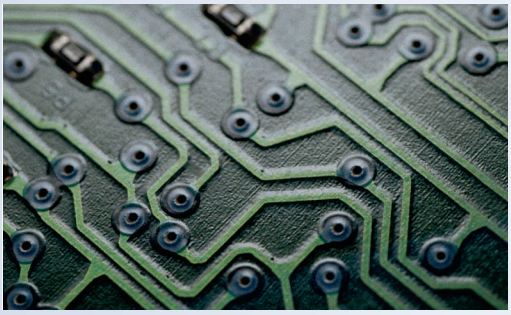
Training for Information Assurance

In 2001, cadets from three military institutions built networks and defended them against a week of attacks led by the National Security Agency. The trial-by-fire exercise, which repeats in 2002, is a step toward preparing future network designers and administrators to think more strategically.



Donald Welch
Daniel Ragsdale
Wayne Schepens
United States Military Academy at West Point

One of the newest threats to our national security comes not from a novel weapon or an unfamiliar adversary, but through the intangible bits and bytes of cyberspace. Information technology has become a two-sided coin: With the opportunity for tremendous advances comes the potential for crippling loss.[1] Thus, IT-literate societies, among which the US ranks highest, are at great risk. Although as of this writing no significant cyberattacks have followed the World Trade Center attacks of 11 September 2001, the FBI believes cyberspace terrorism is becoming a significant challenge to national security.[2]

US military institutions are already preparing for cyberspace terrorism and warfare by educating cadets in information assurance. In January 2000, the US Military Academy at West Point created an information assurance course that centers on a hands-on defensive project: Protect a real network from real attack.

USMA faculty teaching the course challenged academies with similar information assurance curricula to compete in defending an identical network against attacks by a Red Team. The US Air Force Academy and Naval Postgraduate School joined USMA to provide the three network-defense teams. The Red Team consisted of professionals from the US Army Land Information Warfare Activity, US Air Force Information Warfare Center, and National Security Agency.

Besides conducting the attacks, the Red Team evaluated the cadets' performance in defending their networks and recommended a winner. The winning team received a trophy from the NSA, which they were to keep for the academic year. (In the 2001 contest, NPS competed only for exhibition, since it would not be fair for undergraduates to compete with graduate students.)

The cyberdefense exercise (CDX) proved beneficial to all involved. The cadets gained insight into how a real adversary could attempt to compromise a network, and countering those attacks gave them valuable experience in thinking creatively under pressure. The educators identified opportunities to tailor classes and address curriculum gaps. A second exercise will take place in April 2002 with two new competitors: the US Coast Guard Academy and US Naval Academy.

One motivation for CDX is the growing need to view information assurance as more than a static concept—beyond assuming that the quality of the firewall determines the safety of the information behind it. As the "Defense Sun-Tzu Style" sidebar explains, warfare against an information infrastructure has some of the same tenets as warfare on a physical battlefield.

Our intent was to force the cadets to pull together what they know theoretically and apply it to a real network under attack *without* risking the damage that mistakes would cause on a live network. We believe CDX is a first step toward embracing this more holistic approach to learning, which ultimately leads to better equipped IT professionals.

0018-9162/02/$17.00 © 2002 IEEE

Defense Sun-Tzu Style

All the cadets who participated in the cyberdefense exercise (CDX) are comfortable with the notion of protecting and defending a geographical location. All USMA cadets, for example, enter the information assurance course with at least 3.5 years' military experience. During this time, they study military history and military science and participate in field exercises conducted during summer training. Consequently, the cadets who planned and executed the CDX drew heavily on time-tested tenets and principles of military planning and operations.

One of the most significant works cadets are exposed to is *Sun-Tzu Ping-Fa* (*Sun Tzu: The Art of War*). This classic, written more than 2,000 years ago by Sun Wu, Chinese general to the state of Wu, remains one of the most important treatises available on military strategy. Intended solely for the military elite of Wu's time, the treatise has been handed down for centuries, and is still used by both military commanders and business leaders of today.

