;login:
DECEMBER 2014 VOL. 39, NO. 6
ELECTRONIC SUPPLEMENT

Conference Reports

CSET '14: 7th Workshop on Cyber Security Experimentation and Test

August 18, 2014, San Diego, CA

Summarized by Jason Gionta and Michael Rushanan

Panel: Cybersecurity Experimentation of the Future (CEF)

Summarized by Michael Rushanan (micharu123@gmail.com)

Moderator: David Balenson, SRI International
Panelists: Stephen Schwab, USC Information Sciences Institute (ISI); Eric Eide, University of Utah; Laura Tinnel, SRI International

David Balenson started the panel discussion by describing a crystal ball—what should the future of cybersecurity experimentation and testing look like? He noted that a fundamental shift in cybersecurity experimentation is required to meet forecasted problems of the future; specifically, he defined these problems as barriers and limitations researchers experience when studying a cybersecurity need. For example, scale and flexibility are required to discover, validate, and perform ongoing analysis, while keeping pace with changes in technology and extending existing infrastructure (e.g., wireless).

David addressed the panelists with an outline of high-level themes that he would like to see in the discussion: What kind of experiments will be conducted in the future; what are the experimental approaches/methodologies? What are the important characteristics of an experimental infrastructure (e.g., embedded)? What are the capabilities needed to support experimentation? What are the critical gaps of where we are now and where we need to be?

Stephen Schwab kicked off the first discussion by pointing to problems that yield insight and present grand challenges. Steve suggested that more experiments will shift the feasibility frontier by considering extensible and common interfaces—specifically, providing the appropriate interfaces to the usable world.

Eric Eide provided personal insight with respect to test bed development. He said that we are good at provisioning, configuring devices, and performing experiments. However, we are not so good at thinking about what we are doing with rigorous experimental design, being precise about what we want to measure, and handling hidden factors. He suggested that we should ask ourselves what response variables of a system do we want to study, what are we trying achieve, what are the inputs? Eric recommended a solution that considers tangible things (e.g., a networked PC running Windows) because malware is not a closed system. Obviously, this comes at the cost of time, effort, software capability, scale, and setup.

Laura Tinnel commented that many researchers spend a lot of time standing up their own test environment. This includes custom software and environments that are unique and tailored to their needs. The downside of this approach is that there is a large initial cost that could have been spent doing research. In addition, researchers using these environments are accustomed to encountering bugs and errors. Laura suggested incremental improvement on today's processes and access to shared resources (e.g., a repository), providing general frameworks and architectures to all researchers. She also suggested a domain-specific language to define experimentation in, enabling verification of experimental frameworks pre-deployment.

A workshop attendee commented that cybersecurity originated from a game and that to understand the problem we need to better understand the human behavior that drives the game. Laura commented that the shared repository approach would allow us to also share adversarial models, some of which might model behavior. The attendee replied that modeling bad behavior is hard and that shared repositories alone would not address this issue. Eric mentioned projects such as Planet Lab, stating that some services provide interfaces for people to come and actually use it, modeling experimentation that isn't repeatable, but more in line with human factors.

A workshop attendee posed a question to the group with respect to the top three repeatable experiments in the domain. The panelists could not recall any specific experiment names, but Stephen recalled a recent experiment that dealt with humans and picking passwords. The experiment had a well-defined hypothesis, mathematical model, and a result that captured what they already had an intuition for. However, the obvious lack of any top experiment names made the workshop attendee ask why? Other disciplines have famous experiments.

Stephen replied with an example of an anti-censorship study and how the output of the experiment outweighed the methodology for reaching the conclusion. In addition, for this case, certain functions (e.g., collection methods) might not be that direct or considered private for safety reasons.

Someone commented that Laura's slides looked familiar and said that we have an accessible testbed, a well-defined framework, and a language to describe experimentation. That's been done, many times in fact. It always has ended in failure. Laura replied that it might be okay to have more than one experimental framework. Steve added that the open source paradigm is changing the way we do things, and it's a good change (e.g., increased productivity). He stated that it would certainly be a failure to not latch on to the mainstream and embrace an open source approach.

