilities. A long-range solution to this problem is to educate the next generation to understand the security issues of software and hardware design so the mistakes of the past are not repeated. One of the many consequences of failing to improve computer security education will be that even more engineers, programmers and managers will not have security skills and will continue to produce insecure products (CERIAS, 2000); a consequence that is totally opposite of our needs.

To this end of improving information assurance education, this paper seeks to share the experiences and lessons learned from supporting a number of computer security courses with laboratory assignments. The courses are taught under the umbrella of the Naval Postgraduate School Center for Information Systems Security Studies and

Research (NPS CISR) (Irvine et al., 1997). It is hoped that it will be of some benefit for those who are already teaching computer security courses, and for those who are looking for help in getting started.

After some background and support information, the remainder of this paper describes various choices that must be made when supporting a computer security lab, listing the advantages and disadvantages of each approach. This includes choices in hardware, software, lab exercises, formatting of exercise handouts, recommendations for written student assignments, and potential exercise topics.

Background

Courses in information assurance have been taught at NPS since 1991, gradually expanding to a total of eight courses that can lead to a M.S in Computer Science, with a specialty in computer security. In order to receive this degree, students must conduct research and write a thesis on a topic related to information assurance, as approved by a faculty member of NPS CISR.

There are currently ten faculty members and ten support staff who are associated with NPS CISR. The number of students taking computer security courses varies from about 50 to 150 students in a quarter. During the academic year of 2000, there were approximately 355 students who took one of 18 sections of computer security courses offered. NPS operates on a year-round schedule with four 12-week quarters per year.

The computer security lab at NPS was first used sparingly to support classroom instruction during the fall quarter of 1995. It became an integral part of the course *Information Assurance: Introduction to Computer Security* during the winter quarter of 1996. Laboratory support has since been provided for the following courses:

- Information Assurance: Introduction to Computer Security
- Information Assurance: Secure Management of Systems
- Internet Security Resources and Policy
- Network Security
- Secure Systems
- Database Security
- Security Policies, Models, and Formal Methods

The students at NPS are United States military officers (including officers outside the Navy), Department of Defense (DoD) civilian employees, and officers from allied countries. Their job is to be full-time students for a period that ranges from one to two years, depending on their background and the degree they are pursuing.

Support for Laboratories

There are different ways to learn new topics, each related to a different human sense. One common way is to listen to someone who knows about the topic of interest. With respect to education, this is the lecture. An advantage to this approach is that it can be interactive, allowing a student to pose questions and gain a deeper understanding of the topic. This works well in an educational institution as long as the number of students is not overly large.

Another way of learning is by reading about a topic of interest. An advantage to reading is that a section of a book can be pondered until its full meaning gets absorbed before moving on; it can also be reread often as a reminder. This is why many college-level courses require a textbook as part of the course. A textbook can enhance or clarify what is being

said in class, and can lead to good discussion as students seek clarification of issues raised in the textbook. In addition, it may not be possible to cover everything one wishes to convey in a quarter, so it allows a student to learn more than if only lecture was included.

Yet another way to learn is through experience, which includes the sense of touch. This has been used for educational purposes for a very long time, often taking the form of labs . A course in chemistry, for example, would not be the same without regular practical lab sessions where the student must apply what is being taught in the classroom, and where the results of the application are very physical and relatively immediate. The use of lab assignments in computer security education has been elegantly supported in (Irvine, 1999). The next section provides detailed recommendations for supporting laboratory exercises for information assurance.

Lessons Learned

In this section the various options and choices for supporting a lab, including lessons learned, are presented.

Hardware

This subsection addresses hardware-specific issues that must be considered when constructing a laboratory.

Platform

The type of computer selected for the lab is fairly important because it can determine the kinds of operating systems available, the software supported, how much it costs to purchase and maintain a lab, and have an impact on the amount of lab space needed.

The first choice is whether to support a variety of hardware platforms, or to stay with one type, such as choosing between Sun workstations or Personal Computers (PCs), or some of each. The advantage of having only one type of system is that it is easier to maintain a homogenous set of computers. If there is no support staff (or the support staff is the professor herself), this can be very important. In addition, requiring only one kind of platform can reduce the amount of lab space required.

