Small-Scale Cyber Security Competitions

Mike O'Leary, *Towson University*

Abstract – Cyber security competitions are becoming more common and more complex, and faculty interested in hosting a small scale event may be intimidated into thinking that they necessarily require significant investments of time and resources. In this paper, we describe how we a single faculty member has been able to run a number of small scale competitions in a variety of formats, ranging from class and club level up to competitions with five different participating schools.

Index terms – Cyber Security, Collegiate Competitions, Cyber Defense Exercises.

I. Introduction

Cyber security competitions are becoming increasingly prominent parts of the landscape of cyber security education. The most well-known competitions are probably the National Collegiate Cyber Defense Competition (NCCDC) with its precursor regional events [1]; and the UCSB International Capture the Flag Competition (iCTF) [2]. These are both large national and international competitions with scores of institutions and hundreds of students. What if someone wanted to run a small event for a club or for just a few area schools? Is a large infrastructure required? Is a large team of technical assistants necessary? To use an analogy, if these large competitions are the equivalent of a professional NBA basketball game, is it possible to create the equivalent of a simple game of pick-up basketball?

For many years, the author has taught a hands-on capstone course in computer security at Towson University [3] where security competitions take a prominent place; he has also run numerous club-level exercises for the university's cyber-defense club, and run a number of informal competitions involving as many as five schools.

All of these competitions were developed and run by a single faculty member without requiring significant investment of staff time and resources.

This paper describes our experiences running these small scale events and provides recommendations for others who want to build their own comparable environments.

Mike O'Leary, School of Emerging Technologies, Towson University.

II. Cyber Security Competitions

The National Collegiate Cyber Defense competition is now a giant event; in 2012 there will be ten different regional competitions [1] to determine the finalists. The regional events themselves are also growing; the mid-Atlantic region has 25 participating institutions, while the Midwest region has grown so large that they hold nine individual state level qualifying rounds before the regional final.

The genesis and evolution of the NCCDC is well documented, beginning with White and Williams [4], Conklin [5] and White and Dodge [6]. The different regional competitions have also been described; Rosenberg and O'Brien [7] discuss the mid-Atlantic regional, Whitman and Mattord [8] describe the Southeast regional, while Carlin, Manson and Zhu [9] discuss the development of the Western regional competition.

These competitions require the students to demonstrate defensive and administrative skills. The students work in teams of roughly eight students and each of the student teams is given an identical network to defend from a group of outside attackers, called the Red Team. While defending, the student teams must keep their network operational and also remain responsive to various required tasks, called business injects. The final competition score is a combination of how well the students maintained the availability of their network, how well they defended their network from the actions of the Red Team, and how well they performed on the assigned business injects. Student teams are not allowed to perform offensive activities; they may not attack the Red Team or other student teams.

Bei, Kesterson, Gwinnup and Taylor [10] share how the University of Washington Tacoma and Eastern Washington University prepared for the regional competition, while Sroufe, Tate, Dantu and Celikel [11] describe the experiences of University of North Texas at their regional competition.

The UCSB International Capture the Flag Competition of Vigna *et. al.* [2, 12, 13] has been running since 2003. It started out as a traditional capture the flag event modeled after the DEFCON competitions, and has since evolved into a treasure hunt model. This is an attack-only model where student teams of roughly fifteen are presented with

ISBN 1-933510-95-1/$15.00

identically configured networks isolated from the network of other teams. The student teams have to compromise various components of their network in sequence to reach their ultimate goal. The contest is run virtually, with teams from around the world connecting to a VPN for the exercise. In 2009, 56 different teams participated; this number jumped to 72 teams and almost 900 students for the 2010 event [2].

A more specialized but perhaps formative exercise has been the Military Academies Cyber Defense Exercise (CDX) [14, 15, 16]. This is similar to the NCCDC competition in that it is defensive, where student teams defend their network from attack from an outside Red Team. In the CDX, the Red Team is composed of professionals from the NSA and the military. One difference is that the student teams build their own network to specification and within a budget, rather than being presented with an existing network.