Tobias Fiebig commented that the current state of experimentation is small, unreproducible, and unpublished. He thinks that experimentation frameworks should be published with the data in order to evaluate the results and approach. Eric agreed. He believes that this would bring more value to published results, and could enable the measurement of impact with respect to opening up these tools. Tobias Fiebig offered a final comment regarding documenting methodology and data. He has seen a negative opinion toward this approach in science. There is a perceived barrier to sharing (e.g., using code to get the next publication).

Metrics for Quantitative Security Evaluation

Summarized by Michael Rushanan (micharu123@gmail.com)

Effective Entropy: Security-centric Metric for Memory Randomization Techniques

William Herlands, Thomas Hobson, and Paula J. Donovan, MIT Lincoln Laboratory

Thomas Hobson started his discussion with a definition of address space layout randomization (ASLR), a technique to protect against memory corruption attacks by changing the layout of objects in memory. There are a number of variations of ASLR (static, position independent executable, and theoretical fine-grained randomization), and Thomas asserted that we need to measure them to quantify their security benefit. There are two methods for measuring the security benefit: testing a set of exploits (i.e., real attacks) and measuring the amount of entropy provided to each section in the system. Thomas recommended the second method because the other is limiting with respect to identifying fundamental flaws.

Thomas defined effective entropy as a metric for accounting for pointers between memory sections; or, with respect to the adversarial model, determining adversarial capability for circumventing entropy. Thomas then depicted the effective entropy metric with a table that lists sections (e.g., heap and DLLs) paired with integer values (e.g., DLLs have eight bits of effective entropy). He described how to quantify effective entropy with an example: There is an exec function in nmap that an attacker wants, the attacker targets the program image with the knowledge of a gadget to redirect control flow and the address of the exec pointer, and the attacker exploits a stack-based vulnerability and overwrites the return address with the gadget. Thus, nmap in this example has 0 bits of entropy.

Thomas provided more technical detail to his example by describing how he distinguishes static and dynamic pointers. The process includes running the program, pausing it, examining all memory and all registers, and examining all byte offsets to create a list of candidate pointers. He then runs the program a second time and applies randomizations to compare with the first run. Lastly, the effective entropy is computed by examining connections (e.g., if an executable section is connected, it has effective entropy of 0).

Jason Gionta asked whether the effective entropy technique was being done at the instruction level. Thomas replied that they are not yet doing it at this level, but they could extend to their implementation to do this. Eric Eide asked whether effective entropy could be increased by reducing cross object pointers. Thomas gave his intuition on the topic and said that it would be hard to completely remove connections as some process may require it. Eric also commented that a lot of sections seemed to have 0 bits of initial entropy. He then asked whether there were other segments that also had 0 bits of entropy. Thomas replied that when the attacker starts a new program image, it has 0 bits entropy. Jason asked whether disclosing one section allowed an attacker to determine all other sections. Thomas responded that, in the case of position independent executable ASLR, he was able to find all other memory sections. Jason also asked about the tool Thomas and his colleagues used for finding gadgets in memory, and Thomas answered that it was an in-house tool.

DACSA: A Decoupled Architecture for Cloud Security Analysis

Jason Gionta, North Carolina State University; Ahmed Azab, Samsung Electronics Co., Ltd.; William Enck and Peng Ning, North Carolina State University; Xiaolan Zhang, Google Inc.

Jason Gionta introduced us to the cloud provider landscape (e.g., platform-as-a-service) and what its existence means for developers and service providers: reduced costs, increased availability, and scalability. However, studies show that clouds are also used for nefarious purposes. An example of this type of activity includes a recent disclosure of a DoS botnet built using EC2. The problem, Jason stated, was how to ask security-centric questions regarding the clou-—for example, what bad things are people doing in the cloud.

