conference reports

THANKS TO OUR SUMMARIZERS

19th USENIX Security Symposium (USENIX Security '10)

Washington, DC
August 11–13, 2010

OPENING REMARKS AND AWARDS PRESENTATION

Program Chair: Ian Goldberg, University of Waterloo

Summarized by Rik Farrow (rik@usenix.org)

Ian Goldberg thanked the USENIX staff and the program committee, then said there were a record number of submissions for this year's conference: 207 papers were submitted. Five were rejected for double submissions or plagiarism, and 42 more papers were rejected in the first round of reviews. Each Program Committee member read 20–22 papers, with David Wagner reading 38. In the end, 30 papers were accepted, with Capsicum (Watson et al.) winning Best Student Paper, and Vex (Bandhakavi et al.) Best Paper.

KEYNOTE ADDRESS

- ***Proving Voltaire Right: Security Blunders Dumber Than Dog Snot***

Roger G. Johnston, Vulnerability Assessment Team, Argonne National Laboratory

Summarized by Veronika Strnadova (vstrnado@unm.edu)

Johnston began by pointing out numerous, all too common mistakes that his team encounters when assessing security vulnerabilities, along with some countermeasures. Many of these mistakes are avoidable and many vulnerabilities are fixable, but Johnston said that the big problem comes from the fact that a lot of people don't exploit security resources.

One ineffective response to a security threat is the use of what Johnston dubbed "security theater," the practices which involve providing a "feel-good effect" for the public instead of making a true effort to increase security. A glaring example of this is the no-fly list or color-coded security threat level we see at airports. While security theater may provide comfort to some people, many vulnerabilities are being overlooked—Johnston and his team were able to tamper with voting machines, spoof GPS devices, break into containers with expensive cargo (breaking and then replacing seals), proving that common sense is still a much-needed tool not being used in security.

Johnston talked about some alarming vulnerabilities that arise from oversights such as not doing background checks on IAEA nuclear site inspectors, not setting a microprocessor's security bit, or not masking passwords and sensitive data. Most importantly, many people and/or companies don't take Johnston's advice or implement even the simplest security measures. Johnston said that

this often comes from a fear that admitting vulnerabilities means admitting weakness or ineffectiveness. In response to a question from Rik Farrow, he said that when people don't take his advice, he doesn't take it personally and that his team offers solutions but can't expect everyone to implement them.

Peter Neumann asked about implementing common sense, and Johnston suggested that people in physical and cyber security can help and learn from each other to fix security blunders. The two cultures need to learn to work together to avoid making the same mistakes over and over. He said his team has learned that telling people that not everything can be fixed helps alleviate the pressure of improving security. Someone else wondered whether people get fired for vulnerabilities his team finds. Johnston said this only happens after a discovered vulnerability gets exploited. According to Johnston, this is the worst time to fire people (usually not the ones at fault) who can help counter the security threat, but it happens often in physical security. Matt Blaze was impressed with the work done to switch votes in a voting machine, but wondered if it would have been easier to replace the printed overlay. Johnston agreed this was a possibility, but pointed out that his hardware attack could be done at more points before voting.

Johnston said that security needs to be thought about when designing a product, not as an afterthought. Security is not usually something that can be added on, and adding on "layers" of security only makes it more difficult to monitor when security has been broken. Countermeasures are often simple—seals should be unique and difficult to break, the order of candidates in voting machines should be randomized, and people within companies can often find vulnerabilities in security at no extra cost to the company.

No matter how many suggestions Johnston's team makes, the major problem is that there is often no interest in implementing them. There is a need for research-based countermeasures to security threats, but "security theater" is easier. The laziness and blind faith in security "authorities" who often have no experience in dealing with security are the first vulnerabilities that need to be changed.

PROTECTION MECHANISMS

Summarized by Zhiqiang Lin (zlin@cs.purdue.edu)

- ***Adapting Software Fault Isolation to Contemporary CPU Architectures***
  David Sehr, Robert Muth, Cliff Biffle, Victor Khimenko, Egor Pasko, Karl Schimpf, Bennet Yee, and Brad Chen, Google, Inc.

Software Fault Isolation (SFI) is an effective approach to sandboxing untrusted binary code. Native Client (NaCl) is an interesting use case of SFI for running native code in Web applications. Through extending NaCl, David Sehr presented new schemes on how to make SFI effective on some contemporary CPU architectures, namely ARM and x86-64. Their implementations on these two architectures are called ARM-SFI and x86-64-SFI, which are the best known SFI implementations with significantly lower overhead (under 5% on ARM and 7% on x86-64) than previous systems.

Their work was prompted by Web application scenarios in which programmers tend to use their own favorite language but like to have some features such as screening malicious instructions, system calls moderated by a virtualized OS, and performance within 5% of native code execution. With NaCl, users do have these benefits to run applications such as Star Wars and Nexuiz (an openGl Quake) inside the Web browser. Sehr provided some background on SFI such as creating an untrusted memory segment and using instructions to enforce segment boundaries, the control safety and data safety in SFI, and the SFI implementation.