These competitions are also beginning to influence curriculum and curriculum design, especially in the military academies. For example, the Air Force Institute of Technology rearranged its course sequencing and developed new courses to better prepare for the CDX [14], the Naval Academy added material on computer and network forensics to existing courses [15], while the Information Assurance course at West Point is designed around the CDX [16]. Conklin [17] describes some of the impact of competitions on curriculum at the University of Texas San Antonio.

III. Our Environment

Central to our ability to run small-scale competitions is our environment, especially our isolated computer security laboratory. Such laboratories are common in cyber security education, and their development has discussed in many places [18-21]. In our laboratory, we have 24 computers arranged in four tables of six computers each. These each run VMware and have been upgraded with additional memory to aid in its use. Today's machines have 16 GB of memory each which allows us to comfortably run six to eight virtual machines on each host. The use of virtual machines is now common in security courses and security competitions [22].

These machines are networked together on a common isolated network. We also have a single file server from which students can download preconfigured virtual machines, installation discs and other software provided by the instructor.

Students do not have access to the underlying network hardware and do not have access to privileged accounts on the VMware hosts. Because the student activities are limited and can only affect the virtual machines, we can and do use the laboratory for a class at 3:00, run a short 75 minute competition at 5:00, and then use the same laboratory for a different class at 7:00.

Setting up a competition in this environment means developing the virtual machines and deploying them to appropriate hosts in the laboratory, a process that can easily be handled by a single faculty member. This advantage in ease of use is gained at a price though. By restricting our exercises to virtual machines, students do not have the opportunity to get deeply involved in networking and network hardware. Though it is possible to simulate more complex networks with virtual networks inside VMware Workstation, this approach does not allow the students to experiment with hardware firewalls, routers, and switches.

Our competitions are designed specifically for the students that we have at our institution. At Towson University, we have a long standing undergraduate concentration in our computer science major in cyber security; we also have a track in our Master's program in cyber security. The students who participate in our activities are usually junior and senior undergraduates in the security track, with a leavening of graduate students and underclassmen.

These participating students have a solid background in networking, operating systems, and databases. They are familiar with the Linux environment, though not all of them are highly skilled. Most students are still learning about more complex web applications; they are also unlikely to be able to be able to meaningfully craft a new exploit for something like a known buffer overflow in an exercise setting.

We often invite area schools to participate in our events, including two year schools, and have found that they have comparable skills

Our events have been designed for students with these roughly homogeneous skill sets, but it should be noted that there are competitions designed for a more diverse audience of participants, like CANVAS [23] and the MIT Lincoln Laboratory capture the flag competition [24].

IV. Our Competitions

There are a number of models appropriate for small-scale security competitions and hands-on exercises. Perhaps the most common approach is a variant of a capture the flag exercise; this is the approach used by West Point in their scrimmage to prepare students for the CDX [16]. Similar approaches are used by Walden [25], by Aman, Conway and Harr [26], in the "Blunderdome" at the University of Tulsa [27], and in the NetSecLab at Georgia Tech [28]. Other models include the "role-game of the Internet" of

Catuogno and De Santis [29], the "live penetration audit" of Aman [30], and various social engineering competitions [31, 32, 33].

The author has taken three different approaches to cyber security competitions. The first approach is used primarily in our capstone course in computer security [3]. Because these exercises are done in a classroom setting, they are tied directly to the course objectives, and the focus is primarily on pedagogical issues rather than on the competitive side.

In this approach, the students in the class are divided into four teams of three to six students each; this is in line with the architecture of our classroom laboratory and allows each team to work together on a single table. The teams are presented with functional requirements for a network. The students build a network of virtual machines to those specifications using the tools and techniques previously discussed in class. Because the exercise is phrased in functional rather than operational terms, the student teams need to decide on their own the best way to implement the required functions.

In a simple exercise, the student team might need to develop a local DNS architecture, set up one or more Windows domains, set up a remote file server, tools for remote access (*e.g.* SSH) and construct a centralized logging solution.

