The Virtual Invigilator: A Network-based Security System for Technology-enhanced Assessments

Nathan Percival *Member, IAENG*, Jennifer Percival *Member, IAENG*, and Clemens Martin

***Abstract*—The use of computer to assess learning is increasing at colleges and university as the use of technology on campuses increase. The challenge for the instructors at these institutions is to find a way to ensure the integrity of the assessments while still allow students to access network resources during the assessment. A variety of approaches exist that attempt to create a electronic environment that allows students to access only the resources that are permitted. Unfortunately it is nearly impossible to build a system that allow access to the set of resources that a instructor chooses while guaranteeing that no other resources is being accessed. This paper provides an alternate approach to the challenge of securing an assessment and presents a model of a system that can be used to ensure the integrity of the assessment even when unrestricted access to the network is provided.**

***Index Terms*—Technology-enhanced, security, online assessment.**

I. Introduction

In response to employers desires for technology literate graduates, technology-enhanced teaching is being implemented in a greater number of campuses and programs across the globe [1]. Students today are part of the "millennial generation" who have always had ubiquitous network access and portable communications devices such as cell phones, PDAs, and iPods. As theses students entered higher education institutions, the use of technology as a tool for learning has increased. Most schools now expect that students will communicate via email, use word processors, and will desire technology connectivity on campus.

Computers and the internet are continuing to become a more integral part of life a university campuses [2]. This increased prevalence has lead to an increase in the use of technology for assignments and laboratory situations [3]. The increased integration of the technology into courses is creating a need to assess the learning outcomes using those same technologies. For example, a course that teaching computer aided design using a piece of software needs to be able to use that same software for the examination. Simulation systems are also being used more to allow students to learn various concepts and these system can be used during assessments if a method is available to ensure students only use the allowed resources.

In today's global marketplace, engineering and other disciplines require the skills to thoroughly analyze an idea during the initial design phase, even before the creation of a prototype, in order to be competitive. To complete this type of analysis, students need to graduate with more hands-on skills with the tools they need to do this type of analysis [4]. Therefore, program that integrate the industry specific technologies into their curriculum and ensure through technology-enhanced assessment that students have mastered the application of theory through the software will provide a great advantage to their students as well as future employers.

While the ability to test the knowledge of specific software or design and analysis principal using the software is easy for an instructor to design, the ability to do so in an environment that provides a reasonable assurance that the students are not using the computer as a method of cheating is currently impossible. [5] identified the need for the development of a mechanism to deal with cheating during online assessments. Instructors need tools to ensure that testing of students have access to the network are secured to a level similar to traditional paper examinations.

This paper outlines a solution that provides the ability to monitor the electronic communications of students during assessments by providing real-time alerts of suspicious events and creating digital records of all communication events. The digital record of the network traffic can then be used as evidence of academic misconduct or can be examined later to look for other events. The record of the network traffic allows the instructor to examine what happened during the exam to find events that may not have been identified by the system. The examination could also allow the instructor to determine that suspected cheating was not actually in violation of the rules of the assessment.

The model creates a secured testing room for proctored examinations. The solution is intended for use with assessments that are not of a "memorized" nature but for assessments that require synthesis to complete and may require access to a variety of software to complete. The system acts in a manner similar to an person invigilating an assessment. There is no attempt within the model to lockdown the students computer nor the network. It monitors

Manuscript received June 30, 2008.

N. Percival is candidate for Master of Information Technology Security and a staff member at the University of Ontario Institute of Technology, Oshawa, Ontario L1H 7K4 Canada (phone: 905-721-8668; fax: 905-721-3370; e-mail: nathan.percival@uoit.ca).

J. Percival is with the University of Ontario Institute of Technology, Oshawa, Ontario L1H 7K4 Canada (e-mail: jennifer.percival@uoit.ca).