In particular, for SFI implementation, Sehr outlined two basic approaches: using hardware support (e.g., x86 segmentation), and inline guard instruction sequences which require aligned instruction blocks. He also talked about some background on stack pointer optimization proposed by McCamant and Morrisett. After covering the architectures of x86-32, x86-64, and ARM at a high level, he described how they implemented their ARM-SFI and x86-64-SFI. More specifically, he talked about how they ensure the control safety, data safety, and stack updates for untrusted code in ARM and x86-64, respectively. For the performance, they are very pleased with the ARM-SFI, and the results are fairly consistently around an average of 5%. However, interestingly, the performance of x86-64 is bimodal: where code size is important, overhead rises to 30%; where code size is not significant, overhead is low. Their code is available at http://code.google.com/p/nativeclient.

Peter Neumann commented that there is another piece of work called BackerSField by Joshua Kroll. Neumann wondered whether Sehr has discussed this with Kroll, as Google had hired him this summer. Sehr responded that they did have brief discussions on sandboxing improvement (e.g., performance). Another person asked about the wisdom of research on solving problems we probably already have solutions for. Sehr answered that they first looked at the problem from a performance perspective, and they found there are still gaps to be closed. Also, there are still some theoretical problems to explore, such as the formal models of instruction sequences in different architectures they are targeting, and how to make the validator do more formal verifications.

- ***Making Linux Protection Mechanisms Egalitarian with UserFS***
  Taesoo Kim and Nickolai Zeldovich, MIT CSAIL

UserFS "provides egalitarian OS protection mechanisms in Linux [and] allows any user to allocate . . . UNIX user IDs, to use chroot, and to set up firewall rules in order to con-

fine untrusted code." Taesoo Kim began his talk on how to build secure applications, which is simple in principle, but it is difficult in practice (unless you're a root user) because normal users cannot create new principals and cannot reduce privileges. Thus, his talk is about how to help programmers reduce privileges and enforce security policy in Linux by allocating and managing UIDs.

Kim used DokuWiki as a running example to illustrate their techniques. DokuWiki is a PHP-based wiki, and it runs as a single UID but has a number of users. As such, DokuWiki has to perform ACL checks when different users access particular files. If a programmer missed any ACL checks, it could lead to an insufficient permission check vulnerability. The goal of UserFS is to allow any application to use existing protection mechanisms without root privileges, such as creating a new principal, reusing existing protection mechanisms, and using chroot and firewall mechanisms. One key idea in UserFS is to represent user IDs as files in a /proc-like file system, thus allowing applications to manage user IDs like any other files, by setting permissions and passing file descriptors over UNIX domain sockets. There are several challenges in making user IDs egalitarian, including how to reuse UIDs, how to make UIDs persistent, accountability, and resource allocation.

The authors have implemented UserFS as a single kernel module with 3000 lines of code on Linux 2.6.31 using Linux Security Module, Netfilter, and the Virtual File System. In their evaluation, they have modified five applications to take advantage of UserFS. By changing just tens to hundreds of lines of code, they prevented attackers from exploiting application-level vulnerabilities, such as code injection or missing ACL checks in a PHP-based wiki application. Also, UserFS incurs no performance overhead for most operations, making it practical to deploy on real systems. Kim also discussed the limitations of UserFS, such as UID generation numbers only tracked for setuid binaries, and GID allocation not implemented in their current prototype; their future work is to allow a process to have multiple concurrent UIDs.

Someone asked why UserFS didn't store IDs for other files instead of only tracking the generation IDs for setuid binaries. Kim answered, "Because of performance." Someone asked about the impact on the resource limit for UserFS if the resource quota turns on. Kim replied that the programmer in that case has to use set on the resource system call in Linux. The third question concerned comparisons with Plan-9 from Bell Labs—more particularly, on the fact token with the notion of user ID as files, and on who guards the permissions in UserFS. David Reed said that UserFS is a nice mechanism and was curious about the generality of UserFS when compared with state-of-the-art capabilities, cross-domains, etc.

- ***Capsicum: Practical Capabilities for UNIX***
  Robert N.M. Watson and Jonathan Anderson, University of Cambridge; Ben Laurie and Kris Kennaway, Google UK Ltd.

  Awarded Best Student Paper!

Robert N.M. Watson presented Capsicum, a lightweight OS capability and sandbox framework, which extends the POSIX API and provides several new kernel primitives (e.g., sandboxed capability mode and capabilities) and a user-space sandbox API to support object-capability security for UNIX-like OSes. It supplements rather than replaces DAC and MAC.

Watson first described the paradigm shift from multi-user machines to multi-machine users and compartmentalized applications, and from DAC/MAC-centric access control to sandboxing. We are living in a world of browsers which can visit many Web sites (e.g., Webmail, YouTube, bank account) with very different technologies (e.g., traditionally static Web page, dynamic Web pages, virtual machines, and scripting languages). But Web browsers do have security vulnerabilities in large quantities. Watson mentioned the existing work, including microkernels, MAC, and Type Enforcement, to motivate their capability system. A capability is an unforgeable token of authority, and it supports delegation-centric access control. Capsicum supports capabilities with refined file descriptors with fine-grained rights, has a capability mode in which the sandbox denies access to the global namespace, and contains libcapsicum, a library to create and use capabilities and sandboxed components. To demonstrate Capsicum, the authors have added self-compartmentalization to a number of UNIX applications and core system libraries, including tcpdump, dhclient, and gzip using Capsicum. In collaboration with Google, they also have adapted the Chromium Web browser to use Capsicum, showing significant programmability and security benefits over its existing use of UNIX DAC and MAC security primitives. They prototyped Capsicum on FreeBSD 8.x, and their experimental code is BSD-licensed.