More complex exercises require the team add additional network services like web servers, databases, and fully featured web applications. We make use of Zen Cart [34], an open source e-commerce web site in part because it comes complete with a demonstration store. Additional defensive tools also become required elements, including intrusion detection systems and proper firewalling. Our laboratory design precludes us from having the students work directly with the network hardware, so they cannot use hardware firewalls like the Cisco ASA. However, we show the students how to create virtual machines with multiple network interfaces. Using tools like IPFire [35] and the virtual networking in VMware Workstation, the students can build more complex network topologies entirely within VMware including DMZs, firewalling, filtering and virtual private networks.

The students typically have one to two weeks to set up their network to prepare it for the live fire portion of the exercise; that usually happens in a single 75-minute class period. Prior to the live fire, student teams provide the instructor with a complete set of credentials for their network. Portions of these (non-root, non-administrator) credentials are shared with the other teams. However no team knows which of its credentials were shared, nor do they know which team(s) might have received credentials to their network. For this reason, student teams cannot assume *a priori* that connections made to their network are unauthorized and/or hostile.

During the exercise, each team needs to verify that the opposing teams are correctly providing the required services, using the provided credentials. They can then attempt to escalate their privileges and execute any pedagogically appropriate attacks. In the early exercises, students can gain credit for attacks that are little more than information gathering, by mapping out other teams networks and determining the versions of the various services that are provided. In later exercises, students become more comfortable with more sophisticated attacks, often using the Metasploit platform [36].

Though students have a great deal of flexibility in their attacks, they are required to be pedagogically appropriate. Some types of attacks, like denial of service attacks through network flooding or gratuitous damage to an opponent's system by deleting logs or file systems are always prohibited. Other attacks are inappropriate depending on the place in the course. For example, because the teams are on the same isolated local network, ARP spoofing is a very effective technique. However, it is disallowed in the early part of the course, at least until the students learn enough about tools like arpwatch and more sophisticated intrusion detection systems to be able to identify when these malicious activities occur.

Because these exercises are part of a course, student performance is assessed via a written report due a week after the live fire exercise. The report covers how the team set up their network to address the functional requirements, the activities the students performed during the exercise, and how well the team was able to determine what happened on their network in their after-action analysis. Students are graded on these criteria and the overall quality of the written report.

One advantage of this approach to crafting a competition is that it does not require the organizer to build out the network; this is done by the students themselves. This makes this approach particularly suitable for small-scale events with one or a small number of organizers. One issue with this approach however, is that the organizer can never know what the true state of the network actually is. This makes grading the students' reports more challenging, as even the instructor does not have "the answer". This is ameliorated by requiring the students to submit complete documentation of the network that they used and to run one or more network scans as the instructor to validate and verify the students' statements.

The author uses a different type of competition to help our student team prepare for external competitions like the NCCDC. It is much more work to create and develop; typically it is run no more than once in a year.

In this competition structure, the organizer provides each of four teams with a prebuilt network consisting of 6-10 virtual machines per team. Each team gets essentially the same network; only the addresses and names (host names and domain names) vary. The network includes a mix of Windows and Linux machines and a range of services, typically including a Windows domain, DNS, and one or more non-trivial web applications. The teams also get one or more machines, usually Backtrack Linux [37] for offensive purposes.

The machines are deliberately misconfigured to make them easy for the student teams to exploit. The goal of the student teams is to both defend their networks as well as to exploit those of the opposing team. Because the machines are identically (mis)configured, students have the opportunity to play offense while playing defense. Indeed, when they locate a flaw on their own network, they know immediately that this flaw also exists on the corresponding machine for all other competing teams.

The competition usually runs for a full day of six to eight hours on a weekend for two to four teams of four to six students. The students do not have a chance to see the machines before the exercise begins, so there is a learning curve

Scoring is deliberately kept low-key and informal. As the competition progresses, student teams simply let the organizer know what they have done: found a vulnerability on their own network, remediated a flaw on their network, or successfully exploited a system belonging to another team. Points are then assigned and announced to the group at large. To add to the excitement and enjoyment of the students, we have found it valuable to also give points for "style". These are purely discretionary and awarded by the organizer for activities that were particularly clever or stylish. Student participants really appreciate these, and often go out of their way to find creative ideas that might potentially be worth extra style points. Because of the informal nature of the competition, in line with a backyard sports contest, no prizes are awarded save for bragging rights.

