CSET '09: 2nd Workshop on Cyber Security Experimentation and Test

Montreal, Canada
August 10, 2009

Sessions below summarized by Arun Viswanathan (aviswana@usc.edu)

OPENING REMARKS

Douglas Maughan, Program Manager in the Cyber Security R&D center from DHS, opened the conference on behalf of General Chair Terry Benzel from USC/ISI by giving a very brief talk on the importance of testbeds and security experimentation.

He was followed by Jelena Mirkovic from USC/ISI and Angelos Stavrou from George Mason University, who welcomed the attendees and thanked the Program Committee members. Jelena presented the statistics for CSET '09. There were 27 papers submitted for the conference, of which nine were accepted. Three papers were off-topic and were rejected. Of the 22 finally reviewed, 13 were on experimentation (four were accepted), five on testbeds (three accepted) and four on education (two accepted). She noted that the common problems found in rejected papers were lack of novelty, bad timing, and missing lessons learned.

On the future of CSET, she was enthusiastic that total submissions were up this year, with 25/27 papers coming from people unrelated to the DETER testbed. She, along with Angelos, commented on the lack of awareness among researchers about existing testbeds such as DETER/GENI. This situation will, hopefully, improve with newer and larger testbeds like GENI and NCR. Jelena noted a need for more submissions in the areas of education, tools, experiment methodology, and result validation. She concluded by stating that she was hopeful about having CSET '10 co-located with USENIX Security '10.

KEYNOTE ADDRESS

- ***The Future of Cyber Security Experimentation and Test***
  Michael VanPutte, DARPA Program Manager for US NCR

Michael VanPutte started his keynote by giving a short tour of the DARPA mission and key accomplishments from 1960 to date (from contributions during the space era, through their key role in building the Internet, to today's latest in warfare). He then classified today's cybertesting communities into two groups: operational and R&D. The cyber-operational community's mission is operational testing and training, whereas the mission for the R&D community is to experiment with new ideas. The operational community deals with inflexible, expensive, special-purpose testbeds, does manual configuration and management, has rigid test schedules, deals with constraining bureaucratic policies, and is largely driven by operationally focused policies. This leads to unrealistic testing, questionable results, and slow

research-to-operation transition, and it rarely produces production tools. The R&D community, on the other hand, deals with advancing current understanding, generating and testing newer ideas, and managing flexible but potentially unstable systems.

VanPutte talked about the importance of measurement in science in general and cyber research in particular, which is the key reason for the NCR being part of the President's Comprehensive National Cybersecurity Initiative (CNCI) program. The main goal of the NCR is to "provide a realistic and quantifiable assessment of US Cyber research and development technologies to enable a revolution in national Cyber capabilities and accelerate transition of these technologies in support of the CNCI." The NCR will be the measurement capability for cyber research for both civilian and military sectors.

VanPutte then laid out NCR's key challenges: security—securely running multiple tests at multiple security levels; range configuration and management—securely and safely allocating thousands of heterogeneous resources; test configuration and management—using GUIs for configuring and running tests; usability—building recipes for testing, having malware repositories to assist experiments, and having attackers and defenders provided as a service; realism—having 10K nodes along with chip-level heterogeneous VMs; test time—accelerating test time to reduce time to result; scientific measurement—doing forensic data collection, analysis, and presentation of results; and traffic generation—simulating traffic conditions with human behavior.

VanPutte described the program timeline for NCR. The design phase is over and the program is starting the prototype phase. Selected proposals will have 18 months to build a prototype, after which the program will enter the full-scale construction phase. Finally, in closing, VanPutte provided two ways in which everyone could participate in the effort: through government working groups, such as the Security Accreditation Working Group and Joint Working Group, and via upcoming conferences on security metrics, the science of cyber testing, and CONOPS development.