C. Martin is with the University of Ontario Institute of Technology, Oshawa, Ontario L1H 7K4 Canada (e-mail: clemens.martin@uoit.ca).

the actions of the students over the network while they are taking an assessment and searches for events that indicate that student are engaged in activities that are not permitted. Students are normally not allowed to communicate with people outside the room or each other and the invigilator observes the students actions to ensure that this does not happen. The Virtual Invigilator does the equivalent task in the electronic communications systems. It observes that events on the network to ensure that the students are not using the network to communicate.

The Virtual Invigilator may be used in situations where the computers for the assessment are already installed in the room `(such as a computer lab) or the students may be allowed to bring their own laptops. In the case where students are allowed to bring their own laptops, the content of those machines would not be in anyway monitored or checked, creating an open-book examination environment where the student is allowed to use what they bring with them but nothing else.

This paper will examine the existing technologies for securing computer-based exams and discuss the short comings of those solutions. The Virtual Invigilator model will then be presented including the overall concept, as well as the basic technical requirements. This paper will then explain why the Virtual Invigilator is a more secure solution that addresses a wider variety of technology-enhanced assessment conditions than the existing solutions. Finally, the paper will present directions for further development of the virtual invigilator system.

II. Securing Technology-Enhanced Assessment Environments

There are a number of challenges when attempting to secure examinations requiring a computer with network access. The type of challenges faced varies with the style and content of the test, as well as the specific network resources required. An examination may simply be conducted using multiple choice and short answer questions on a Learning Management System (LMS) such as Blackboard, Desire2Learn or Moodle. The examination may also be much more complex needing the use of software that requires access to a license server, access to a shared data space to access files or templates, and access to a system to submit the files that were created during the exam. The two assessment situations both require network access but the technical restrictions are very different.

The most basic method of securing an online assessment is a password. Instructors setup an online assessment in a LMS and have the LMS require that the student enter a password before they are allowed access to the assessment. This system has proven ineffective. In one case, a password released seconds before an assessment did not prevent 10 students not in the room from completing the assessment [5]. It is suspected that students were sharing the password with friends using email or instant messaging systems. Even the more complex restriction of access to certain IP address ranges does not always stop students, as the student may only need to be near the room in which the assessment is occurring and not physically in it. The significant shortcomings of password only solutions and basic IP address filtering has forced instructors to look for alternative methods to secure tests.

The systems that are currently used approach the problem of securing an exam by attempting to create an environment that is impossible to cheat in. These systems attempt to prevent students from cheating by making the assessment the only item on the computer they are able to access. Another approach is to physical monitor the student using addition hardware such as cameras and microphones that try to monitor the students' surroundings. Both of these systems require software to be installed on the students' computers and the monitoring system requires that the student have additional hardware installed on the computer. The requirement to install the security application on the computer means that the student taking the assessment is required to have run a specific operating system and is given full access to the code of the product allowing them to reverse engineer the source code.

A. Secured testing environment solutions

A number of prototype systems have been proposed to create a secure testing environment. [6] proposed a system using a bootable zip disk and [7] created a method that worked for a CDROM. An alternative approach to limit network activities using a distributed firewall was presented by [8]. While all of these models could provide security for a specific type of examination on a specific hardware and operation system platform, they are not robust enough to handle most assessments.

Currently, the two best-known commercial products that create a secure environment for conducting assessments through a LMS are Securexam Browser from SoftwareSecure [9] and Respondus Lockdown Browser from Respondus [10]. Both of these products attempt to provide a secured web browser that only allows access to assessments provided by a single LMS. To achieve this goal, the products attempt to take control of the entire system. Securexam Browser disables copy and paste functionality, blocks access to the task manager, disables launching scheduled tasks, and prevents access to any other application [11]. Securexam Browser currently is limited to the North American Versions of Windows XP or Vista and works with only the two most common LMS, WebCT and Blackboard. Securexam Browser. The company states that there is a MacOS version but provides no additional details about this version of the product [11]. Securexam Browser uses an algorithm that takes the title of the assessment in WebCT and generates a password with the title as the input. The methods used to determine the password is not publicized but the code to calculate the password must be in the application that is installed on the end user workstation and hence can be reverse engineered. If a student can get the password to the exam either by reverse engineering the software or access the online password generator provided by Secure Software for Securexam browser [11] (or by any other means), then the student is able to take the exam without using Securexam. There is no method for the professor to detect this security breach based on the exam submitted by the student. As with all password only security systems, the entire security of Securexam is based on the assumption that the students will

not be able to find the password to access the exam except by using their product.