Though fun and enjoyable for the students, this type of competition is significantly more work for the faculty organizer. The most difficult portion is creating the virtual machines that the students will use and to populate those machines with flaws that are simple enough for students to find in a one-day competition, but not so simple that they are immediately located. Some of flaws used in recent editions of this competition format have included:

- Pre-sharing keys for SSH, letting students SSH from machines on their network to corresponding machines on their opponent's network without needing the passwords.
- Trojaning Metasploit; in particular configuring the system so that running either the command line tool (msfcli) or the console application (msfconsole) results first in a call to netcat (renamed to /bin/mysql) that opens a netcat listener on TCP/3306 bound to a root shell before the Metasploit command runs.
- Configuring a VNC server like UVNC to start silently at boot (*e.g.* via HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run), using the same set of credentials for each machine (or simply not requiring credentials at all).
- Modifying the path variable to allow common commands like netstat to be trojaned so that they display false or misleading information.
- Misconfiguring FTP servers to run as root or administrator and allow anonymous read/write access.
- Putting tools like PHPShell on the web server.
- Adding users with various privileges to various services including the database systems, web applications, and the core operating system. This was often accomplished through the use of cron jobs or at tasks so as to occur after the exercise start.
- Using xinetd to bind a root shell to a port.
- Changing the permissions on key files like /etc/shadow.

The first few times this approach to a competition was tried, the organizer provided each team the credentials needed to log in to all of their systems at the start of the event. In the most recent instance of this competition however, students were only provided with credentials to one machine. That machine contained a file (executable/.pdf/.doc/.docx) with instructions on how to log on to second machine which contained a file with instructions on how to log in to a third and so on. Some of these files, but not all, were malicious, containing *e.g.* Metasploit payloads. Some student teams decided that it was more important to immediately gain access to their systems and clean up the resulting problems later. Other teams first tried to decompile the files to see if they could find the credentials without actually running the file; when that failed, they downloaded and installed an antivirus solution before opening the file.

Students have found this type of competition to be particularly enjoyable as they get the chance to engage in both offensive and defensive activities.

The problem with this second type of exercise is that it requires a considerable amount of work for the organizer. Setting up the first copy of each virtual machine and configuring it with a handful of meaningful holes and misconfigurations takes two to three hours per machine; it

then usually takes another hour or so to make replicas for the other teams. With six to ten virtual machines per team, this comes to twenty to forty hours of effort- just within the time limitations of a single faculty member. Moreover, we have found that we have been unable to meaningfully recycle the virtual machines between competitions, in part because many of our students participate in our events for more than a single year

In our third kind of competition, the students build their own network to precise specifications. A typical exercise might ask the students to create

- Windows 2003 Server (Not R2), DC, DNS, RDP
- Windows 2008 Server (Not R2), DC, RDP, Exchange
- Fedora Core 4, DNS, SSH, MySQL (4.1.1)
- Ubuntu 8.04, SSH, Apache, PHP, MySQL, Joomla 1.5.12
- Fedora 6, SSH, vsftpd 2.0.5
- Windows XP (No SP) joined to the domain
- Windows Vista SP1, joined to the domain.

Each system is required to have a web browser, e-mail client, pdf reader and Adobe flash installed, with specified version numbers for each component. By specifying the versions of the software, we ensure that the student teams' networks are comparable. Software versions are also selected with a very careful eye towards the constellation of known vulnerabilities; this lets the organizer calibrate how difficult it would be for an attacker to compromise the system using existing known exploits.

Our students have access to the Windows software through the Microsoft Developer Network Academic Alliance (MSDNAA); the Linux systems are all open source. We have found sites like Oldapps.com [38] to be invaluable resources, as they contain older released versions of a large number of common applications, primarily but not exclusively Windows applications.

These competitions have been run where the student team has a significant lead time to develop their network, on the order of two or three weeks. This model has been preferred when the student teams are still relatively new and our interest was as much in helping the students learn how to secure the systems as in the competition. Lead times on the order of two days have been used when the teams are known to be more sophisticated. The precise lead time needs to be carefully considered however. If the students are skilled and have too much time, they craft workarounds for most of the primary problems and prepare automated attack scripts that are ready to launch as soon as the face-to-face portion begins.