Peter Neumann asked about the dichotomy between capabilities and mandatory access control, given that, historically, systems in the 1970s that adopted this approach worked. Watson answered that the two composed quite well, but that things might prove more interesting in the case of applications already constrained by Type Enforcement when the use of capabilities was indicated. Helen Wang commented that capability-based sandboxing is definitely the right way to go, especially for the Web browser. She observed the similarity between the Chromium browser structure presented and the Gazelle project, as well as the observation regarding multi-user vs. single-user OSes. Watson responded that the browser architecture used under Capsicum is the model already present in Chromium, but that Chromium would benefit from much more use of sandboxing; another concern is how to address the windowing system. One exciting change has been in the use of new security models in mobile phones (such as the iPhone and Android),

where breaking existing applications was acceptable. Wang pointed out that the Web single-origin policy is one of the areas where Web browsers have gotten things right. Jinpeng Wei asked how to handle revocation capability in Capsicum. Watson answered that the revocation of capabilities is usually done through interposition, which is supported for userland capabilities (IPC objects) but not yet for kernel objects. Crispin Cowan asked for a comparison of AppArmor with Capsicum. Watson replied that AppArmor and Mac OS X's Seatbelt are very similar systems in terms of how policies are bound to applications, so a similar analysis would likely apply, but that a capability-oriented architecture had significant benefits.

INVITED TALK

- ***Toward an Open and Secure Platform for Using the Web***
  Will Drewry, Software Security Engineer, Google

  Summarized by Joshua Schiffman (jschiffm@cse.psu.edu)

Will Drewry began by discussing the design goals of Chrome OS and how they address the concerns of the typical user, who is unaware of security and unsafe browsing practices. In particular, he outlined three main areas: survivability of the system, data protection, and the openness of the platform. Starting from a baseline of a simple Linux distribution using Gentoo Linux's portage and no user-installed local applications, users only interact through the Chrome running on Xorg, which is supported by a mix of new and existing daemons underneath.

In terms of survivability, Google wanted a system that can recover from most forms of compromise, such as rootkits, trojans, BIOS modification, etc. To do this, they use a combination of mitigation techniques popular on clients such as ASLR, default non-execute heap and stack, sandboxing Chrome's renderers, and DAC. They also included protection features used on servers, such as a read-only root file system, restricted mount flags for non-rootfs, a set of kernel patches, and capabilities. They also use grsecurity and Tomoyo for mandatory access control instead of AppArmor or SELinux, because the Google team felt it was easier to keep the MAC policies in sync with feature development using them. Another tool they added was Breakpad, which is Google's crash dump logger that was linked into every binary.

Drewry then discussed how Chrome OS performs auto-updates by dividing the rootfs into an active and passive partition, which is then swapped after an update is applied to the passive partition and verified at boot. Updates are streamed to the disk using delta differences based on a block dependency graph, which Drewry noted was more efficient than something like bsdiff that requires the entire file system to be loaded into memory. Drewry also mentioned that Trusted Platform Module (TPM) was used only for its lockable NVRAM to provide rollback protection, but not for measuring files, since the Google team did not want to deal with managing the administrative passwords the TPM requires for those features.

Drewry then moved on to how the active partition was verified at boot time in order to assure users that Chrome OS is currently running. This approach uses a Static Root of Trust model to help prevent persistent basic attacks. The root of trust is a key that lives in read-only firmware, which verifies a subkey in a writeable firmware portion. This subkey is then used to verify the RW firmware, and this process is then repeated for the OS kernel and command line in the rootfs. Verification of the rootfs is done using a hash tree approach whereby each 4KB block is hashed and then 4KB of hashes is hashed repeatedly to form a single root hash that is then passed as a kernel line parameter. In this way, the OS can check the rootfs incrementally instead of all at once, which takes longer and slows the boot process.

Drewry then moved to the second design goal, which is protecting user data. Currently, users log into Chrome OS using their Google accounts or Google Account for your Domain. In the future, they would like to support OpenID providers, but mentioned that there are issues with passing attributes and that generic programmatic Web login is an open challenge. He also mentioned that users could browse without signing in and that such sessions are stored in a tmpfs. Actual user accounts have their data protected by a daemon known as Cryptohome, which manages user partitions encrypted with eCryptfs. This daemon is used in place of the standard eCryptfs utilities and handles offline authentication and partition keys. A user's passphrase is needed for decryption, but they mitigate brute force attempts by wrapping the derived key with a TPM key, which forces a brute force attacker to be subjected to slow hardware. If a TPM is not available, Colin Percival's scrypt for memory-hardening is used.

On the topic of openness, Drewry mentioned that Chrome OS is based on the open source Chromium OS and that the team frequently contributes back to the project. On the hardware side, a developer mode switch is specified to be under the battery to let knowledgeable users disable the boot process and load self-signed OS images. This process clears the TPM and zeroes RAM to prevent this from being abused by attackers that use boot shims to read memory, but does not prevent more sophisticated cold boot attacks. In the future, he said, they would like to integrate a platform-supported trusted UI and a peripheral firmware validation mechanism.