Respondus Lockdown Browser has many of the same limitations as Securexam Browser. The product only operates on Windows XP, Windows Vista and MacOS. The system limits student access by blocking access to the task manager, copy and pasting functions, and function keys. In addition, Respondus Lockdown browser needs software to be installed on the server for some LMS. This software allows the LMS to verify that the assessment is being taken in the Respondus Lockdown Browser [12]. This function could also be beaten by determining how Respondus Lockdown Browser identifies itself and customizing another application to respond in a similar manner.

There is another method for breaking the security of both of these products. This is to run them in on a virtual machine. Currently there are more than 15 virtual machines able to run Microsoft Windows listed on Wikipedia [13]. Based on testing of a recent version of Securexam, it was determined that is detects virtual machines from VMWare and Microsoft Virtual machine. Through testing of Securexam Browser, it failed to detect VirtualBox and Virtuozzo. It is likely that there is a similar problem with Respondus Lockdown Browser detection of a at least some virtual environments. As the number and design of virtual environments continues to grow, it is virtually impossible for commercial software to ensure that it can specifically address each format as soon as it is available on the Internet.

Both of these software base solutions are limited to a total of three different LMS. These commercial products do not support the two main open-source LMS, Moodle and Sakai [11, 12].

An additional software package from SoftwareSecure allows the use of a word processing environment while locking out all other software packages. While both Securexam and Respondus Lockdown Browser systems support these very specific types of testing, they do not support the ability to use additional applications that are used within a course such as Computer Aided Design software (CAD), computer programming envinronements, statistical analysis programs, or any other application that may be taught as part of a course. This limits the ability of a professor to be able to conduct secure technology-enhanced assessments to only the formats supported in the LMS (multiple choice, short answer, and essays).

B. Video Monitoring Solutions

Video monitoring of student actions has been proposed as alternative method for securing computer-based testing. [14] has developed prototype of these type of security system. This type of secured environment is an integrated solution involving both hardware and software components. The system attempts to monitor the actions of the student writing the exam, as well as both the audible and visual environment around the student. SoftwareSecure is currently developing such as system [11]. The system attempts to detect any "abnormal" changes in the environment and then records them for review by the professor at a later time. The problem for this type of system would be determining programmatically when the activity is suspicious. The level of sound and motion change in a quite room, an office cubical, or a coffee house would all be drastically different, making the detection of only suspicious changes quite challenging. The recorded environment around the student could include anybody or anything around them. This may therefore accidently invade other's privacy by recording their actions without their knowledge and the student writing the exam would have no power to stop this.

This type of environment is only useful when a student using the system can be placed in an isolated location. It would be virtual useless when used in a room with a number of other students nearby taking exams because the motion and noise nearby students would constant be identified as interesting event. This would render the system nearly useless as the amount of data that a professor would need to review would be overwhelming. In addition, the bandwidth used to send the video could result in slower response time for a student's computer and might increase the chances of other technical failures. The video monitoring solution requires equipment for every student taking an assessment. This leads to additional costs for the students. The equipment would also need to be installed prior to the assessment by each student on their laptop or onto the computers if in a lab setting. If the equipment was installed in a lab, it would either need to be installed and removed for an assessment or there would be an increased risk that the equipment could be damaged or stolen leaving the lab short of equipment during an assessment.

C. Summary

Both software that attempts to provide a secure testing environment and systems that provided video monitoring of students taking exams have serious shortcomings. Most solutions require the installation of software on the student's machine, which could be reversed engineered and hence defeated. In addition , the solutions are limited to one or two operating systems such as Microsoft Windows and MacOS. Support for any other OS, such as Linux or Solaris that are used in teaching Computer Science and Software Engineering are not securable using these technologies. Of the solutions presented, only the video monitoring systems are capable of allowing students to use arbitrary software applications or a specific set of websites. To secure a heterogeneous set of computer requires a technical solution not based on the hardware or software used by the students taking the test.