Jason proposed using any and all information that cloud providers can collect to do analysis across the infrastructure, transforming the cloud infrastructure into a security testbed. Jason outlined the types of data sources available to cloud providers that could be used: network data, virtual machine monitoring, and out-of-virtual machine monitoring. He ruled against network data because encryption presents a loss of context, and virtual machine monitoring can directly impact resources. Out-of-virtual machine monitoring is promising because scans can record and replay a virtual machine, but the current approach lacks scalability.

Jason described the need to decouple analysis from data acquisition, thus limiting the impact on both host and client. He introduced DACCA as the architecture for achieving just this. Data sources are considered sensors, and each sensor will have some context for acquisition. This data is carved out from the virtual machine and then analyzed.

Jason stated that this would enable cloud providers to test for security violations such as a botnet running on a virtual machine. Jason reported that DACSA is capable of fast snapshotting of virtual machines by making a logical copy of guest memory with the copy and write feature, limiting the impact on the host. Jason also reported a minimal impact to virtual machines being analyzed (3% CPU increase and 0% memory utilization). DACSA efficacy was evaluated with real malware, the Cerberus remote access tool in this case, and it was successfully identified.

A workshop attendee asked about the difference between this work and VM introspection. Jason answered that introspection techniques are heavy on resource utilization and thus aren't being fully utilized. In addition, he pointed out that cloud providers are not doing live monitoring and thus are losing semantic knowledge of how the machine is running. The attendee followed up by asking how this approach affected privacy? Jason responded with the traditional trust relationship between the user and the cloud provider, and that his technique requires no additional trust than already assumed.

Jason was asked if his technique would require a change in the terms-of-service, and he responded yes, noting that it would be an opt-in type of control. Eric Eide commented on the optimized snapshot and that the VMM has access to all data. Jason responded that data will not be accessed and is treated as a black box. Eric then asked what analysis was being done now? Jason replied that virus scanning as a service is being done now, and that a full scan of a 1 GB VM takes approximately 10 minutes. A final question was whether the technique required a full memory dump each time. Jason informed the group that this is not the case, and in fact you can scan the process to dump those pages specifically.

A Metric for the Evaluation and Comparison of Keylogger Performance

Tobias Fiebig, Janis Danisevskis, and Marta Piekarska, Technische Universität Berlin

Tobias began the presentation by describing an issue he had found in a smartphone application GPU library that can read bitmaps attached to the GPU without sufficient privileges. Tobias stated that he was interested in the performance of the keylogger; for example, how long does it take to recover passwords, how many iterations are required, and how could he make this reproducible?

Tobias then described his experimental setup to evaluate the viability and reproducibility of his keylogger. The setup uses a small test study that includes one keylogger and two test environments. Both environments included a full capture of what the user types into a smartphone and tablet device, providing ground truth for comparison against what the keylogger captures. Tobias found that his keylogger performed differently between the devices.

In fact, the keylogger worked better for the smartphone than for the tablet, and he was able to observe some bias due to a higher clocked CPU on the tablet. This result enabled Tobias to filter keys that were missed and other defects to remove biases. Tobias has provided this testing framework to the open source community via GitHub.

A workshop attendee asked whether timing could be included. Tobias said that the implementation locks the time. Jason Gionta commented that this work seems like a vulnerability per the mobile device, and he asked whether Tobias had contacted the vendor. Tobias said that he had. Eric Eide asked whether Tobias was worried about the nefarious use of his work that he open sourced. Tobias stated that he is interested in feedback of his open sourced work but he did not address nefarious use.

Panel: Human Engagement Challenges in Cyber Testing and Training

Summarized by Jason Gionta (jjgionta@ncsu.edu)

Moderator: Chris Kanich, University of Illinois at Chicago
Panelists: Jose Fernandez, École Polytechnique de Montréal; Stefan Boesen, Dartmouth College; Richard Weiss, The Evergreen State College; Melissa Danforth, California State University, Bakersfield

The panel began with short presentations of each of the three extended abstracts. First, Jose Fernandez presented a four-month pilot study including 50 users to look at how malware infects computers and how user behavior affects the probability of infection. Jose outlined the approval process that was taken to run the pilot and setting up the collection platform. All participants were sold laptops containing the same configuration. Over four months, participants' behaviors were tracked by monitoring software to capture all applications installed, updates applied, files downloaded, different locations connected to the Internet, number of hours per day the laptop was connected to the Internet, among other dimensions. Jose stressed the importance of estimating the required population size and strategically selecting users on a scientific and ethical basis. Finally, Jose stressed the importance of defining what data was needed to do the analysis and developing an analysis for the large amount of data collected.