The disadvantage of having only one type of system is that there is the potential of ending up with a pile of unsupported hardware, if the choice is not made wisely. For example, there once was a company that produced Sun clone workstations at a fraction of the cost. When that company went out of business, anyone with those systems could no longer update their operating systems, because they were somewhat less than transparent clones. This is less of a risk if PCs are used.

Something else to consider is the life expectancy of the lab systems. PCs have a relatively short time before they are considered too slow , or need more memory and/or disk space. Sun workstations, on the other hand, will perform adequately much longer. This of course comes with the knowledge that workstations tend to cost a lot more, and PCs can be relatively inexpensive. Performance, however, is not to be taken lightly. There should be good enough performance so the student does not need to wait long for applications to start or respond. The NPS lab has had some experience requiring students to use a particular security system that was getting a little old and whose performance was not good, with respect to today s expectations. Some

students went away with the wrong impression, having incorrectly connected their frustration with the performance with the kind of functionality they were learning about.

For teaching support, the NPS computer security labs currently use a combination of different Sun workstations and PC platforms. Over the past five years the NPS lab has used HP workstations and other miscellaneous platforms, but the burden of supporting more than two platforms was too much, in terms of hardware, software, and system administration. However, many of the Sun workstations are old (SPARCstation10), and they will be phased out with new PCs.

An advantage of using PC hardware is that multiple operating systems are available for it, whereas a Sun workstation has very few options. In other words, the PC offers flexibility, which may be important in an environment where change is a constant.

Unless there is an overwhelming reason to use something else, it is recommended that PC hardware be used in the lab, and that it be purchased with as much speed, hard disk capacity and memory as financially possible in order to extend its useful lifetime. For example, the PCs purchased for the NPS computer security lab today are specified with 256 MB of RAM, even though they really do not need it (yet), leaving at least one of the memory slots open in case an upgrade is required in the future.

Booting Multiple Operating Systems

As noted in the last section, multiple operating systems are available for the PC platform, offering a lot of flexibility. This flexibility extends beyond being able to change operating systems at a later date. In fact, one PC can be used to support multiple courses during the same quarter/semester, even if the courses require different operating systems. This can translate into fewer required systems and a smaller lab space to support classes, which means that a smaller budget is needed to start a laboratory. It may be possible to turn one lab with one purpose into a lab with two purposes, such that no new PCs need to be purchased. The question then is: what is the best way to install and boot multiple operating systems on a single PC? There are three possible ways, each with their own advantages and disadvantages: multiple operating systems on a single hard drive; swappable hard drives with a single operating system on each disk; and virtual machine monitor technology, such as that available from VMware.

A multi-boot system is one where multiple operating systems are installed on multiple hard disk partitions on a single disk, and where a menu is presented during the boot process to choose the one desired. The advantage to this approach is that it does not require coordination by anyone to swap out drives at the right time, or to worry about the theft of selectable drives sitting out on a table. There may be some amount of effort to get the operating systems to install and work in this fashion. Or it may require the purchase of a product that will do it for you, such as System Commander, from V Communications, Inc., or Partition Magic from PowerQuest Corp. A multi-boot system that does not require special software to do the booting is probably the cheapest of the three solutions.

Another disadvantage to a multi-boot system is that older PC BIOS s assumed that IDE drives had a maximum capacity of 8.4GB, so older PCs cannot boot a partition that is beyond the 8.4 GB barrier. If a number of operating systems are required, and they require a large amount of space, the partitions have to be carefully mapped out so the bootable partitions are below this barrier.

Another potential disadvantage is that if a class assignment requires students to install an operating system, they may accidentally delete the partitioning structure or format the wrong partition(s), requiring extra time and effort by the instructor or lab assistant(s) to recover from the loss.

The second approach is to use swappable hard disks, where each hard disk is dedicated to a particular operating system, or maybe even to a particular course. This approach reduces the threat of accidental deletion of other installations, if a student is required to install other software. In addition, there is the potential for operating system combinations that will not cooperate with a multi-boot situation, such as when two operating systems insist on being installed in the first hard disk partition, and without support for a mutually recognized file system. However, the swapping of disks requires an instructor or lab assistant to coordinate their swapping at the appropriate time(s), which may or may not be a problem, depending on staffing and other logistics.