III. The Virtual Invigilator System

The Virtual Invigilator is a system designed to secure the electronic communications of an assessment environment. It is designed to assist in the proctoring of an assessment in a controlled location such as a classroom with one or more invigilators monitoring the activity within the room. The system assists by monitoring all network traffic from the computers within the classroom, recording it, and simultaneously identifying in real-time any activity that is suspected of violating the rules setup for the exam.

The system is designed to be hardware and software independent with the only technical requirements being that the networking equipment support monitoring of the traffic.

Students with any computer running any operating system and any type of application can be monitored using the system.

The Virtual Invigilator is unique in that it does not attempt to create a perfectly secured testing environment on an unsecured piece of hardware nor does it attempt to directly monitor the actual human that is completing the exam using technology. The students are allowed to use their local computers in any way they see fit and the monitoring of human actions is left up to other humans who can still notice suspicious behavior better than any proven and commercially available computer system. The Virtual Invigilator is instead acting to detect cheating by watch the network communications of a student. The Virtual Invigilator allows students to access the network resources that they need but detects when the student start to use the network in a way that is not allowed. The Virtual Invigilator supports the traditional invigilation process by providing the capability to monitor the portion of the assessment environment that cannot be easily monitored by traditional observations methods.

This monitoring can be compared to an exam where the students are allowed to bring in the course textbook but not the slides from the lectures. If a student hides the lecture slides within the textbook it is quite possible that the invigilator will notice this and hence find the students attempts to beat the system. There is no process in place in this situation that attempts to keep the student from being able to cheat. Instead there as a system in place that has a high likelihood of detecting the attempts be a student to cheat. It is this model that the Virtual Invigilator is attempting to extend to assessments that require students to have access to network resources and computer-enhanced assessment environments.

A. General goals and assumptions

The goal of the Virtual Invigilator is to do an equivalent job for network traffic as a real invigilator does for actions of the students in the physical room. This means that the Virtual Invigilator needs to monitor all network events and determine that which events are suspicious and require further investigation. The system makes no attempt to prevent students from cheating or otherwise violating the rules of the assessment but instead detects any action of a student that is in violation of the rules under which the assessment is being written.

The overall model of the Virtual Invigilator system shown in Figure 1.The system is designed to operate on an local Ethernet network and expects that all acceptable traffic between the student's computer and other system will us Internet Protocol (IP) and that any other intercomputer traffic is unacceptable. This could be altered by the professor if a particular network need for other protocol support exists. The monitored network only needs to include the network traffic that is either to or from a machine that is in the room being proctored. The network equipment must be capable of providing a copy of all network communications to a single network such as an Remote Switch Port Analyzer (RSPAN) port [15]. To best support the Virtual Invigilator system, the RSPAN port used to link the Virtual Invigilator monitoring system to the network should be faster than those used by the students, preferably a Gigabit Ethernet port. The higher speed port allows the Virtual Invigilator system to capture more data then any one machine in the system could generate helping to ensure that all network traffic is successfully recorded.

It also assumes that the volume of network traffic is small enough to be completely recorded by the system. The effectiveness of the system is not diminished if the system is unable to record all traffic as it will still capture a significant portion of the traffic. Because the system is based in the network, the students taking an assessment would not be able to control which traffic the network fails to deliver to the Virtual Invigilator. To accomplish this, they would first need to compromise the security of the network equipment and this compromise would be detect by the Virtual Invigilator if it was attempted during an examination.

The Virtual Invigilator also access the switches, through Simple Network Management Protocol (SNMP) to identify the physical port to which a computer is connected by identifying the port on the switch that is associated with Ethernet Media Access Control (MAC) address for that computer. The Virtual Invigilator system will need to have a mapping of the physical switch port to actual location in the assessment room to provide to the invigilator if any suspicious activity is found.