Andy Thompson from JHU asked about the possibility of open sourcing NCR. VanPutte said that it is a possibility but will strongly depend on the transition partner. Roy Maxion from CMU commented that he liked VanPutte's presentation because it clearly compared how things are with how they should be. Jelena Mirkovic from USC/ISI asked if the NCR will develop a workforce for attack technologies. VanPutte responded that the NCR may be used to evaluate the security of systems but will not create attack technologies. Jelena made a comment that the public knowledge base of NCR should have the ability to take inputs from the knowledge bases of already established testbeds. Angelos Stavrou asked how we could achieve diversity in hardware in the testbed. VanPutte acknowledged that it was a hard question but said that people have been experimenting with segmenting testbeds to achieve hardware diversity. Minaxi Gupta from Indiana University commented on the importance of real data sets to understand attacker behavior. She asked about efforts to make available real-time data sets. VanPutte said that real data from real attacks may include operational data and thus are difficult to unclassify. He said he would still need to look into the specifics of this. Ken Zatyko from BBN asked about the usage of NCR and the kinds of tests that would be run on it. VanPutte responded that NCR is primarily meant for large-scale tests for now but it would heavily depend on the transition partner. Steve Schwab from Sparta asked, "How big is big enough" for a testbed? VanPutte said they need to do the math to determine the statistically significant size for specific experiments.

SECURITY EDUCATION

- ***A Highly Immersive Approach to Teaching Reverse Engineering***
  Golden G. Richard III, University of New Orleans

Golden Richard presented his experiences with developing a hands-on reverse engineering course at the University of New Orleans. He described the course focus as being on reverse engineering malware, with an emphasis on understanding the theory of reversing.

An education in reverse engineering is absent from academia because a course in RE could be really hard on instructors, there is a perception that a semester is not enough to teach RE, the university might object to it, and, finally, there is a perception of limited student interest, which turned out to be quite untrue. To overcome these issues, Richard stressed the importance of building trust with the university and the students. He did not have any problems with the university and he laid down the law for student conduct and informed them of the impact of being involved in malicious activities. Students were thus careful and self-policing. Richard's reasons behind teaching reverse engineering were to train students for deep systems research and teach proper ASM/OS skills, apart from the fact that students were begging for such a course and he himself wanted to do it.

His audience for the course, taught for the first time in spring 2009, consisted of 25 students (2/3 graduate and 1/3 undergraduate). About 1/5th of the students had some OS internals knowledge and very few had any serious ASM skills, which proved challenging. The topics covered included the basic importance of RE, ethical and legal issues, techniques/tools used for RE, basic malware background, Intel assembler introduction, Windows PE formats, C basics, common malware functionality (e.g., delta offset calculation, API address discovery), and ended with anti-debugging/anti-VM technology. The lab setup for the course consisted of an isolated gigabit network, with workstations running Linux with Windows XP VMware images running as guests. The XP image consisted of popular tools like OllyDbg, IDA Pro, Sysinternals Suite, HBGary Responder,

VC++ , MASM32 SDK, and some industry-grade forensic tools.

Richard's approach to teaching the class was to immerse students in reversing malware samples immediately. Students started with very simple malware like Michelangelo, which required very basic skills, and progressed on to more difficult samples like Harulf and Conficker. Lectures were first given using PowerPoint, then students were given reversing assignments (performed in teams), followed by use of a document camera for assembly code walk-through, followed by lab sessions, and, finally, students producing documented ASM code for the assignments. The exams were based on assignments and were mostly focused on converting malware code to documented assembly. In conclusion, Richard described the course experience as fun, with great student interest and positive feedback. His course will now be offered on a regular basis at UNO.

Someone asked about the schedule of the course. Richard said that it was a 15-week course with two sessions of 80 minutes each per week. Angelos Stavrou asked about the kind of support students were provided in lab. Richard said that there was no real support team in the lab other than the professor. Stephen Schwab asked if the course is teachable using only open source tools, to which Richard said it was possible but that IDA Pro is well worth the cost of licenses. What key challenge did the students face in the course? Lack of assembly skills. Doug Maughan of DHS offered to make HBGary available for the course. What did the students think RE was about when they first went in? Richard answered, "Cool hacker street cred." Someone asked about the audience for the course. Richard said it was a mix of undergrads and grads; he found the undergrads to be more dedicated, whereas grads varied. Could the course be offered online? It would be very difficult, especially because it heavily relied on the document camera.