Common themes in Wu's rich work are the need to know your opponent's tactics, techniques, and vulnerabilities, as well as to thoroughly understand your own weaknesses and how your opponent might exploit them:

> One who knows the enemy and knows himself will not be in danger in a hundred battles. One who does not know the enemy but knows himself will sometimes win, sometimes lose. One who does not know the enemy and does not know himself will be in danger in every battle.*

This notion was the foundation of our information assurance course. Simply knowing about defensive technologies and policies is not sufficient. To be successful, a defender in cyberspace must get inside a potential adversary's head.

Defining attacker characteristics

Cadets who participate in the course first learn about and then employ the same tools and techniques that attackers employ, but they can implement these offensive, counterattack techniques safely, in the isolated network at USMA. Cadets write malicious applets and use port scanners, network sniffers, and vulnerability scanners to find holes in a system's defenses. They use scripts, Trojan horses, and other tools to gain root-level access to target hosts. Thus, they become familiar with the capabilities of potential adversaries, and they better understand vulnerabilities in existing information systems and how an attacker might exploit them.

Once cadets understand some representative tools and techniques, they learn about the many kinds of counterassurance operatives. Since human behavior is incredibly complex, it helps to understand who the attackers are by generalizing their behavior according to motivation and skill level. To determine motivation, cadets probe an attacker's background—is the adversary part of a criminal syndicate or a teenager on a cyberspace joy ride? A criminal attacker is likely to have considerable resources and be wary of being discovered. A teenager who doesn't find easy success is likely to move on to other targets.

Skill level, another defining attacker characteristic, must be considered apart from motivation. Some skilled attackers are explorers and researchers with pure motives, and some professional cyberwarriors are only slightly better than an everyday teenage hacker. Does the attacker use only published exploits or can he discover new vulnerabilities and craft new exploits? Simply keeping up with current patches can thwart published exploits; it takes combinations of countermeasures and considerable effort to defend against unknown vulnerabilities.

Challenging the adversary

When the cadets know counterassurance tools and appreciate their potential adversaries' skills and motivations, we challenge them to think as their adversaries would. One method is to have the cadets create an attack tree, a formal way to represent the steps in a potential attack that will let an adversary accomplish a specific goal. Each attack tree is based on an analysis that considers the adversary's motivation, tools and techniques, available resources, and skill level.

The analysis normally reveals vulnerabilities that the cadets would miss had they looked at the network only from a defender's view. For example, in the 2001 CDX, cadets realized that their adversary could insert a system backdoor by transforming a system binary into a Trojan. Once they knew this, they implemented an integrity-checking program that raised an alarm when the system binary appeared to have been modified. This understanding enabled them to eliminate a potential avenue of attack.

Finally, cadets must be able to outthink their adversaries. During the CDX, cadets had to react quickly to an attacker action or the network would fail. Consequently, cadets came away with a much deeper appreciation for the many ways an adversary might be able to defeat, or at least subvert, their defenses. By the end of the course, cadets had gone from ignorance about potential adversaries to an understanding of how an adversary thinks and ways an adversary could exploit a network weakness. In Sun-Tzu terms, our cadets would not be in danger in a hundred cyberbattles.

System administrator's role

Another significant theme in Wu's work is the overriding importance for the generals of the time to fully understand the nature of warfare:

> A general who understands warfare is the guardian of people's lives, and the ruler of the nation's security.

Because of our increasing dependence on interconnected information systems, the administrators of these systems will have a far larger role in our national security—possibly greater than most civilians have had since the militia fought the War of 1812. As the 11 September attacks painfully illustrated, a person no longer needs an army to attack a nation. A major cyberspace attack could disable parts of our critical infrastructure and greatly affect our society. Civilian system and network administrators will bear the brunt of any such attack. We believe our information assurance course better equips these administrators to serve as guardians and protectors of our society's security.