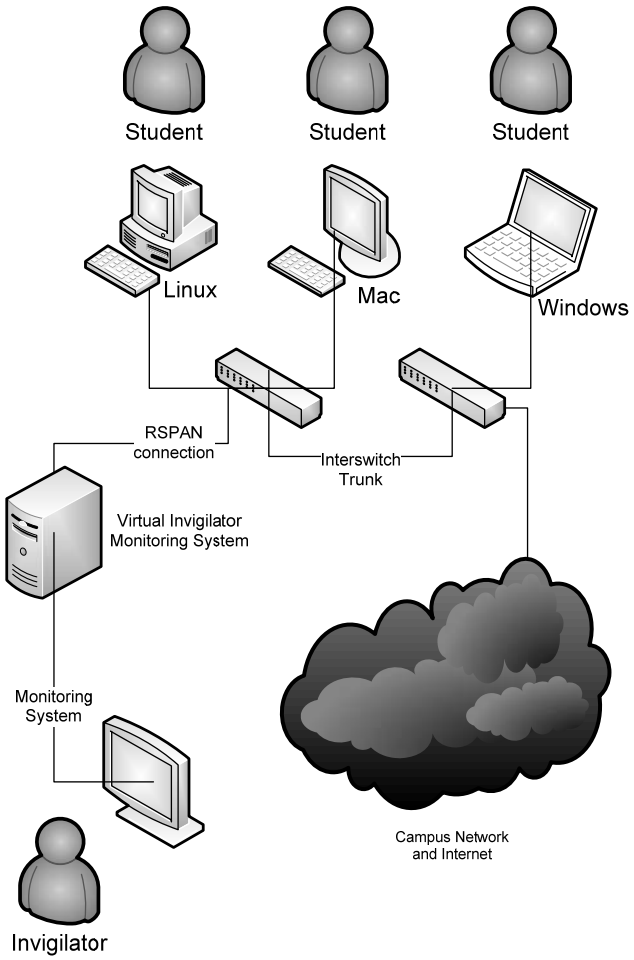

Figure 1 – Model System

The system assumes that the parameters of an assessment can be clearly defined for input into the Virtual Invigilator system. The instructor needs to be able to use a predefined

access level (such as only the LMS) or be able to list the sites that the students should be allowed to access. The ability to detect unauthorized events is limited by the specificity of the rule that it is told to operate under.

The Virtual Invigilator is highly customizable and will allow an invigilator to be easily setup through the invigilator interface with the criteria by which they are to be alerted to suspicious activity. The system has default setups that allow either no traffic or any traffic. Additional automatic setups could include only access to a LMS, full network access but no peer-to-peer communications or other communications systems (such as email or instant messaging). The system will also allow the invigilator to setup other adjustments such as allowing specific additional websites or access to necessary license servers.

To allow the invigilators to select default setting, the Virtual Invigilator has the ability to be customized at a technical level by staff that are familiar with the technical requirements for the systems and applications that commonly need to be accessed. This allows the Virtual Invigilator system to be customized for any campus and again for any room by the technical staff. This allows the invigilators to have a simple interface for use when preparing a room for an assessment. The technical customization system can also be use by an instructor to prepare the room for an assessment where the instructor has a complex set of requirements and is familiar with the technology.

B. Architecture

The network packets are captured by the system and simultaneously processed in two separate ways. First, all traffic is stored to disk for possible later use. Second, the network traffic is processed to identify any packets or set of packets that are violation of the policies that have been setup by the professor in the Virtual Invigilator.

The storage to disk of the data allows the information captured by the Virtual Invigilator to be used to document the events that occurred on the network. Then, if students are caught by the system, even if they claim that the system detected the network traffic in error or that the traffic captured was the result of someone having hacked into their computer this can be fully investigated. Since all network traffic and not only the traffic that was identified as suspicious are recorded, the claims of the student can be easily verified or disproven. The record of the network communications can also be used during the formal appeals process as evidence to support the case against the student who was caught by the Virtual Invigilator.