- ***Collective Views of the NSA/CSS Cyber Defense Exercise on Curricula and Learning Objectives***
  William J. Adams, United States Military Academy; Efstratios Gavas, United States Merchant Marine Academy; Tim Lacey, Air Force Institute of Technology; Sylvain P. Leblanc, Royal Military College of Canada

Efstratios Gavas described their experiences with NSA/CSS Cyber Defense Exercises (CDX) and its effectiveness in teaching information assurance. Gavas started out with an overview of CDX, which is in its ninth year of competition. It is a four-day exercise but typically requires months of preparation. CDX involves a red team vs. blue team competition, with a white team monitoring. Eight teams participated in the exercise (AFIT, NPS, RMC, USAFA, USCGA, USMMA, USMA, USNA), with RMC from Canada participating for the first time. Each team was given a network and a mock budget to secure a poorly configured network. The network is supposed to be fully functional and provide services like email, IM, a Web server, etc., in the presence of live attacks from the NSA red team. The teams were also supposed to deal with exercise "injects" such as forensics, help-desk requests, DNS, and network reconfigs, which are purposely introduced to simulate real-world administrative chores.

Gavas first gave an overview of the USMMA and its preparations for CDX. USMMA has no formal computer science or information assurance program for participating in the CDX. The USMMA also had only five students participating in CDX this year. As preparation for CDX, the team used a number of virus scanners to detect malware in their systems, used a bunch of network and process monitoring tools to detect suspicious activity, rebuilt their Web servers, and used graphical management tools (monowall and eBox) to simplify administration for their network. The team's results for the exercise were mixed.

Next, Gavas shifted to the results of other academies and their experiences with the exercise. He pointed out that differences between participating academies arise because of the different curriculum and learning objectives. USMA participates with a large team of 30–60 students. They have a very security-active CS department with an ACM chapter and a senior-level capstone elective titled "Information Assurance," which form a basis for USMA participation in CDX. As for the CDX experience, USMA cleaned workstations with a homemade Tripwire-like script and rebuilt the DB and Web server without seeing any significant compromises. As for AFIT, Gavas mentioned that they have a very good graduate program, with courses and labs specifically built for CDX training. Their participation was with two teams of 15. For the CDX, AFIT used IPSec effectively, utilized proxy servers, and mitigated compromises with least-user privileges. RMC from Canada participated for the first time in the competition. Details were not provided about their experience in CDX.

Gavas concluded his talk by giving details of the attacks used by the red team. There were 21 significant distinct compromises made; the most effective attack for the red team was malware callbacks, and the most interesting exploit was the OpenFire remote access exploit, which became public only a few days before the exercise. There was no time left for questions.

SECURITY EXPERIMENTATION

- ***Evaluating Security Products with Clinical Trials***
  Anil Somayaji and Yiru Li, Carleton University; Hajime Inoue, ATC-NY; José M. Fernandez, École Polytechnique Montréal; Richard Ford, Florida Institute of Technology

Anil Somayaji presented an alternative method to evaluate security solutions using Security Clinical Trials, which sparked a very lively and interactive session with lots of discussion. Somayaji made two observations: that regular users face a huge challenge in evaluating security products and standard lab-based practices used for evaluating and comparing security products prove very ineffective for users

in reality. Standard practices do not account for a lot of real-world variables such as interaction of the product with different software, users, systems, uses, and attack profiles, and thus cannot measure the actual security provided by the product. He proposed the idea of learning from the field of medicine, where they use clinical trials to overcome similar challenges of genetic diversity, environmental diversity, individual history, etc., in identifying effective remedies. Applied to the field of security, the idea proposed is to evaluate security products "in the field" with real users. Questions answered will be of the nature, "Does it work?" rather than "Why?" or "How?"