Someone asked about the time frame for official Chrome OS laptops, and Drewry answeredthat it would be sometime this year. Another audience member asked whether Chrome OS could function as a VM. Chrome OS would not, but Chromium is QEMU-friendly and has been shown to run in KVM and Virtual Box. One reason for not supporting VMs in Chrome OS is the difficulty in verifying the system, but one option would be to use some features in the EFI standard. Drewry also demoed a reboot of a Chrome OS laptop,

which took about 12 seconds due to the unmodified Dell firmware.

PRIVACY

Summarized by Tamara Denning (tdenning@cs.washington.edu)

- ***Structuring Protocol Implementations to Protect Sensitive Data***
  Petr Marchenko and Brad Karp, University College London

Petr Marchenko presented this work via Skype. The goal of this work is to help protect sensitive data in network applications such as Web servers. While SSL connections protect the confidentiality and integrity of customer data while in transit, they do not guarantee these properties if an attacker exploits a vulnerability in the Web application itself.

While breaking down the application into compartments and applying the principle of least privilege helps with security, current applications do not treat the session key as privileged information. Additionally, if an attacker compromises an unprivileged compartment, he can get a privileged compartment to sign arbitrary data in what is known as an oracle attack.

The researchers identified these attacks and developed some defenses. They employ a "session key barrier" by creating new unprivileged compartments after the session key negotiation. In addition, they created nine principles to prevent against oracle attacks. The researchers created a hardened version of the OpenSSH client and server and a drop-in replacement for the OpenSSL library that reduce the TCB and protect against the session key attack and all known oracle attacks.

Questions from the audience addressed the aspects of a Web application that are not protected by these defenses. For example, if user data is handled by an unprivileged compartment that can be compromised by an attacker, there is still a breach in the confidentiality of the user's data. The speaker commented that the work is focused on one specific aspect of the security of network applications and that other vulnerabilities may still exist.

- ***PrETP: Privacy-Preserving Electronic Toll Pricing***
  Josep Balasch, Alfredo Rial, Carmela Troncoso, Bart Preneel, and Ingrid Verbauwhede, IBBT-K.U. Leuven, ESAT/COSIC; Christophe Geuens, K.U. Leuven, ICRI

Josep Balasch presented this work on privacy for electronic toll systems. The EU has chosen to employ satellite-based electronic toll systems that consist of a GPS and GSM in an on-board unit (OBU) in a user's car. The system tracks the user's location and periodically sends information to the toll server. The goal of this work is to help design an electronic toll pricing system that respects user privacy, provides user verifiability of costs, and allows the providers to detect misuse of the system.

To protect a user's privacy, one solution is to have fee calculations be done on the OBU. However, this provides the user with the opportunity to tamper with price tables, subfees, and fee totals before their transmission to the provider. These attacks are in addition to the possibility of performing GPS spoofing or turning off the OBU. The researchers propose employing TPMs and homomorphic commitments to prices as part of a proof of adherence for the provider. These cryptographic techniques would be used in addition to spot checks performed by the provider (for example, via license plate cameras).

The researchers built a proof-of-concept OBU and demonstrated that the OBU and server time required are practical. For example, using 1-mile road segments and processing GPS strings every second, 1536-bit keys support a maximum vehicle speed of 124 mph. The researchers also showed that the amount of server processing required supports a reasonable number of vehicles on the road.

An audience question brought up a discussion of interoperability between EU countries and the different levels of interest in privacy in those countries. While the EU plan for an electronic toll system has no privacy requirements, some countries have expressed an interest in making the system privacy-respecting. Another question addressed the possibility of optimizations in the system implementation, such as using elliptic curves. Balasch said that while their system employed optimized assembly routines, no attempts were made to optimize the cryptography involved.

- ***An Analysis of Private Browsing Modes in Modern Browsers***
  Gaurav Aggarwal and Elie Burzstein, Stanford University; Collin Jackson, CMU; Dan Boneh, Stanford University

Elie Burzstein presented this work on a general survey of the characteristics of private browsing modes in different modern browsers and the characteristics of users who employ private browsing modes. The researchers gathered data on the browsing histories of browsers in private mode by purchasing ads and detecting whether or not a link displays as having been visited.

The researchers found that private browsing mode is most common in users of Safari and Firefox, and that private browsing is most often employed when visiting sites with adult content. The authors presented a theory that private browsing is so prevalent in Safari because the private mode indicator is discreet, and therefore easy to leave on accidentally.

The researchers also analyzed the private browsing mode behavior of common browsers by leveraging unit tests. Browser violations of indistinguishability after a private browsing session included SSL certificates and site-specific preferences. Additionally, popular browser extensions frequently do not check for private mode or alter their behavior accordingly. The authors propose manual review of extensions to check that they are privacy-respecting and an

opt-in model for extensions when running in private browsing mode.

Someone brought up the question of whether, since studies show that the primary use of private browsing mode is for adult content, researchers are not acknowledging or addressing the main uses of privacy technologies in their discussions and research. Another question regarded the threat model of this work, which assumes that the computer is under user control until the private browsing session begins, and therefore excludes scenarios such as IT-managed computers in work settings.