Experience has taught us that the students underestimate how much work it takes to actually configure their network, even in cases where they were given a significant lead time to develop it. This is in line with the experiences of Fanelli and O'Connor [16] who, in describing the USMA's preparation for CDX, wrote "Our students, for example, consistently underestimate how much time it will take them to implement their Blue Cell network, leading to a weekend-long 'death march' of final CDX preparation.... On the other hand, there may be a learning benefit from being on such a 'death march' at least once."

This type of exercise has been best when used in a full-day format, where the students have six to eight hours to attack and defend their network. Scoring in these events has also remained low-key and informal. When run with more sophisticated students, we have also included simple injects that are also scored. These have included:

- Setting up a private NTP server on Linux for the network.
- Setting up a log server like Splunk for the network.
- Adding a new web application like Joomla to the network.

V. Red Team

All three types of exercises have been designed to be run without the requiring the presence of a Red Team and each has been successfully run without a Red Team on multiple occasions.

However, we have also found that adding a Red Team greatly enhances the exercises. They expose students to sophisticated and complex attacks, and require that the students think that much harder about what they see on their network during an exercise.

At first, recruiting a Red Team for a small-scale student exercise can seem a daunting task. Where can a single faculty member find external experts willing to attack a group of students in such an exercise? At Towson, we began by asking both graduate students and members of the university's IT shop if they would be willing to participate, where we found willing volunteers. We also have approached members of the Red Team at competitions like the regional NCCDC to see if they would be also be willing to Red Team for us. They have not only been gracious enough to help us, but also often referred other, more local experts who would also help.

Interestingly, once Red Teams became common in the exercises run in our capstone class, graduating students would then ask if they could come back the following year and act as the Red Team for the class and for the other competitions that we would hold. After some years of this process, we have developed a substantial corps of volunteers, with a mix of graduates and friends of the program.

VI. Recruiting

We have found that these cyber security competitions are becoming quite popular among the students in our institution. We have taken an active role in promoting both the student team and the competitions. Our security laboratory is now festooned with banners from the various regional competitions to which Towson University has sent teams, including the mid-Atlantic Regional CCDC, the Cyb3r Battl3ground event run by CSC, and the Maryland Cyber Challenge and Conference (MDC3). Students of all skill levels are invited to participate in our competitions and to come to meetings of the Cyber Defense club. That group has ballooned in size, with meeting attendance peaking at more than forty students for a student demonstration of Metasploit. Interest in the becoming a member of the student team has now become so intense that we have had to resort to try-outs to determine which students will be eligible to compete in events like the NCCDC where team size is capped.

We are not the only institution to note an upswing in interest in cyber security as a result of competitions. Augustine and DeLooze [39] describe how the Naval Academy developed a "cyber-weekend" for new incoming students that involved both education and cyber-security competitions with both offensive and defensive components. Afterwards, they found that they nearly doubled the number of students who chose computer science or information technology as their major.

VII. Conclusions

Cyber security competitions do not need to be large events with dozens of organizers and hundreds of participants. It is possible for a single faculty member to develop, organize and run a small scale-competition for two, three or four teams of students with a reasonable investment of time and effort. This investment can pay dividends not only in improving student learning, but can also generate student interest and excitement as well as help recruiting.

VIII. References

[1] The National Collegiate Cyber Defense Competition. Web Site, February 2012: http://www.nationalccdc.org/

[2] The UCSB iCTF. Web Site, February 2012: http://ictf.cs.ucsb.edu/

[3] O'Leary, Mike. A laboratory based capstone course in computer security for undergraduates. In *Proc. of the 37th SIGCSE Technical Symposium on Computer Science Education*, Houston, TX, March 2006.

[4] White, Gregory B. and Williams, Dwayne. The Collegiate Cyber Defense Competition™. In *Proceedings of the 9th Colloquium for Information Systems Security Education Georgia Institute of Technology*, Atlanta, GA, June 2005.