Their approach will be to isolate variables of interest via sampling and randomization and then measure indicators and outcomes. Somayaji presented a simple example for anti-malware software evaluation, where 1000 customers of a major home ISP are randomly selected. They are given incentives like free tech support and automatic off-site backups to encourage them to participate. Users are then assigned one of three major antivirus programs. A variety of measures are then used to monitor users and computers involved in the study over a period of three years, to learn the effectiveness of the antivirus solutions. Somayaji then discussed objections to this approach: the significant differences between biology and computers, the utility of such an approach, and the expenses involved. Although there were lots of issues with this approach, Somayaji said in conclusion, clinical trials are one way to determine the effectiveness of solutions in practice and complement lab-testing approaches.

Ken Zatyko from BBN asked why they chose to make a comparison with medicine and not with the criminal system. Somayaji said that the medical perspective was for looking at which defenses are the best. Steve Schwab asked about the legal issues arising out of comparing different organizations. Somayaji acknowledged that there was no way this could be done without the support of the organizations being tested, and he talked of some already willing to do this. John McHugh from Dalhousie University pointed out that medical companies participate in trials because of legal requirements, but antivirus vendors may not have any incentive to participate. Somayaji said this was a public policy question. There was also discussion on self-selection biases negating such trials. Somayaji pointed out that they are incentivizing random users to join the study by providing free backups, technical support, etc., to take care of self-selection biases. Angelos Stavrou asked why the ISPs could not do this themselves by monitoring user traffic, their product updates, and incidents. Somayaji said that this would potentially create a large biased sample and thus was a question of experiment design. Someone asked how they measure the outcomes. Somayaji responded that for now their method is to do retrospective analysis on automated low-level backups. Ray Maxion from CMU concluded the Q&A by interjecting that the "audience is inflicting death by a thousand cuts." His point was that they make such stuff work at CMU all the time and hence this should not just be dismissed. His last point to Somayaji was that as they are proposing a methodology, they must compare it with other methodologies.

- ***The Heisenberg Measuring Uncertainty in Lightweight Virtualization Testbeds***
  Quan Jia, Zhaohui Wang, and Angelos Stavrou, George Mason University

Zhaohui Wang started with an overview of the Heisenberg uncertainty principle followed by a brief discussion of the advantages of lightweight virtualization: process-level isolation, no interprocess communication, high efficiency, no requirement for any I/O or device driver virtualization, and only one copy of the OS image required. This work addresses the question of determining the maximum number of OpenVZ containers that could be run on a server.

The testbed architecture consisted of a Dell PowerEdge 1950 server equipped with two QuadCore Intel Xeon 2.66GHz processors, 8GB RAM, and Gigabit Ethernet. The software used was OpenVZ on a vanilla Linux kernel 2.6.24, along with the UnionFS stackable file system to reduce the memory requirements of the system. Each OpenVZ container ran only five processes: init, syslogd, dbus, sshd, and wget. The measurement approach used was to statically determine the shared and non-shared memory pages for each container and then evaluate the runtime CPU and memory consumption of the Virtual Execution Environments (VEEs) by monitoring /proc file system from the host. The experiments consisted of running containers in groups of 100, 200, 400, 600, 800, 1000, 1200, and 1400 containers, with each container running a wget process that would continuously fetch a pages from an Apache server in random intervals varying from 1 to 10 seconds. The monitoring process was run with varying sampling intervals of 0.1, 0.01, 0.005, and 0.001 seconds.

The results from the experiments showed that the completion times for the experiment increased as the number of containers was increased, but there was a profound increase when the frequency of measurements was increased. The conclusion drawn was that the more you measure, the more you lose. Zhaohui claimed that their work unveiled for the first time the uncertainty problem due to system resource contention in a lightweight virtualization environment. He pointed out that it was not a trivial task to determine the maximum number of VEEs that can be run on a physical host, due to this form of Heisenbergian uncertainty.