*Translations are from http://www.sonshi.com/sun2.html.

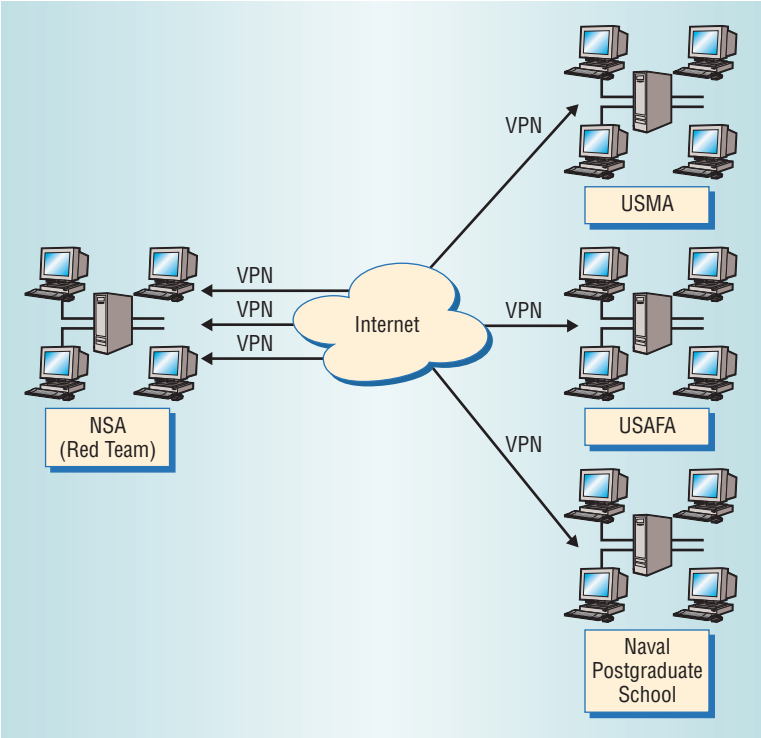


Figure 1. Elements of the cyberdefense exercise. The Red Team at the National Security Agency connected through virtual private networks (VPNs) to the cyberdefense network, which cadets designed and installed at each school.

THE BATTLEFIELD

Under our direction, cadets from USMA's information system design course planned a small network that each location could set up relatively easily.

The cadets had four major tasks:

- design a cyberdefense network, including various operating systems, network services, databases, and applications typical of current military and commercial information infrastructures;
- provide secure, remote connectivity to the cyberdefense network for the Red Team;
- ensure that the cyberdefense network is electronically separate from all the school's backbone networks; and
- provide installation instructions and CDs that contain the necessary network and application software so that other participating schools could copy the identical configuration.

We instructed the design students to make the network intentionally weak in security safeguards so that cadets in the contest could practice the network-defense skills taught as part of their respective information assurance courses. As Figure 1 shows, Internet-hosted virtual private networks (VPNs) connected the Red Team and the three participating schools. The VPNs provided a way for the Red Team to enter the battlefield without damaging the production network or causing false alarms for the production network's security monitors.

Figure 2 shows the cyberdefense network that we used in 2001, which consisted of platforms running Sun Solaris, Linux, Windows 2000, Windows 98, and Windows NT. The systems provided Web, database, file, and e-mail servers, as well as the normal contingent of network utilities. Exercise participants used Internet access to download the latest patches and software updates. Representatives from the participating schools and the Red Team approved the final design in a January 2001 summit.

THE ATTACKERS

The Red Team was a large part of CDX's success. As early as September 2000, the 92nd Aggressor Squadron at Kelly Air Force Base had learned about the cyberdefense exercise through one member's chance meeting with a USMA faculty member at an information assurance conference. Squadron members immediately expressed interest in becoming part of the Red Team and proposed an organization and mission role at the January 2001 summit. They also agreed to provide the evaluation criteria to determine the winning student team.