One problem that may need to be overcome when using multiple operating systems on the same PC, whether swapping hard disks or using a multi-boot solution, is finding the right combination of hardware devices that will work for every operating system installed. For example, a new PC may have a new high-powered video card, and that card will likely have drivers for Microsoft NT 4.0, but probably not for the PC version of Sun Solaris. Therefore, sometimes an older PC will have better chances of working with multiple OS s. But if really good performance is needed, this may not be acceptable.

The third approach is to use a product called VMware from VMware, Inc. The VMware product is a Type II virtual machine monitor (Goldberg, 1972; Robin et al., 2000) and allows an operating system to be booted as an application (known as a guest operating system) within another operating system. For example, Linux can be booted as an application running on top of NT, and vice versa. This eliminates the need to deal with hard disk partitioning and booting problems, or dealing with the management of swappable disks. It also resolves the device driver problem since the guest operating system makes use of the host operating system device drivers. However, the functionality comes with a performance penalty for the guest operating system, and requires the PC to have enough RAM to handle multiple operating systems simultaneously. The host operating system is limited to Windows NT, Windows 2000, or Linux. Guest operating systems are limited to Windows 9x, Windows NT, Windows 2000, FreeBSD, or Linux. In addition, the product introduces additional costs, though volume and educational discounts are available. The NPS security labs do not use VMware to support classroom instruction because of the cost of upgrading the PCs to use it, and the cost of the additional licenses for VMware.

If it is necessary to use PCs with multiple operating systems, there is no easy recommendation because it depends on the operating systems that need to be used and the PCs they are being installed on. If students are going to install an operating system as part of an assignment, however, then it is recommended that removable hard disks be used, and that there is a locking mechanism to keep them from being stolen. Otherwise, a multi-boot system appears to be the better alternative in most situations.

Lab Manual for the Introductory Course

Great effort has gone into producing a number of tutorials, or lab exercises, to be used by the

students in *Information Assurance: Introduction to Computer Security*. These tutorials have been bound into an internal document that is referred to as a lab manual. The NPS computer security labs are not big enough to hold enough computer systems to have an entire class participate at the same time, as you would expect with something like a chemistry lab, so the assignments have been carefully written to allow the students to work on the exercises independently and at their convenience. The lab supporting this course is open during working hours, and is available after hours and weekends by issuing them a combination to the lab door cipher lock.

An oddity has been observed over the years with lab manuals: the smartest students have the most problems with the lab exercises. Because they are smart, they are more likely to try to perform an exercise without reading it through first. They therefore will not fully understand the purpose of the lab, nor follow system-specific instructions, and often will paint themselves into a corner, or miss the fact that the answer to their problem is described in the next paragraph. Therefore, during the initial lab tour, they are encouraged to read the material first, and told why. Otherwise, only a few problems arise.

Years of experience have produced a lab manual that can be used by students independently, as described in the following sub-sections. Each of the sub-sections describe one of the three sections of the lab manual:

- Introductory material
- Lab exercises
- Appendices

Introductory Material

This section should not be very long or it will not be read. It should contain information that will be helpful to the student to successfully complete the assignments. For example, a warning that a printed number one 1 and the lower-case letter ell l look pretty close to the same thing, but that a command will not work if the wrong one is typed in. Another good warning is that a . at the end of a Unix command is not the end of a sentence, but is a necessary part of the command.

Lab Tutorials

As already established, an entire class cannot participate in the lab at the same time, forcing the students to work independently. Unless students perform an assignment during working hours they will not have any help available. Even then, they will have to go looking for help and may not find it if the instructor is not available. So the tutorials need to be self-guided and easy to follow.

Ideally there are lab assignments that correspond unit-by-unit and week-by-week with what is being taught in class. This can be difficult to do because some topics do not lend themselves well to an introductory tutorial, but some attempt should be made to synchronize them as closely as possible. For the introductory course, week seven of the quarter typically covers encryption, so a tutorial has been written that gives students hands-on experience with it. The real difficulty is producing an independent exercise that will almost always be problem-free.

One method for providing a problem-free manual is to format the written instructions in a way that is easy to follow. Figure 1 shows an example page with the format used in the NPS lab.

At a glance, the page in Figure 1 may not show any obvious benefits. However, note that when there is purely explanatory text,

is done by pushing the **Add** button to bring up the "Add Users and Groups" window. By default, this window only shows group names. To see user names, press the **Show Users** button. Select your instructor's login name and push the **Add** button, then select the type of access you wish to give him/her. Then press the **OK** button. The permissions do not take effect until you press the **OK** button again.