Roy Maxion from CMU asked why the CPU utilization maxed out at 600 VEEs for 0.1sec frequency in Figure 4? Zhaohui answered that the contention between the containers caused them to reach a threshold. As for the graphs at other frequencies, they were already affected because of over-measuring.

TESTBEDS

- ***The Virtual Power System Testbed and Inter-Testbed Integration***
  David C. Bergman, Dong Jin, David M. Nicol, and Tim Yardley, University of Illinois at Urbana-Champaign

Tim Yardley from UIUC presented their work on the Virtual Power System Testbed (VPST), which is a part of the larger Trustworthy Cyber Infrastructure for Power Grid (TCIP) project. TCIP works on securing devices, communication, and data systems that make up the power grid. VPST is designed to support exploration of security technologies being developed for large-scale power grid infrastructure. VPST at the core consists of RINSE, which is a network analyzer and simulator. RINSE is capable of performing high-performance, high-capability network analysis along with multi-resolution modeling of traffic and topology. VPST itself can be connected via secure links to external testbeds and utility power stations.

Yardley mentioned that SCADA systems prompted the work on VPST. SCADA research has a high barrier for entry and thus emulation of these systems can alleviate part of this concern by using accurate models. He mentioned that VPST is designed to leverage valuable resources from other testbeds such as DETER. Yardley then described the interconnection requirements of VPST. Secure interconnection between testbeds and between VPST and utility companies is a prime requirement. He mentioned use of Open PCS Security Architecture for Interoperable Design (OPSAID) in their architecture. Next, performance is a key requirement, as it is very important to keep latency low across multiple testbeds. VPST implements look ahead to keep simulation as close to real time as possible. Resource allocation is the next key aspect, and VPST tries to use a decentralized approach where interfaces to other testbeds are decomposed into modules for ease of customization. Reproducibility is important in SCADA systems because the dynamics of real SCADA networks cover a wide range of conditions, such as size of network, type of underlying physical medium, available bandwidth, and time-varying traffic patterns. Reproducibility is complicated due to human interactions with the system. The system must be able to record interactions and replay. Fidelity is the last of the key requirements, which means that VPST must be as transparent as possible to real devices. This also means that access is needed to real-time data patterns from utility companies.

Yardley next described the use cases for VPST. The first use case is in the training and human-in-the-loop event analysis. VPST allows captured system state to be replayed on the testbed, which can help in making better control decisions and rectifying decisions which may have led to failures in real situations. The second use case is for analysis of incremental deployment. As SCADA networks are large and complex, introducing any new technology must be done carefully. VPST can provide an alternate deployment for testing new technology before deploying it directly into real networks. The third use case is in analyzing the robustness of a design against attacks. Yardley concluded his talk by mentioning their future work on developing a black-box implementation of VPST for DETER.

Roy Maxion from CMU asked how they validate their results. Yardley replied that the system is not yet fully implemented and validation issues have not been fully addressed. Yardley also said that connection to real utility company networks is limited by legal constraints. Angelos Stavrou of GMU asked how they validate fidelity of each component in the network. Yardley said it depends on whether they are using models or real devices. For models it depends on the implementation of the model. Roy Maxion asked about the impact of errors introduced in simulation due to modeling proprietary devices. Yardley said that the issue had not yet been addressed.

Sessions below summarized by Eric Eide (eeide@cs.utah.edu)

- ***Dartmouth Internet Security Testbed (DIST): Building a Campus-wide Wireless Testbed***
  Sergey Bratus, David Kotz, Keren Tan, William Taylor, Anna Shubina, and Bennet Vance, Dartmouth College; Michael E. Locasto, George Mason University

Anna Shubina described her group's experiences in developing and deploying the wireless portion of the Dartmouth Internet Security Testbed (DIST). The wireless infrastructure supports experiments that require access to real-world network traffic. The hardware architecture includes 200+ WiFi access points, called "air monitors," distributed over ten buildings at Dartmouth. The air monitors send captured frames to DIST servers, which process the frames. An experiment describes the kinds of frames to be collected at the monitors and the processing steps to be run at the servers.