INVITED TALK

- ***Windows 7 Security from a UNIX Perspective***
  Crispin Cowan, Senior Program Manager, Window Core Security, Microsoft, Inc.

  Summarized by Adam J. Aviv (aviv@cis.upenn.edu)

"Windows" and "security" are not words normally placed in the same context, especially not at USENIX Security, but it was the primary focus of a very well-attended invited talk in front of a raucous crowd. In the introduction, Crispin Cowan was described as one of the most outspoken critics of Windows just five years ago, but as of 2008, he is on the "other side." As a project manager at Microsoft, he was brought in to help with security on an OS that needed it, and, in his own words, "It hasn't been disputed that they needed it."

Cowan's thesis is simple. Yes, back in the day, Windows had lousy security (if any), but now Windows is leading the way. He provided a few key examples of this; some received jeers from the crowd, others nods of approval. Overall, it was an exciting talk that held the attention of everyone in the room.

Cowan began by describing the state of the world prior to Microsoft's security revelation. From Windows 3.1, 95, 98, up to XP, "all code that got to run on the box had complete ownership of the box." There were no privileged or unprivileged users; everything essentially ran as root: "Run as non-root, good. Run as root, not so good." He noted that NT was fundamentally a secure OS, but Windows applications grew up in a world without privileges, and so the default user had to be administrator, which just defeated the whole purpose.

This issue was just a small part of the "coin-operated" computer design of the time, pushing functionality over security. This was fine until the rise of the Internet. Then it became about not just what a user can do, but what others could do. In 2002, a memo by Bill Gates was circulated, saying as paraphrased by Cowan, "You will learn security."

This brought about a number of changes in the Microsoft development mantra, namely the SDL (Secure Development Lifecycle). Cowan described the SDL revelation this way: "All that stuff they teach you in college . . . what if we did that? Turns out it works!" As an example, in 2003 Microsoft SQL Server had a serious buffer overflow which resulted in a "flash worm." In 2004, after the SDL, and an update of SQL Server, there was just one vulnerability in three years. He compared this to MySQL, which had twelve serious vulnerabilities in the same time period.

Another example of Windows leading the way was the recompiling of Windows XP SP2 with StackGuard (something close to Cowan's heart, see "StackGuard" in Sec '98). He pointed out that Windows was the first OS to ship with the feature (2004) and now every other major OS does the same. It caused a stir in the audience when Angelos Keromytis said from the back, "OpenBSD came first, everything had SG." "When did OBSD do it?" Cowan asked. "Took time, we had to clean up all the ports. In 2003, version 3.3. So we won by a year and a half," was the response.

"After XP SP2, that's what I thought," replied Cowan unfazed, moving on to described more features of Windows XP Service Pack 2. These included a default firewall, pop-up blocker, and image blocking by default in emails, "which is optional in Thunderbird." A "boo" rose from the audience; "Well, someone should update the Wikipedia page," retorted Cowan to laughter. He also noted other email security features, including Attachment Execution Service (AES), where applications downloaded from the Internet via an email attachment are marked as such. Prompts are raised when a user clicks on the application to execute it. Perry Metzger then asked, "What happens when it gets copied?"

"Not so good," replied Cowan. "But when you copy it, you have to click." To which Sandy Clark replied, "It's kinda like it's not Microsoft's problem?"

"Then users just click to open. Prompt fatigue," sighed Cowan. "Prompts are *not* inherently evil. Prompts that users always say yes to, are a problem." He noted that this is a problem he is actively working on at Microsoft.

Moving on to some browser security features, Cowan described the Windows sandboxing features. He tipped his hat to similar features in the UNIX world, but the PMIE and MIC (Window's sandbox) is on by default. He noted clickjacking defenses: "Don't frame me bro!" Additionally, a SmartScreen blocker for phishing sites and ActiveX filtering for malware. This also caused a stir. Someone from the audience shouted, "Are you now saying ActiveX is secure?"

"No! More secure than it used to be. Security is not a Boolean."

"It's running arbitrary code from the Internet."

"It's not arbitrary, it comes with a certificate!" That received quite a bit of laughter. "If it is ActiveX," continued Cowan, "it is running inside the MIC." He continued to note that plug-ins in Firefox do not use a MIC, and that the SmartScreen has blocked 1 billion malware downloads. He showed a graph to this effect, which also caused a stir.

Cowan deflected comments left and right as he finished his talk. He discussed how Windows undercut the "run as administrator" culture via the UAC, and the number of apps requiring privileged access went from 900,000 to 180,000.

In Windows 7, Microsoft Essentials provided key antivirus protection. AppLock ensured that users weren't using out-of-date and vulnerable apps using a flexible policy. BitLocker was included for full drive encryption, and the virtual accounts feature allowed for per-application user accounts (something UNIX has had since the late '80's).

Cowan concluded by acknowledging that UNIX had a very large security lead, and this was because Microsoft wasn't really trying that hard; Windows has closed the gap across the board, but "once the gap is closed, do users really care which was first?" Finally, he noted that Windows is still the big target and where the money is. The number one security benefit of UNIX may very well be its obscurity; however, don't confuse most obscure with most secure.