[5] Conklin, Art. The Use of a Collegiate Cyber Defense Competition in Information Security Education. In *Proceedings of the Information Security Curriculum Development (InfoSecCD) Conference '05*, Kennesaw, GA, September 2005.

[6] White, Gregory B. and Dodge Jr., Ronald C. The National Collegiate Cyber Defense Competition. In *Proceedings of the 10th Colloquium for Information Systems Security Education,* Adelphi, MD, June 2006.

[7] Rosenberg, Tim and O'Brien, Casey. The Growth of the Mid-Atlantic CCDC: Public - Private Partnerships at Work. *Proceedings of the 12th Colloquium for Information Systems Security Education*, Dallas, TX, June 2008.

[8] Whitman, Michael E. and Mattord, Herbert J. The Southeast Collegiate Cyber Defense Competition. In *Proceedings of the Information Security Curriculum Development (InfoSecCD) Conference '08*, Kennesaw, GA, September 2008.

[9] Carlin, Anna; Manson, Daniel P. and Zhu, Jake. Developing the Cyber Defenders of Tomorrow with Regional Collegiate Cyber Defense Competitions (CCDC). Information Systems Education Journal, **8**(14), 2010.

[10] Bei, Yan; Kesterson, Robert; Gwinnup, Kyle and Taylor, Carol. Cyber defense competition: a tale of two teams. *Journal of Computing Sciences in College*. **27**(1), 2011, pp. 171-177.

[11] Sroufe, Paul; Tate, Steve; Dantu, Ram and Celikel, Ebru. Experiences During a Collegiate Cyber Defense Competition, *Journal of Applied Security Research*, **5**(3), 2010, pp. 382–396.

[12] Childers, Nicholas; Boe, Bryce; Cavallaro, Lorenzo; Cavedon, Ludovico; Cova, Marco; Egele, Manuel and Vigna, Giovanni. Organizing Large Scale Hacking Competitions. In *Detection of Intrusions and Malware Vulnerability Assessment,* Lecture Notes in Computer Science, 2010, Volume 6201/2010, pp. 132-152.

[13] Doupé, Adam; Egele, Manuel; Caillat, Benjamin ; Stringhini, Gianluca; Yakin, Gorkem; Zand, Ali; Cavedon, Ludovico and Vigna, Giovanni. Hit 'em where it hurts: a live security exercise on cyber situational awareness. In *Proceedings of the 27th Annual Computer*

Security Applications Conference (ACSAC '11). ACM, New York, NY, USA, 51-61

[14] Mullins, Barry E.; Lacey, Timothy H.; Mills, Robert F.; Trechter, Joseph E. and Bass, Samuel D. How the Cyber Defense Exercise Shaped an Information-Assurance Curriculum. *IEEE Security and Privacy*, **5**(5) (September 2007), pp. 40-49

[15] DeLooze, Lori L. Counter Hack: Creating a Context for a Cyber Forensics Course. In *Frontiers in Education Conference 2008* (FIE 2008), Saratoga Springs, NY, October 2008.

[16] Fanelli Robert L. and O'Connor, Terrence J. Experiences with practice-focused undergraduate security education. In *Proceedings of the 3rd international conference on Cyber security experimentation and test* (CSET'10). Washington, DC, August 2010.

[17] Conklin, Art. Cyber Defense Competitions and Information Security Education: An Active Learning Solution for a Capstone Course. In *Proceedings of the 39th Hawaii International Conference on System Sciences*, Kauai, HI, January 2006.

[18] Schafer, Joseph; Ragsdale, Daniel J.; Surdu, John R. and Carver, Curtis A. The IWAR range: a laboratory for undergraduate information assurance education Journal of Computing Sciences in Colleges, **16**(4), 2001, pp. 223-232.

[19] Hill, John M. D.; Carver Jr., Curtis A.; Humphries, Jeffrey W. and Pooch, Udo W. Using an isolated network laboratory to teach advanced networks and security. *SIGCSE Bull*. **33**(1), 2001, pp. 36-40.

[20] Micco, Mary and Rossman, Hart. Building a cyberwar lab: lessons learned: teaching cybersecurity principles to undergraduates. *SIGCSE Bull*. **34**(1), 2002, pp. 23-27.