The software architecture is carefully designed to protect users' privacy and enforce experimenters' accountability. The air monitors discard all but the MAC layer of each captured frame. The frames are encrypted before being sent to the DIST servers; the servers decrypt and anonymize the frames before making them available to an experiment for analysis or storage. Unsanitized data is never written to disk. The testbed enforces accountability by keeping careful audit trails. For example, DIST policy is that an experiment's source code be checked into DIST's revision-control system before it can be deployed.

One of the technical lessons learned was that a long-running testbed in a production environment must be designed to survive unexpected changes. An unannounced change to Dartmouth's network highlighted the need for a fallback control channel to the air monitors. Shubina also described the many lessons learned in obtaining approval to deploy the wireless network at all. The project required extended negotiations with many organizations within Dartmouth, with issues ranging from the system's security architecture to the aesthetics of signage and the deployed hardware.

After the talk, a CSET attendee asked how often the encryption keys are changed at the air monitors. Shubina replied that they are changed for every experiment. In response to another question, Shubina said that their system does not stop collecting data when the number of network users is low; protecting privacy in such situations is a research issue. Finally, someone asked how long it took to solve all the administrative and social deployment issues. Shubina said that it took two years from start to end.

- ***An Emulation of GENI Access Control***
  Soner Sevinc and Larry Peterson, Princeton University; Trevor Jim and Mary Fernández, AT&T Labs Research

GENI is a planned testbed for exploring new network architectures at scale. It is designed as a federated testbed, with resources controlled by multiple administrative domains. As such, the evolving GENI security architecture is designed to support features such as distributed access control. In this talk, Soner Sevinc described an experiment that he and his colleagues performed to evaluate their design of a distributed access-control mechanism for GENI, driven by data collected from an existing large-scale testbed, PlanetLab.

To perform an operation in GENI, an agent must supply a set of cryptographically signed certificates to show that it is authorized. This involves collecting a chain of certificates, from the root GENI authority down, to establish the agent's identity and privileges. Building these chains means obtaining certificates from multiple administrative authorities. Soner and his colleagues designed a system to optimize the process of certificate collection. Their system, based on a framework called CERTDIST, handles both distribution of certificates and the evaluation of security policy. CERTDIST uses a distributed hash table (DHT) to cache certificates, load-balance requests, and provide fault tolerance.

How can this distributed access-control system for GENI be expected to perform in deployment? To answer that question, Soner and his co-authors started by collecting traces of access-control events in PlanetLab. From these traces, they produced equivalent scripts of GENI access-control events, translating from PlanetLab's centralized model onto their new distributed model. Finally, Soner and his colleagues used 550 PlanetLab nodes to carry out the events in the translated traces. Their experiments led to three main conclusions about the behavior of their distributed access-control system. First, the DHT effectively reduces the request load seen by certificate authorities, although the system still experiences minor "flash crowds" when popular certificates expire. Second, for the request load in the emulated traces, the DHT-based system does not reduce the latency of requests. Third, when the request load is increased by a factor of 10, the DHT improves the success rate of queries by balancing the load.

Future work will explore caching and retrieval strategies for certificates, to address the issues revealed by their evaluation. The PlanetLab traces that drove their experiments are publicly available at http://www.planet-lab.org/.

EXPERIMENTATION TOOLS

- ***Payoff Based IDS Evaluation***
  Michael Collins, RedJack, LLC

Michael Collins proposed a new approach for evaluating the efficiency of an intrusion detection system (IDS). The traditional method for evaluating an IDS is to view the system as a binary (yes/no) classifier: its false positive and negative rates measured as functions of the system's discrimination threshold. In contrast, Michael proposed modeling the IDS as if the attacker were aware of its capacities—treating the IDS as a constraint on the attacker's behavior and modeling how the attacker would respond.