Commands

Exit the Properties window.

Using Explorer, copy the ***labxxx.txt*** file into the **U:\3600\Lab_DAC\Homework** directory where your instructor can verify your work.

Questions

Now look at the permissions on the new file at **U:\3600\Lab_DAC\Homework\labxxx.txt**.

Question 1: Why are the permissions different than the permissions you set for the file in your home directory?

Permissions are accumulated in NT. Therefore, if several groups are listed in an ACL, each with a different permission set, and there exists a user who is a member of all the listed groups, then his access to the associated file is the sum of all the group permissions. The one exception to this is if the user is a member of a group with "No Access", then the user is not given any access to the file, even if one of the entries gives the user "Full Control".

For example, assume a file has the following ACL entries:

Group Name	Basic Permission
Administrators	RWXDPO
Domain Users	RWX
Engineers	RXP
Lab Users	RX
Politicians	(no access)
Power Users	RWXD

Commentary and Explanations

Then assuming that a user is a member of the Domain Users and Engineers groups, his total access to the file is the sum of the two accesses: RWXP. If another user is a member of the same two groups, but is also a member of the Politicians group, then he has no access whatsoever to the file.

Figure 1. Sample Page Formatting for a Lab Exercise

meaning that the student will not find a step he is to perform, the paragraph will fit against the page s left margin with a bigger right margin to set it off from the other paragraphs. The paragraphs that contain commands and/or procedures for completing the exercise have a bigger left margin and fit up against the right margin of the page. In addition, actual steps, whether they are mouse actions or commands input by hand, are given in bold. Lastly, any questions dispersed throughout the tutorial are given in italics for quick reference.

Questions must be given and answers provided to ensure that the tutorials are performed. Each tutorial includes a series of questions that must be answered and turned into the instructor. This ensures that the student has performed the exercise. In addition, exam questions associated with lab exercises may be given.

The tutorial questions must be carefully chosen to make the student think. In addition, there must be some number of questions that can only be answered by completing the laboratory exercise. In other words, it should not be possible to answer all the tutorial questions by merely listening to a lecture or reading the textbook. Alternatively, there can be a mechanism to make sure the students used the assigned computer system during the assigned period of time, such as an audit record of login/logout, but this may require every instructor to have special privileges to read the audit records. Another way to ensure completion of the exercise is to require the student to leave evidence of some kind, related to the exercise. An example of this approach in the NPS lab manual is found in the Discretionary Access Control (DAC) tutorial, where students are required to create a file in a certain directory with specified permissions, which can later be checked by the instructor.

At the end of each tutorial, the same question is asked: How can this lab be improved? This has resulted in some good input, letting the instructor know which parts gave the students a problem, or just providing an avenue for good suggestions. Even without regular input, it must be understood that time and effort is required to maintain a lab manual.

Another problem to deal with is the fact that some number of students may be completely unfamiliar with the operating system and/or user environment. When the NPS computer security lab was first started, it was assumed that most students were familiar with Unix, because almost all the computer labs on campus were Unix-based. There was no need to do any handholding, or explain how to use an editor in Unix. With the Navy IT-21 Initiative that almost exclusively embraced Microsoft NT and Windows 2000, it is now assumed that no one knows how to do even basic Unix commands, such as listing the contents of a directory. Instead of trying to teach the students how to use one of the Unix editors, the tutorial steps requiring the creation of files are kept very simple, using the touch and echo commands, which makes life easier for the student and whoever is in charge of helping them when problems arise, like starting the vi editor and not knowing how to get out of it.

Appendices

The third and last section of the lab manual is the appendix. This is a good place to put what amounts to frequently asked questions. The NPS lab manual has four appendices: basic Unix commands; the various editors available, for the curious; a more detailed explanation of how the covert channel exercise works; and a more detailed explanation of how

steganography works with bitmapped (BMP) files.

Tutorial Subjects for the Introductory Course

There are currently nine lab exercises for *Information Assurance: Introduction to Computer Security,* covering eight topics. Table 1 shows the topics and the operating systems they are currently hosted on.