The second way that the packets are processed is by a real-time event recognition engine. This system will look at individual packets and sets of packets in detail to determine if the packet is acceptable given the rules that have been setup for the assessment. The concept of analyzing data packets as they occur and identifying certain packets or set of packets that are of interest is commonly done on most networks today and using Intrusion / Incident Detection Systems (IDS) for Internet-based security attacks. Most of these systems inspect network packets as they arrive and compare those packets to a set of parameters that help identify suspicious network traffic [16]. Most of the IDS system rules currently used are designed to detect network traffic that is malicious in nature such as attempts to hack into a network.

The Virtual Invigilator uses a set of rules that is designed not to detect malicious behavior but to detect behavior that is contrary to the rules for the assessment that it monitoring. Similar to traditional IDS, the detection process is designed around the assumption that anything that is not explicitly allowed is suspicious traffic and the system should be alert. Contrary to the software based secure testing system, it is assumed that for most assessments, it is much easier to specify explicitly what is allows than to attempt to list every possible thing that is not allowed. The use of new application or operating systems, new hardware and even the use of virtualization do not have a significant impact of the Virtual Invigilator. Changes to any of these part of a room used to conduct assessment will at most require an adjustment of the rules to handle slight variation in the way these systems operate.

C. Security features

To monitor the testing environment, the Virtual Invigilator uses the RSPAN port capability of the network switches in order to capture a copy of all network packets. All network ports within the room are configured so that a copy of all traffic, in either direction is mirrored to the RSPAN port. This ensures that all network packets can be captured, regardless of the configuration of the end machines.

The RSPAN port will be connected to the Virtual Invigilator monitoring system and will record all network packets to disk and conduct the analysis of the network traffic according to the rules that have been setup for the assessment as shown in Figure 1. This network interface will be setup in 'promiscuous mode' to allow it to capture all network traffic regardless of the intended destination. The assessment network usage rules will be processed in sequence and any network traffic that is found to be acceptable by a rule will not be passed on for further processing. When network traffic fails to be classified as acceptable by any rule, the system will consider it suspicious and will send a copy of the suspicious packet to the Virtual Invigilator Management system. In addition, the identification of the packet as suspicious will be logged to the recording of the network traffic being made on the monitoring system so that it can be reviewed later. The communication with the management machines is over a separate network from the network being monitor by using a second network interface on the monitoring system computer. This allows the monitoring system to communicate with the management system on a network that is not being monitored, thus ensuring the traffic between the Virtual Invigilator systems does not cause alerts on its own communication.

Once the monitoring system receives a suspicious packet is will provide notification to the invigilator through a visual alert. The monitoring system will analyze the packet provided by the monitoring system and provide as much information as possible to the invigilator about the suspicious network activity. The system will identify to the invigilator the location, based on the network port in use, of the computer creating the suspicious network traffic. The invigilator can then investigate what is happening in the room to make a determination of whether the packet that was

flagged suspicious is truly a violation, is a false positive and hence is not a violation, or that the network traffic requires further investigation. The management system will allow the invigilator to flag all suspicious packets as any of these three classifications. The resulting determination by the invigilator will be logged into the monitoring system for analysis after the exam.

After an assessment is completed, the instructor or invigilator can use the management system to examine more thoroughly the suspicious activity that was found by the Virtual Invigilator. This may allow for the identification of students whom may have not been detected by the original rules but did violate the rules of the assessment. It may also assist the instructor in determining cases of collusion that might be hard to find by simply grading a large set of assessments. The false positive results can also be examined by the instructor and by the technical staff to determine if there is some network traffic like an automatic update system that could be identified as acceptable and added to the assessment rules to allow for more accurate detect during future assessments.

IV. Conclusions and Future Work

This paper has provided a model for a new system to secure the proctored classroom assessments that require network access. It provides a method that does not attempt to create a secured environment that the students must use, but instead uses a method similar to the proctoring process where the actions of the students completing the assessment are monitored to ensure that their actions conform to the rules of the examination. Because there is no attempt to prevent actions, the system is much more flexible as it can be used with any combination of hardware, software, and network need.