The general idea is to model an IDS as a zero-sum game between an attacker and the IDS. The game is played over an "observable attack space" (OAS), which is defined by the set of attributes the IDS is designed to monitor. For example, if an IDS is designed to use flow data only, the OAS would not have attributes based on packet payloads. The OAS covers observations during normal network behavior and observations under network attack. For every point in the OAS, two functions are defined. The first is the payoff function: for an attack that maps to a particular OAS point, what value does the attacker receive? The second function describes detection: for a given OAS point, what is the probability that the IDS will detect the attacker? Given this setup, an attack is a multi-round game in which the attacker moves through the OAS, collecting the payoff values. After each attacker move, the IDS may detect the attacker and take corrective action. This model provides a basis for comparing intrusion detection systems: over a given OAS and period of time, the best IDS is the one that minimizes the attacker's total payoff.

Michael illustrated his IDS evaluation methodology over four games involving node acquisition (bots), network reconnaissance, maintaining a back-channel, and network saturation (DDoS). Using the evaluation methodology, for example, one can evaluate different strategies for a DDoS attacker. The game models presented in the talk were purely synthetic. Michael said that his future work will focus on developing more realistic models, based on real-world behavior.

After the talk, someone asked about models in which apparently "normal" network observations still permit high-payoff attacker behavior. Michael replied that this was an interesting question and a topic for future research into models of real-world observable behaviors and attacks. John McHugh asked whether network defense is not a two-party game but a multi-party game in which most of the players are normal users. Michael said that modeling intrusion detection as a three-party game might be reasonable; normal users might be modeled as a third party or as part of the game rules themselves.

- ***Toward Instrumenting Network Warfare Competitions to Generate Labeled Datasets***
  Benjamin Sangster, T.J. O'Connor, Thomas Cook, Robert Fanelli, Erik Dean, William J. Adams, Chris Morrell, and Gregory Conti, United States Military Academy

The final paper was presented by Benjamin Sangster and T.J. O'Connor, who shared their experience in collecting network traffic data from the 2009 Inter-Service Academy Cyber Defense Exercise (CDX). As described in another CSET talk, the CDX is an annual competition in which military academies defend networks from a National Security Agency (NSA) red team.

By instrumenting the CDX, the USMA team sought to address the lack of useful network-traffic data sets for security research. Most commonly used data sets are dated, artificial, and contain trivial artifacts: they are not representative of modern-day adversaries. In contrast, the CDX and similar network warfare games are designed to reflect the design and concerns of current networks: e.g., modern hardware and software, networks at scale, and threats such as zero-day attacks. The CDX involves human decision-makers as both attackers and defenders. A potential method for producing useful network traces for research, therefore, is to instrument network warfare competitions. This approach could automatically label the collected traffic as red-team (attacker), blue-team (defender), or white-team (ordinary use).

To evaluate this approach, and with the approval of NSA, the USMA team deployed three traffic-collection points during the 2009 CDX. One was placed at the border of the NSA team: it collected both red-team and white-team traffic from NSA. The second was installed on the network connection just inside the USMA team's VPN router, and the third was placed on the central switch of the USMA team's internal network. The second and third sensors therefore witnessed the ingress and egress filtering performed at the perimeter of the USMA network. They observed a mix of red, blue, and white traffic.

Benjamin and T.J. described the strengths and shortcomings of the data that were collected. They observed that the 2009 CDX dataset has a significantly different "personality" from some older DARPA datasets, due to the use of modern tools and the involvement of humans in the exercise. The CDX dataset is thus more representative of modern networks in both of these respects. However, the CDX dataset is limited by the nature of the exercise. It has less diversity and volume than a production network would have, and the dataset only covers a four-day period. It was also difficult to clearly label some traffic, for instance, due to the mixture of NSA red traffic with white "cover traffic." The authors believe that automatic labeling could be improved by collecting additional red-team logs, either automatically or manually.