DETECTION OF NETWORK ATTACKS

Summarized by Prithvi Bisht (pbisht@cs.uic.edu)

- ***BotGrep: Finding P2P Bots with Structured Graph Analysis***
  Shishir Nagaraja, Prateek Mittal, Chi-Yao Hong, Matthew Caesar, and Nikita Borisov, University of Illinois at Urbana-Champaign

Prateek Mittal presented an algorithm to find P2P botnets. He mentioned that botnet sizes are increasing and that botnets are adopting P2P networking. Such networks are often robust, as there is no central node to be found and brought down. In addition, P2P networks use structured layered topology to be robust and scalable. The current detection schemes detect misuse based on the amount of attack traffic or detect anomalies either by noting deviations from a certain threshold or by using clustering algorithms to isolate botnet traffic. The BotGrep system requires constructing a communication graph whose vertices are Internet hosts and edges represent communication between them. The goal is to extract P2P botnet structure from this graph.

The BotGrep system uses traffic monitors at different ISP sites to construct a host-level communication graph. Inputs from misuse detection systems help differentiate between benign and hostile P2P traffic. An inference algorithm then splits this graph into a botnet graph and a background Internet graph. The inference algorithm uses structural properties of P2P networks. Specifically, random walks compare the relative mixing rates of the P2P subgraph and the rest of the communication graph. The subgraph corresponding to structured P2P traffic is expected to have a faster mixing rate than the subgraph corresponding to the rest of the network traffic.

The approach consists of three main steps: (1) a pre-filtering step reduces huge communication graphs into a smaller set of candidate P2P nodes by using short random walks; (2) a key recursive graph partitioning step uses a modified SybilInfer algorithm, with the intuition that for short random walks the state probability mass is homogeneous, to eliminate non-P2P nodes; (3) a validation step uses heuristics, namely graph conductance, entropy comparison, and degree-homogeneity tests, to decide if a partition is P2P and to terminate the iterations. The scheme was tested using synthesized de Bruijn P2P graphs embedded in the Abilene communication graph. The detection rate was over 90%, and false positives were reported to be manageable. The detection rate remained above 90% for LEET-Chord graphs, but the false positive rate increased significantly. In the presence of large background graphs, performance remained unchanged, and false positives were not dependent on the size of background graphs. Mittal concluded by mentioning that graph algorithms can be used to find botnets, inter-ISP cooperation is useful for security, and stealth and robustness seem inversely proportional.

Yip Fong from MSR noted that the experiments were conducted with a single botnet and asked how the system would work if there are nodes from multiple botnets. Mittal responded that BotGrep can handle multiple overlapping communities through clustering. Fong asked if the scheme could conclude that nodes from different botnets belonged to the same botnet based on a small fraction of nodes. Mittal responded that clustering based on edges instead of vertices would take into account nodes that are part of multiple communities, but the experiments were not done on these lines. A researcher noted that there weren't many details on how the graphs were generated for botnets and what they reflected (e.g., in the case of Kademlia). Mittal responded that the experiments only considered P2P nodes of these graphs; that is, in Kademlia the node degree is logarithmic to the size of the network. The same researcher noted that it was not accurate, graphs did not resemble what the Storm botnet would generate, the generated tool was testing against a flawed model; he wondered if it would work with real bot traffic such as Storm. Mittal agreed; the only modeled component was the P2P overlay maintenance traffic. Someone from Columbia University asked about the motivation for using the Markov-based model for detection. Mittal responded that the most common feature of all topologies was that they were extremely fast-mixing and hence the random-walk-based scheme was used. Someone from the University of New Mexico noted that the false positive rate was 0% and there should be more latent botnets. Mittal said that he didn't have a good answer and that views from multiple ISPs might have identified more botnets.

- ***Fast Regular Expression Matching Using Small TCAMs for Network Intrusion Detection and Prevention Systems***
  Chad R. Meiners, Jignesh Patel, Eric Norige, Eric Torng, and Alex X. Liu, Michigan State University

Jignesh Patel presented a fast regular expression matching scheme using Ternary Content Addressable Memory (TCAM) for Intrusion Detection Systems (IDS)/Intrusion Prevention Systems (IPS). The problem was to quickly scan a packet payload to see if it matched a given regular expression (RE). Existing techniques based on software (use ASIC chips) or hardware (use FPGA) were unsuitable for fast RE updates. TCAM can have three values: 0, 1, *(don't

care), and search is conducted with the content instead of addresses as in traditional memory. The idea is to search against all entries in TCAM in parallel and return the first match. However, the key problem is that the basic implementation produces large TCAM tables. Two optimizations were presented to reduce the space bloat because of transitions. The first optimization exploits the fact that many transitions from one state have common destinations. All such common transitions are merged using the bit-weaving algorithm by Meiners et al. The second optimization reduces common transitions across states. To do that, it reassigns state IDs that are unique to avoid matching unrelated states. However, the optimization needs to retain ordering of states, which is addressed by using the D2FA algorithm by Kumar et al. Patel also mentioned that D2FA has the problem of long chains of deferments which the TCAM-based approach avoids if it finds the deferred state in a single lookup.