The Virtual Invigilator model provide a strong basis on which to build a prototype system for securing a classroom for network connected proctored examinations. The Virtual Invigilator system needs to be setup so that it can support a large variation in the requirements of the assessment as well as large variety of abilities of the invigilator to configure the system to meet their needs.

To build the system further work is underway to create the rules that will affect all exam such as DHCP and DNS rules. These rules will need to be specific enough that they do not provide false positive results by generic enough that they can be used by the system in any context.

A test system will be implemented and used during an actual assessment at a laptop-based campus to determine the Virtual Invigilator's effectiveness. In addition the test system will test the network hardware required to collect all network traffic from a group of students and develop an estimate the number of students that can be monitored using a single implementation of the monitoring system.

Finally, a method of detect communication between students where there communications is done using a Wiki, Blog or other similar dynamic webpage must be created. The posting of content by a student and the reading of this content needs to be correlated since the reading of a Wiki page may not be a violation of the rules of the assessment but the use of them to communicate with other students would be a direct violation of the assessment rules.

References

[1] S. Elwood, C. Changchit and R. Cutshall, "Investigating students' perceptions on laptop initiative in higher education : an extension of the technology acceptance model," *Campus-wide Information Systems*, vol. 23, pp. 336-349, 2006.

[2] D. G. Brown, J. J. Burg and J. L. Dominick. "A strategic plan : for ubiquitous laptop computing," *Communications of the ACM*, vol. 41, pp. 26-35, January. 1998.

[3] S. L. Howell, "E-Learning and Paper Testing: Why the gap?" *Educause Quarterly*, no 4, pp. 8-10, 2003,

[4] H. R. Jacobs. The utilization of a mobile computing environment in undergraduate education. *Proceedings of Frontiers in Education Conference, 1996. 26th Annual Conference*, 1996, pp 656-658.

[5] A. B. Campbell and R. P. Pargas, "Laptops in the classroom," *Technical Symposium on Computer Science Education : Proceedings of the 34th SIGCSE Technical Symposium on Computer Science Education*, 2003, pp. 98-102.

[6] C.C. Ko, C.D. Cheng. "Flexible and secure computer-based assessment using a single zip disk" *Computers & Education*, vol. 50, pp 915-926, 2008.

[7] M.C. Carlisle, L.C. Baird III. "Design and Use of a secure testing environment on untrusted hardware" *Proceedings of the 2007 IEEE Workshop on Information Assurance*, pp. 349-354, 2007.

[8] C. Pan, K, Yang, T, Lee. "Secure Online Examination Architecture Based on Distributed Firewall" *Conference on e-Technology, e-Commerce and e-Service, 2004. EEE '04. 2004 IEEE International*, pp. 533-536, 2004.

[9] Software Secure Inc. (2008, Securexam - securexam browser. 2008(6/28/2008), Available: http://www.softwaresecure.com/browser.htm

[10] Respondus Inc. (2008) Respondus lockdown browser. 2008(6/28/2008), Available: http://www.respondus.com/products/lockdown.shtml

[11] Software Secure Inc. (2008) Securexam - frequent questions. 2008(May 13), Available: http://www.softwaresecure.com/faq.htm

[12] Respondus Inc. (2008) FAQs about Respondus LockDown browser (LDB). 2008(June 4), Available: http://www.respondus.com/lockdown/faq.shtml

[13] Anonymous (2008, June 17). Comparison of virtual machines. 2008(June 19), Available: http://en.wikipedia.org/wiki/Comparison_of_virtual_machines

[14] C.C. Kong, C.D. Cheng. "Secure Internet examination system based on video monitoring" *Internet Research: Electronic Networking Applications and Policy*, vol. 14, no, 1, pp. 48-61, 2004.

[15] RSPAN Cisco Systems Inc., (2008) Remote SPAN (RSPAN) - Cisco Systems. 2008(June 10), Available: http://www.cisco.com/en/US/tech/tk389/tk816/tk835/tsd_technology_support_sub-protocol_home.html

[16] P. Innella. (2001, Nov 16). The evolution of intrusion detection systems. 2008(6/28/2008), Available: http://www.securityfocus.com/infocus/1514