[21] Caltagirone, Sergio; Ortman, Paul; Melton, Sean; Manz, David; King, Kyle and Oman, Paul. Design and Implementation of a Multi-use Attack-Defend Computer Security Lab. In *Proceedings of the 39th Hawaii International Conference on System Sciences*, Kauai, HI, January 2006.

[22] Bullers, Jr., William I.; Burd, Stephen and Seazzu. Alessandro F. Virtual machines - an idea whose time has returned: application to network, security, and database courses. *SIGCSE Bull.* **38**(1), 2006, pp. 102-106.

[23] Collins, Michael; Schweitzer, Dino; Massey, Dan. CANVAS: a Regional Assessment Exercise for Teaching Security Concepts. In *Proceedings of the 12th Colloquium for Information Systems Security Education*, Dallas, TX June 2008

[24] Werther, Joseph; Zhivich, Michael; Leek, Tim and Zeldovich, Nickolai. Experiences in cyber security education: the MIT Lincoln laboratory capture-the-flag exercise. In *Proceedings of the 4th conference on Cyber security experimentation and test* (CSET'11), San Francisco, CA, August 2011.

[25] Walden, James. A Real-Time Information Warfare Exercise on a Virtual Network. *SIGCSE Bull*. **37**(1), 2005, pp. 86-90.

[26] Aman, J. R.; Conway, James E. and Harr, Christopher. A Capstone Exercise for a Cybersecurity Course. *Journal of Computing Sciences in Colleges*, **25**(5), 2010, pp. 207-212.

[27] Louthan, George; Roberts, Warren; Butler, Matthew and Hale, John. The Blunderdome: an offensive exercise for building network, systems, and web security awareness. *In Proceedings of the 3rd international conference on Cyber security experimentation and test* (CSET'10). Washington, DC, August 2010.

[28] Lee, Christopher P.; Uluagac, A. Selcuk; Fairbanks, Kevin D.; and Copeland, John A. The Design of NetSecLab: A Small Competition-Based Network Security Lab. *IEEE Transactions on Education*, **54**(1), 2011, pp. 149-155.

[29] Catuogno, Luigi and De Santis, Alfredo. An internet role-game for the laboratory of network security course. In *Proceedings of the 13th annual conference on Innovation and technology in computer science education* (ITiCSE '08), Madrid, Spain, June 2008.

[30] Aman, James R. Black Hat/White Hat: an aggressive approach to the graduate computer security course *Journal of Computing Sciences in Colleges*, **22**(2), 2006, pp. 52-58.

[31] Endicott-Popovsky, Barbara and Lockwood, Diane L. A Social Engineering Project in a Computer Security Course. *Academy of Information and Management Sciences Journal*, **9**(1), 2006, pp.37-44.

[32] Dimkov, Trajce; van Cleeff, André; Pieters, Wolter and Hartel, Pieter. Two methodologies for physical penetration testing using social engineering. In *Proceedings of the 26th Annual Computer Security Applications Conference* (ACSAC '10), Austin, TX, December 2010.

[33] Dimkov, Trajce; Pieters, Wolter and Hartel, Pieter. Training Students to Steal: A Practical Assignment in

Computer Security Education. *In Proceedings of the 42nd ACM technical symposium on Computer science education* (SIGCSE '11). Dallas, TX, March 2011.

[34] The Zen Cart Web Site, February 2012: http://www.zen-cart.org

[35] The IPFire Web Site, February 2012:http://www.ipfire.org

[36] Metasploit Penetration Testing Software Web Site, February 2012: http://www.metasploit.com/

[37] BackTrack Linux – Penetration Testing Website, February 2012: http://www.backtrack-linux.org/

[38] OldApps Web Site, February 2012: http://www.oldapps.com/

[39] Augustine, Thomas A.; DeLooze, Lori L.; Monroe, Justin C.; and Wheeler, Christopher G. Cyber competitions as a computer science recruiting tool. *Journal of Computing Sciences in Colleges,* **26**(2), 2010, pp. 14-21.