In January 2001, USMA briefed the NSA Executive Command on the exercise, and the agency offered not only to provide Red Team members but also to host the Red Team. Since USMA is an Army institution, it made sense to complete the team with members of the Army's Land Information Warfare Activity.

The Red Team gathered at NSA headquarters to plan and conduct the attack under NSA leadership. One of the team's tasks was to draft a final report on its ability to defend its networks and to recommend a winning cadet team to a board of representatives from each service academy. The board, which also included the group who designed the exercise and representatives from the Red Team itself, then certified the winner.

THE BATTLE

From the cadets' view, the goal of the exercise was to minimize the risk of a security breach while ensuring that necessary operational services remained functional. Each team comprised groups of three or four cadets, and each group had a subordinate task, which the supervising faculty assigned, as well as a coordinating task that spanned groups. At USMA, each group was responsible for a specific operating system, and the entire USMA team had a task that spanned operating systems. For example, one group was responsible for the Windows 2000 machines as well as for developing an attack tree[3] to assess the network's overall defensive strength.

The student teams at the three participating schools then developed defense plans but did not

actually alter the network. Once they had finalized these plans, the cadet teams had 10 days to configure and secure their respective networks in accordance with their plans. This meant that they had to implement the necessary changes without compromising the network's services. Once the 10 days were over, the team had to have a network that was ready, at least theoretically, for the week of Red Team attacks.

Rules of engagement

During the week of attacks, each team had to transmit the order of the day to all workstations within the cyberdefense network while maintaining the message's confidentiality and integrity. This order included a complete report of all hostile activity that the team had detected to that point. As such, the order represented the "flag" that the Red Team was trying to capture. The cadets were trying to protect all aspects of the network, but the supreme object of protection was the order, which was exposed when the cadets transmitted and stored it on the network machines.

To ensure that the exercise ran as smoothly as possible, we imposed several controls. The cadets had to adhere to their normal schedule as much as possible, which meant that during the day, they had to attend their classes and other activities such as intramurals, drills, and ceremonies. The Red Team conducted its attacks during the morning and early afternoon. The cadets would check the network as they had time during the day. Only in the evening, after they had completed their required duties, would the cadets meet to determine what had happened to the network and plan their response. They also had to be in their rooms for taps at 11:00 p.m. This allowed them approximately five hours each night to deal with the damage the Red Team had caused.

Cadets could react to attacks only by reconfiguring the network to make it more secure. We did not allow any preemptive operations against the Red Team or other exercise participants. We also ruled out social engineering—using psychological tricks on legitimate users to get system access information (usernames and passwords, for example). Although we realize that social engineering is a major threat to information assurance, allowing it would have made the exercise more complex without much additional return.

Building defenses

Most of the cadets' initial abstract plans for network defense were immature. Even though we had instructed the cadets to develop attack trees, the resulting plans were neither broad nor deep. The second-iteration plans were much better, but, if fully implemented, they still would not form a completely coherent defense. Clearly, theoretical discussions of network defense did not lead to an in-depth understanding of an actual defense.