Further, space optimization was achieved by consolidating the TCAM tables. This optimization is based on an observation that even after the previous optimizations, some transition tables share common entries although destination states are different. The key idea is to merge multiple transition tables into one table. The two main challenges are: (1) how to merge k tables; (2) which states should be consolidated. The former is addressed by local state consolidation and the latter with the global state consolidation, which uses graph matching and dynamic programming. After minimization, Patel presented a variable-striding algorithm to improve the throughput. To avoid state space explosion, a solution based on k-var-stride DFA (deterministic finite automaton) that consumed 1–k characters was proposed. This led to a linear increase in space. Experiments were conducted with 8 regular expression sets. With transition sharing, the approach generated TCAM entries in the range 1.18 to 2.07 for each state. This was reduced to .32 to 1.17 for each state, below the fundamental limit of 1 entry per state. The highest throughput was approximately 18.58 Gbps. Patel concluded by highlighting that this is the first TCAM-based RE-matching algorithm.

Niels Provos from Google asked whether regular expressions from snort rules were used. Patel responded that 3 of the 8 RE sets in current experiments were from snort, and the work was being extended to cover the rest of them. Had they tried REs used for backtracking? The current focus was on the snort rule set, and backtracking-based REs could not be expressed by DFAs. For handling such REs, DFAs could be annotated with some counting mechanism or scratch memory.

- ***Searching the Searchers with SearchAudit***
  John P. John, University of Washington and Microsoft Research Silicon Valley; Fang Yu and Yinglian Xie, Microsoft Research Silicon Valley; Martín Abadi, Microsoft Research Silicon Valley and University of California, Santa Cruz; Arvind Krishnamurthy, University of Washington

John presented SearchAudit, a tool to leverage search engine audit logs for security analysis. Attackers can craft malicious queries to find misconfigured or vulnerable servers such as the DataLifeEngine server, which was vulnerable to Remote File Inclusion (RFI) and was found with the search term "Powered by DataLife Engine." The idea here was to audit search logs of search engines to understand attack behavior and possibly detect new attacks, and use it for case studies. SearchAudit starts with a seed set of 500 known malicious queries that were taken from underground forums. The seed set was expanded by including queries issued from same IP addresses and finding other queries by issuers of known malicious queries. As attackers use variants of malicious queries, queries in the expanded set were generalized. The generalization consisted of creating regular expressions from each query that captured the essence of the query. This process was repeated as a fixed-point computation.

John discussed validating the outcome of the queries using statistical techniques, e.g., links clicked in results. Queries found by SearchAudit showed significant differences when compared to normal queries in the search logs. John also discussed three sets of case studies. The first case study aimed at early detection of vulnerable servers by analyzing queries that search for vulnerabilities. Such detection can be confirmed by identifying vulnerable servers that subsequently appear in blacklisted domains. The findings indicated that 5% of the identified servers appeared in blacklists subsequently and 12% may be vulnerable to SQL injection. The second case study analyzed queries that search for forums to spam. Findings indicated that some IP addresses sent a huge number of queries, and these findings were consistent with HoneyNet Project findings. The last case study focused on queries for exploiting MSN messenger and found 400 common domains that generated this traffic. Behavior of compromised accounts was analyzed through IM logs and found to be deviating from the normal Messenger logins that typically originated from fewer than four different subnets. John concluded that search queries can provide early indications of attacks and help detect and prevent attacks at an early stage.

Someone asked if the regular expression generation was automated. John confirmed that it was automated and scalable. Niels Provos of Google asked how the expansion of the seed set would work in the presence of churn (people switching IP address) and computers used to make regular queries, and how that would expand to getting false positives. John responded that the expansion was on a per-day basis, assuming that the DHCP changes on less or more than a day's granularity. Further, he referred to the paper

for a technique to detect and eliminate proxies to reduce the false positives.

INVITED TALK

- ***Docile No More: The Tussle to Redefine the Internet***
  James Lewis, Senior Fellow and Program Director at the Center for Strategic and International Studies

  Summarized by Ronnie Garduño (koko@rpg-free.com)

James Lewis focused on the increasing global tension between various countries and the US and its allies over the control of the Internet. The Internet is a mostly decentralized network, but ICANN, the Internet Corporation for Assigned Names and Numbers, does make certain decisions that guide Internet development. The governments of some countries, such as Brazil, China, India, and Russia, charge that ICANN has too much power over the Internet and that it is controlled mainly by the US government. These governments have been working on their policies for controlling the Internet in their various countries for some time now, and they have come to the conclusion that the Internet as it now exists is too open and destabilizing, and that there should be more government control over it. Their attempts to make changes in line with these views are leading to the possibility of a fragmented Internet, with each country's populace able to communicate only with others within that country's fragment of the Internet.

Lewis argued that conceiving of the Internet as a "global commons" ignores the true reality of the situation, which is that the Internet exists due to physical connections and servers, each of which falls under the sovereign control of one nation or another. Lewis further argued that no other country on Earth sees the Internet as a global commons, and suggests instead the idea of a global condominium, within which the citizens of each nation dwell in a shared space with few rules. In this model, each country feels that the Internet in their country is "theirs," not the "world's," leading to the feeling that their sovereignty should also extend to the telecommunications within their borders.