Richard Weiss spoke about EDURange and its use in teaching cybersecurity skills. The authors architected EDURange focusing on flexibility and stability of the teaching platform while honing student analysis skills. Richard discussed experiences and challenges with rapidly deploying a teaching environment to classroom students. Examples included challenges in distributing a common teaching platform (e.g., VMs) and distributing credentials. EDURange is hosted on EC2, allowing students and teachers to access resources on-demand via Web browser or SSH client. Lessons learned from the experience are that students need significant scaffolding for guidance requiring more materials and tools to assist in learning, while faculty want immediate exercises.

Finally, Melissa Danforth presented on experiences running and teaching cybersecurity concepts to high school students as part of a four-week summer program held at California State University Bakersfield. Students attended the program four days a week for six hours a day and received a stipend. Students were chosen based on an application and rank preference. The area was Hispanic serving, and the only requirement was math preparedness. The ratio of males and females was 50/50. Two cybersecurity-centric sessions were provided: cryptography and general security. The cryptography session began with substitution ciphers and modular arithmetic. Next, the session covered Fermat's little theorem, modular exponentiation, and RSA encryption. The session ended with factoring RSA and elliptic curves. The general security sessions started with the ethics and the legality of undermining security protections and cracking passwords. The session went on to cover secure authentication, network attacks, and social engineering. Finally, malware and access control were discussed. The sessions culminated in posters that can be found at http://www.cs.csub.edu/~melissa/revs-up/.

Someone asked the panelists if they had IRB approval to be working with users. Richard said that the small institutions make it easier to get permissions. The IRB at Evergreen required a letter be sent to each participant. Melissa replied that she went through the IRB and ethics coordinator for a survey that was passed out to students. She attended training on ethics protocol. Jose said he had to go through the computer security board risk committee, which imposed constraints that they could not collect browsing history.

Someone else asked about how EDURange is paid for and whether the platform is available. Richard stated that EDURange hosting is supported by an Amazon grant. EDURange is run on EC2 micro-instances, costing about 1.3 cents per hour. The code can be found at https://github.com/edurange/edurange. Comments were made about the DETER testbed as a turnkey system for sharing and teaching cyber education.

Another person asked whether what they had done changed what they would do in the future. Jose said he is writing a book on cybersecurity. He found that young people are getting infected faster than older people. Richard replied that he will focus on assessment and how to score exercises on EDURange. In addition, he will concentrate on how to write exercises that will be used to teach a concept. Richard plans to work with faculty at other schools to teach cybersecurity using EDURange. Melissa will look at students as more capable of understanding the very complex concepts discussed in the sessions. Melissa plans to have more hands-on activities for the first week and cross-year focus with different modules.

Someone asked about exposing students to the dark side. Jose said it's difficult to teach dark-side concepts given the low-level knowledge required. Richard said dark-side concepts can make learning assembly more interesting and is essential to understanding how a program works. Furthermore, many of the skills used by the dark side are also used by software engineers. Melissa stated that many students had no C/C++ experience, so they taught concepts through diagrams. They highlighted how easy it is to make mistakes or be taken advantage of via spoofed Web sites. Melissa spoke about the lessons on password cracking and the use of salts. This provided insights into secure design requirements.

Interested parties can sign up for EDURange at edurange.org, or contact Jose at jose.fernandez@polymtl.ca for access to their data sets. Melissa has lots of teaching materials at http://www.cs.csub.edu/~melissa/revs-up/.