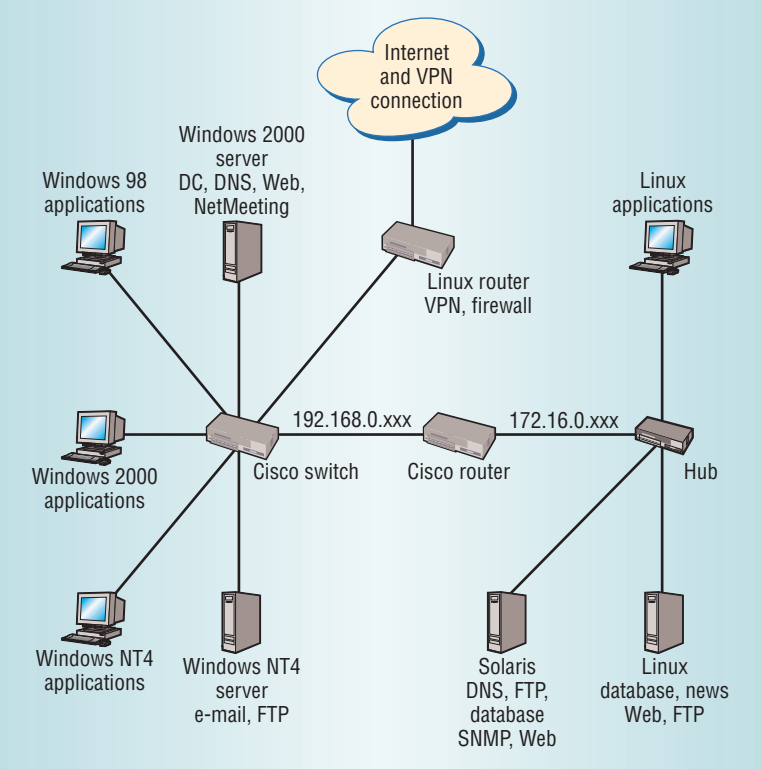


Figure 2. The cyberdefense network. This simple network balances the need for broad representation of different operating systems, networking equipment, and services against size and simplicity. DC: domain controller; DNS: domain name system; FTP: file-transfer protocol; SNMP: simple network-management protocol; VPN: virtual private network.

Only when the teams began reconfiguration did they finally grasp what it means to secure a network. This pattern is similar to what we've observed in teaching design as part of software engineering—students generally understand what is required in a complete, consistent design only when they begin to implement it.

All the teams found it challenging just to keep services running as they changed the network to secure it. In fact, the USMA team was a day late in even having an operational network for the Red Team to attack. Everyone learned a key lesson: It's not easy to administer a network in the first place, and increasing security makes administration even more difficult.

We had talked in class, for example, about turning off unused services—a simple concept in the abstract—but cadets found that turning off some "unused" services can actually break a required network service. To make matters worse, having fewer running services makes it harder to diagnose and fix problems. The cadets learned the importance of addressing network security from a systemwide perspective and of basing their security decisions on operational need versus associated risk instead of just taking measures to make an individual computer more secure.

The team learned firsthand how much work it takes to make default installations more secure.

Another lesson was one they had already learned in software development but had to relearn in systems administration: It is easier to diagnose and repair a problem when you've made only a small system change than when you've made large changes without testing. One group dutifully followed a well-known checklist for securing a server only to find that, when they had finished, much of the server no longer worked. After hours of trying to determine if the fault lay in their implementation or the incompatibility of the additions with the original network, they gave up and rebuilt the system.

The battle begins

Because of restrictions we placed on the exercise, the Red Team could not use many of the tools typically open to an attacker. For example, an attacker might spend weeks or months probing and scanning the target network to understand it and to wait for network defenders to make more mistakes and show more weaknesses. The Red Team had to run attacks in one week. Also, the student teams were highly attuned to security, so they were not as vulnerable to being tricked into helping the attack.

Nonetheless—to the USMA team's dismay—the Red Team made several serious penetrations and effectively owned the network within three hours. That evening, over the four hours the cadets had available, they removed vulnerability after vulnerability, discovering the security measures they should have taken during the initial configuration.

The team already knew that it had to make default installations more secure, but they quickly learned firsthand how much work it takes to accomplish this. The Red Team's early successes were all against default services that were running but not required for the exercise. The Solaris 7.0 server was vulnerable to two exploits against the Solaris AnswerBook2 service (a help service), for example. A lack of authentication checks within the administration interface of AnswerBook2 let the Red Team create a remote user account. Once the Red Team had user-level access, it was able to get an unauthorized user account with elevated privileges. With that account, the team could execute commands with root permissions.