Topic	Operating System
Passwords	Windows NT
Discretionary Access Control	Windows NT
Mandatory Access Control	Trusted Solaris
Integrity	XTS-300
Viruses	Trusted Solaris
Encryption	Trusted Solaris
Covert Channels	Trusted Solaris
Steganography	Trusted Solaris

Table 1. Assignment Topics

With the exception of the access control exercises, all can be hosted on any modern OS. The Discretionary Access Control exercise works well on NT, but the Mandatory Access Control (MAC) exercises can only be implemented on selected operating systems. There is an effort underway at NPS to move the MAC-based assignments to another platform for a variety of reasons, including functionality and cost (Clark, 2000).

Application Support

One problem with developing lab exercises is that it may require the development and maintenance of some software to adequately support a topic. Maintaining the software includes an occasional bug fix, or the addition of new features, either of which requires time and expertise. It is advisable to write OS-independent software so applications can be easily moved to another operating system, if necessary.

Exercises in Other Courses

For the non-introductory courses, the lab exercises are moving targets as the instructor changes and/or the industry and governmental security environments change. An example of some exercises that were recently assigned to groups of students in *Information Assurance: Secure Management of Systems* are listed below:

- Install NT and configure it according to the *Secure Windows NT Installation and Configuration Guide* (DoN 1998).
- Install and use a trial (or free) version of a vulnerability scanner, either host- or network-based.
- Install a trial (or free) version of an Intrusion Detection System (IDS), either host- or network-based.
- Install a hacker tool and try to use it either on their own group s system or on another system, in cooperation with that group.
- Perform a risk assessment survey on one of the school s other computer labs.

Testimonials

With respect to the computer security labs at NPS, students have regularly indicated that the lab assignments have enhanced their learning experience. There is anecdotal support to the claim that some topics were not fully understood by some students until they had the opportunity to tinker in the lab through directed assignments. The following quote is an example of unsolicited and anonymous

comments written by students on instructor evaluation forms, with respect to the NPS security labs: Labs were very organized and beneficial. The most constructive computer labs for learning I have ever taken in any CS course. (Clark, 2001)

Summary

In summary, it is the experience at NPS that PCs are the best platform to set up a security lab for classroom use because they have more flexibility. There are some limitations to PCs that must be accounted for. Limited lab space can be overcome by creating exercises that can be performed by students independently. In addition, one PC can be used for multiple courses, even when different operating systems are required.

This paper has provided guidance for constructing tutorials and exercises to decrease problems that may be encountered by students. Without a carefully written exercise, students will be unable to work independently, which increases lab space requirements, and requires more time for lab assistants and/or instructors.

The lab assignments for the NPS information assurance courses have been carefully chosen to support classroom instruction and readings. The majority of the introductory courses are supported using software that was developed internally, with the understanding that there is an ongoing maintenance investment.

It is hoped that this paper will provide help to those who are looking for advice and guidance with respect to supporting a computer security laboratory.

References

CERIAS (2000), *Call to Action*, http://www.cerias.purdue.edu/events/accenture_cta_1q2001.pdf.

Clark, P. (2000), Policy-Enhanced Linux, *National Information Systems Security Conference,* Volume 1, pp. 418-432.

Clark, P. (2001), Author s personal files.

DoN (1998), Space and Naval Warfare Systems Command, Secure Windows NT Installation and Configuration Guide.

Goldberg, R. (1972), Architectural Principles for Virtual Computer Systems, *Ph.D. thesis, Harvard University*, Cambridge, MA.

Irvine, C. (1997), Warren, D., Clark, P., The NPS CISR Graduate Program in INFOSEC: Six Years of Experience, *National Information Systems Security Conference,* Volume 1, pp. 22-29.

Irvine, C. (1999), Amplifying Security Education in the Laboratory, *First World Conference on Information Security Education*, pp. 139-146.

Marsh, R., et al. (1997), *Critical Foundations: Protecting America s Infrastructures*, http://www.info-sec.com/pccip/web/report_index.html

Robin, J.S., Irvine, C. (2000), Analysis of the Intel Pentium s Ability to Support a Secure Virtual Machine Monitor, *USENIX Security Symposium*, pp. 129-144.

Spafford, E. (1997), *One view of a Critical Need: Support for Information Security Education and Research*, http://fas.org/irp/congress/1997_hr/h970211s.htm