The network data and other logs collected during the 2009 CDX are publicly available from http://www.itoc.usma.edu/research/dataset/.

PANEL ON SCIENCE OF SECURITY EXPERIMENTATION

Panelists: John McHugh, Dalhousie University; Jennifer Bayuk, Jennifer L. Bayuk LLC; Minaxi Gupta, Indiana University; Roy Maxion, Carnegie Mellon University

The final CSET event was a spirited panel discussion about the challenges in doing scientifically rigorous experiments on security topics. Jelena Mirkovic invited each of the panelists to start by describing his or her most important "hard problems" that stand in the way of scientific approaches to security. Each of these led to a great deal of discussion between the panel members and the audience.

Minaxi Gupta said that her favorite topics deal with access to data, both immediate and long-term. For instance, as a security researcher you may not know who has the data you want—and even if you do, you may not be able to get access to it. If you get the data you need, you may not have the resources needed to store it and analyze it. Finally, there is currently no standard practice for going backwards from publications to the datasets on which the publications are based. Minaxi concluded that the security community needs repositories that make long-term (multi-year) datasets available in real time, both raw datasets and derived data products. Doug Maughan and others at the workshop noted that the DHS PREDICT repository (https://www.predict.org/) is an important step toward making security datasets available to the public. Roy Maxion said that while it may be difficult to provide data to others, it is possible, and he offered a benchmark dataset for keystroke dynamics that accompanies a paper on his Web page (http://www.cs.cmu.edu/~maxion/). The data can be used for many tasks that are typical in intrusion and insider detection.

Jennifer Bayuk claimed that the hard problem is the "community problem." One aspect of this is competition, rather than cooperation, among security researchers: while researchers compete against each other, the attackers continue to advance. Competition over small problems does not help the community solve the actual problems being faced, such as how to make maximum use of existing tools and techniques in defense of common attacks. A second aspect is the lack of a basis for cooperation: problems that lack existing datasets are simply not being addressed. In response, John McHugh noted that datasets require a great deal of metadata in order to be useful. Sergey Bratus also added that recent testbeds, such as Dartmouth's DIST, can help to address the "unannotated dataset" problem by enforcing good practices.

John McHugh said that computer scientists have no excuses for bad science; they simply have bad practices. In general, computer scientists are not properly trained to conduct experimental science. They lack background in statistics, for example, and often do not collect data properly. McHugh gave an example in which an analysis of a large dataset was rendered invalid because the analysis assumed that clocks were synchronized over multiple data collectors. In fact, they were not—for most of the data-collection period.

The missing metadata for the dataset, which would have described how the data collectors were configured and calibrated, made the dataset significantly less valuable for scientific study. Finally, McHugh said that the requirements for funding and publishing are currently in conflict with rigorous science. Jelena Mirkovic suggested that funding agencies understand the need for good science, but the security community as a whole does not.

Roy Maxion said that the panel had not yet talked about what it means to have science in security. Science first requires having a tightly focused question—the hypothesis. Constructing a well-formed hypothesis is in fact a very difficult task, because it so often involves putting structure on an ill-structured problem. Second, science requires repeatability and reproducibility. Repeatability means that a single experimenter can perform a procedure several times and come up with the same result; reproducibility means that those results can be obtained by other investigators. Third, science depends on validity. Maxion asserted, "This is the issue that assails our field the most." Internal validity means that an experiment is logically consistent, and there are no explanations for the results obtained, other than the proposed explanation (e.g., no confounds). External validity means that the results are generalizable to a larger population. Maxion suggested that conference program committees demand better descriptions of experimental methods in submitted work. Anil Somayaji responded that the security community was still several steps away from rigor, because nobody currently builds on another person's work. The unanimous response from the panel was that the time for change has come!