Once the Red Team had access to the Solaris server, it used a combination of a standard root kit and a homegrown backdoor to keep access. From the Solaris machine, the team fanned out and attacked the rest of the network, exploiting the normal trust relationship in the network through the .rhosts and .equiv files. Although Sun offers a patch to preclude execution of either of these exploits, the cadets had neglected to install it.

The Red Team planted Trojans wherever it could. The USMA team left an open share on the Windows 98 machine, which became a Trojan magnet. The attackers were then able to remotely connect and access system files. Because Windows 98 has few network-access controls, keeping the machine functional made the entire network vulnerable.

One Web server—Microsoft Internet Information Services (IIS)—proved vulnerable to a Unicode exploit and provided a launching point for further attacks. The USMA team detected these exploits and countered by disabling file sharing on the system and installing a Microsoft patch for IIS.

The Linux workstation was vulnerable to a remote buffer overflow against the printer software, thus allowing an unauthorized user to remotely execute commands with root permissions. The cadets discovered this vulnerability and patched it before the Red Team could take advantage of it.

The Cisco switch had a read/write-enabled public SNMP string, which let the Red Team download the switch's configuration. The Red Team then cracked the password and completely compromised the network. With the Red Team owning the switch, none of the computers in the network could reliably communicate with each other. After discovering the switch's vulnerability, the team reconfigured it to regain control of the network.

Finally, the Windows 2000 server allowed an unauthorized user to connect remotely with local administrator privileges through the default installation. The cadets had installed a flawed security patch during the network preparation and had to rebuild the server. While rebuilding the server under pressure, they neglected to make all the needed configuration changes, which left this server vulnerable. This is a common pitfall in all types of network administration: Fixing one problem often creates another.

Feedback

After the exercise, the Red Team gave the cadets feedback in a teleconference. All the teams were highly motivated to know exactly what happened, and as exercise designers, we found it helpful to identify what the cadets didn't understand.

A good example is the administrator password on the Solaris server. One USMA team member noticed a compromise and, to be safe, changed the password, leaving a note about the new password for the next shift. When the new shift arrived, how-

ever, they could not get access using the new password. After much frustration, they found the old password was still working. This happened repeatedly, and the other team members were becoming angry with the bewildered cadet, who kept claiming that he had changed the password. The team finally resolved the problem by completely reinstalling the operating system.

During the feedback session, the Red Team revealed that they had installed backdoor accounts and made a copy of the password file. To preclude the loss of that backdoor, they automatically copied the old file over the live password file every 30 minutes, thus overwriting the changes. This incident left an indelible impression. Cadets now know that if something seems wrong, it probably *is* wrong and requires further investigation. We know of no way to duplicate this depth of understanding without firsthand experience and meaningful feedback.

War stories

As might be expected in an exercise of this scale, we had a rocky execution. All the cadets were computer science majors and so understood much of the theory behind network operation. Unfortunately, they had little experience in actually administering a network. There were problems, but we viewed each one as an educational opportunity.

Most cadets thought that the main challenge would be securing the network, not keeping it functional. To their surprise, they found that changing a configuration setting, installing a patch, or implementing some other security measure often had consequences far beyond what they had expected. This gave us the opportunity to focus on topics the cadets had learned in other courses. In trying to debug a failed operating system patch, the cadets realized that they had to understand what was happening inside the operating system—the way software modules fit together—as well as what was happening over the network.

The most important outcome of the exercise was that the cadets eventually did figure out how to defend the network. The Red Team attacks gave them real-time feedback that was far more meaningful than any professor's red pen.

When the exercise was over, each group of cadets within each team put together a final briefing about what they did and didn't do correctly during the exercise. The goal was for the groups to think about what happened and how their actions, as well as those of the Red Team, led to the outcomes. This activity was essential to producing a coherent view of the exercise, and we cannot overemphasize its importance: The cadets learned a great deal during their reflection, and it helped everyone to put those ideas in a structured form.

Each group listed what, if anything, they were likely to remember about the exercise in five years. The overwhelming knee-jerk reaction was, "Why didn't you train us how to defend a network?" We explained that one course cannot do all the possible defense scenarios justice and that summarizing the material or taking other shortcuts diminishes the chance that students will absorb the information in a meaningful way. Rather, our goal was to ensure that the cadets experienced the defense of a real network and learned from their mistakes. In the long term, this better prepares them to defend a network against a variety of attackers.

Even if these cadets never install another patch, they will understand the struggle that administrators go through to keep a network both functional and secure.

Many cadets noted "how difficult it is to administer a network, and how little a computer science education prepared us to do so." What they didn't seem to understand was that they did a credible job of learning to run the network even without specific training. More to the point, the Computer Science Accreditation Board does not emphasize network administration, but rather software creation. Thus, these cadets learned the same lessons that a system administrator would also learn by experience.

One group said that they would remember "how insecure default systems installations are." If this is the only point that sinks in, the networks these cadets will eventually administer should be better off. Another group added that "never assume anything" would stay with them for years. Having a healthy mistrust and retesting assumptions are good leadership strategies for a network administrator.

Even if these cadets never install another patch, they will understand the struggle that administrators go through to keep a network both functional and secure. They will also understand how overburdened administrators can and will make mistakes that leave networks vulnerable.

Another group said they would remember "the importance of teamwork in defending a network." A good leader will recognize that this lesson applies in cyberwar as well as conventional battle.

The final group said they would remember that "you can always make the network more secure." What more can we ask from the leaders responsible for running our information infrastructure? We want them to be thinking constantly about potential threats and how to counter them.

All these comments indicate a depth of understanding in information assurance that is rarely

The cadets gained a deep understanding of how to apply the firewall and how technologies and policies interact.

encountered among the leaders and managers currently responsible for our nation's critical information infrastructure.

What the cadets did not say was also encouraging. They were engrossed in the minutiae of network configuration and security for weeks. They discovered that one of the most effective tools in the arsenal was the host-based firewall, but they also understood that the key to defensive success is not the firewall per se. Rather, it is a deep understanding of how to apply the firewall and how technologies and policies interact. This thinking beyond how to use a specific package is another indicator of the value of this exercise.

Finally, even those on the Red Team—professional security auditors—had interesting impressions. Although some had been tasked to conduct penetration exercises before, the exercises typically had a limited scope and took place over extended periods. In the cyberdefense exercise, they had greater freedom in choosing an attack, and the entire attack took only one week. The compressed schedule gave them a different perspective of a network attack. According to the Red Team leader, an NSA representative, "We probably learned about as much as [the cadets] did."

And the victor is...

USMA became the first owner of the NSA Information Assurance Director's Trophy. The decision was close, but the trophy was awarded to the USMA team because of its ability to keep the required services running as well as its prevention and detection of penetrations. In reality, the contest had no losers. This may sound trite, but all participants learned so much that the exercise became a win-win competition.

DESIGNING A TRAINING EXERCISE

The cyberdefense exercise took a great deal of time and effort on the part of many organizations, but the educational benefits were off the scale. We saw impressive and steady growth in the cadets throughout the exercise, and by the end of the week, the USMA network was essentially secure from attack.

All the Red Team organizations gave generously of their personnel, equipment, and financial resources—which makes this exercise unique in some ways. Although the resources to duplicate it may be out of reach for many programs, others can still benefit from smaller exercises. We recommend basing any similar exercise on five key principles:

Make it a culminating experience. The exercise took place near the end of each cadet's undergraduate study of computer science. Because information assurance pulls together many fields from within and outside computer science, it is a natural medium through which students can draw on their entire education and apply it in a practical way.

Let the students own it. The cadets were exposed to situations they had never seen or even expected to see. They had to understand what was happening, determine how to fix it, and then actually implement the corrective action. They discovered the hard way how to plan and coordinate reactions to attacks. Their ability to do this of their own accord later in the exercise proves that they received an education beyond the normal classroom experience.

In an exercise of this magnitude, the students must have the responsibility for running and securing the network; they must also be able to control most aspects of the exercise. Giving them this kind of control is risky, but without a sense of ownership, the students will tend to quit when faced with a thorny problem. Keep instructors in a facilitating role only.

Pick a meaningful adversary. Matching wits with professional minds is a great incentive for students to perform well. It is one thing to be told why something you did was wrong. It is quite another to discover that you've been outwitted and have to clean up the resulting mess. Tremendous learning goes on when students have to scramble to determine what is happening in the network they are responsible for.

Incorporate some form of competition. Competition is an irreplaceable motivator. An exercise like this is difficult and frustrating. Students need something besides a grade to motivate them to spend a spring weekend in a hot, stuffy laboratory. CDX involved two kinds of competition. In addition to the obvious competition between the service academies for the best defense, there was also an adversarial competition between the Red Team and the cadets. This competition played on the well-established athletic rivalry between the service academies, but lower-key competition can still be effective. Even an exercise between two teams from the same school can be a sufficient motivator.

Have a network sandbox. The exercise must place restrictions on what students can and cannot do, and these restrictions must be enforced. Otherwise, collateral damage to a live network or the Internet will cause real problems for everyone. Having a network sandbox—an isolated area where the

cadets and Red Team could operate safely—was critical to maintaining student ownership. The NSA helped us establish identical, first-class networks, but an effective exercise does not require having top-of-the-line equipment. Most organizations can make a capable network with very little funding. We built our first isolated laboratory using leftover and obsolete equipment.

Upon graduation, all USMA cadets become officers in the US Army, and many will be responsible for the security of critical Army information systems. With a firm foundation in the fundamentals of information assurance, these officers will have the intellectual skills needed for the continued self-education that is vital in evolving technical disciplines.

With two more schools participating and with our lessons learned from running the 2001 exercise, we expect the 2002 cyberdefense exercise to be even better. Of course, in an exercise this complex, it is impossible to anticipate all problems; the key is to react quickly in dealing with the problems that do arise.

We encourage other organizations to follow our lead. Information assurance is a topic that is becoming integral to US national security. We can no longer rely solely on our armed forces to defend the nation. Professionals in the commercial and government sectors must do their part to defend our critical information infrastructures from cyberattacks. A competitive exercise such as the one we have described is a first step in strengthening those additional defenses. ■

References

1. "Critical Foundations: Protecting America's Infrastructures," Government Printing Office, Washington, D.C., Oct. 1997; http://www.info-sec.com/pccip/pccip2/report_index.html.
2. M. Vatis et al., "Cyber Attacks During the War on Terrorism: A Predictive Analysis," tech. report, Institute for Security Technology Studies, Dartmouth College, New Hampshire, Sept. 2001.
3. B. Schneier, "Attack Trees: Modeling Security Threats," *Dr. Dobb's J.*, Dec. 1999, pp. 21-29; http://www.ddj.com/print/documentID=12345.

Donald Welch *is an associate dean at the United States Military Academy at West Point. His research interests include information assurance education, information warfare, and simulation of information security. He received a PhD in computer science from the University of Maryland, College Park, and is a senior member of the IEEE. Contact him at welch@usma.edu.*

Daniel Ragsdale *is an assistant professor of computer science at the United States Military Academy at West Point and director of USMA's Information Technology and Operations Center. His research interests include information assurance education and intrusion detection. He received a PhD in computer science from Texas A&M University and is a member of the IEEE. Contact him at Daniel-Ragsdale@usma.edu.*

Wayne Schepens *is the NSA Visiting Fellow at the United States Military Academy at West Point. His research interests include public-key infrastructures, secure software construction, and information assurance education. He received an MS in civil engineering from Virginia Polytechnic Institute and State University. Contact him at Wayne-Schepens@usma.edu.*