One arena in which this struggle for dominance is likely to take place is in the ongoing standardization efforts. These standards are usually written with intentions to be open, simple, and flexible, goals that do not satisfy those seeking greater control over the Internet. A large part of the problem other countries have with US control over the Internet is due to a disagreement over the extent of the control the US government has over companies within its borders. Lewis disclosed that he has spoken to individuals within governments outside the US who do not believe that the official US government stance on freedom of speech in the media and the operation of companies is an accurate description of the reality of the situation. These people (and probably their respective governments) feel that it would be unrealistic to believe that these activities go on in the US without intervention, and thus they postulate that the US government has a similar level of control to that which exists in various other countries, such as China.

Some other countries claim that the US is a hegemony, a cultural, financial, political, and military force of leadership over the world. Likewise, other countries tend to feel that the US is a kind of controlling power which insists upon assimilating others into its power structures. Some of their fears are economic. There are governments that feel that global organizations like ICANN, the WTO, etc., are part of an overarching strategy to ensure economic dominance on the part of the US. These governments are very interested in the Internet as a tool for economic expansion, but they dislike its political effects and feel that their sovereign powers should extend to the Internet in a way that is currently not entirely feasible, given the Internet's current architecture and control mechanisms. Lewis shared a quote from a Chinese government worker: "Twitter is an American plot to destabilize Iran."

One major problem with a fragmented Internet is that it may not work as well as the current Internet setup. There are two factors held in tension in many countries: the government wants to connect globally for commerce, for research, and for education, but wants to disconnect from the rest of the globe when it comes to politics. The US has mostly left this issue alone, trusting in the strength of current alliances, technologies, and social and economic forces to keep the Internet the way it is. While this is the case, some people are looking to the US for guidance on the future of the Internet, a step which is slowly and quietly taking shape. Lewis argued that this is a necessary step, and says that if the US does not step up to shape the Internet's future, other countries will, and probably in a way which will displease many parties globally. This is interesting in light of a recent poll cited which suggests that the global community has decided that open access to information is a fundamental human right.

Someone asked if foreign governments view American companies developing privacy-enhancing technologies for use outside of the US as part of a coordinated US effort to subvert their control. Lewis pointed out that Hillary Clinton's speech commenting on the Google-China censorship issue implied that we are supporting Google, tarring their reputation even if they are not directly government-supported. Foreign governments don't need total control and don't mind a few people evading their technologies control, but don't like the idea of everyone being able to do it. They also don't understand that American companies are not always working in direct cooperation with the government, since that is not the case in most other countries in the world.

Someone else asked how foreign governments view things like Wikileaks, which don't seem to be controlled by the US government. These other governments often view things like this as proof that the US government is no longer competent. To them, it is evidence of the failure of independent freedoms to ensure social stability, and it reinforces their

need for political control over the flow of information and the Internet.

Another person wondered about countries that are allies of the US, such as Australia and the UK, embracing Internet censorship; how will the US make the case to countries like China to embrace the open flow of information? Lewis replied that he hadn't heard a thing about that yet, although he has been waiting for it. Australia's problem is that they were open about what they were doing. They would probably have done fine if they had kept it a secret, although that may be unrealistic in their case. Many Western European countries are looking at similar implementations of censorship technology, and some of them have already set such systems up, facilitated by the close relationships between the governments of these countries and the telecommunications companies in them. This means that we have already faced accusations of hypocrisy, along the lines of "If you guys can do it, why can't we?" on the issue of censorship. The usual response to these allegations is that the Western world does not generally engage in political censorship or industrial espionage, unlike some of the other countries involved, like Russia. That would be the case the Western world would have to present, that censorship for some limited purposes is acceptable, but not broad-scale political censorship. The technologies to control the Internet have been developed, and countries are realizing that they can extend their control into the domain of the Internet in their boundaries. It is a complex issue because many of these technologies can be used to secure networks, but many people worry that they will be used for censorship. This may mean that we are at a disadvantage as far as cybersecurity goes, since we cannot reasonably implement such technologies.

Someone asked about the future of Net neutrality in the United States. Is the government going to step in and establish rules, or are the corporations going to set up their own? Lewis said he sees the general trend as being that governments are taking a larger role in the control of the Internet. In many countries, this may not be desirable, but it is the case. In the United States, the situation is more complicated because of conflicting interests in Congress. These interests are about evenly matched, so in the end there may be no action taken at all. The telecommunications corporations are insisting that they need to set up their own rules to recoup the expenses of setting of their networks. They also complain that many new Internet technologies and applications are expanding use to the point of straining the networks. A specific example is that of AT&T's 3G network; this network is under constant strain by AT&T's exclusive iPhone and iPad users, many of whom are streaming a great deal. The balance, then, is between return on investment and openness of information flow. Many countries may come to their decisions on this process more quickly than the US, but the messy American political process may prove to be a boon in this case, by allowing enough time for the debate to be fully engaged in by the country at large.

Finally, someone asked about the situation in the United Arab Emirates, in which they recently banned the use of BlackBerry smartphones due to concerns over encryption. Lewis said he knew exactly what their concerns are, in this case, because the US faced those issues more than a decade ago. For a while, the US policy was to restrict the export of encryption technologies, to enable the monitoring of communications from overseas for security and law enforcement reasons. The problem was that these restrictions were unenforceable given the Internet, and many were getting around them by downloading freeware. Some of that freeware was even secretly sponsored by other countries and had back doors, allowing the governments of those countries access to the encrypted communications sent by users of that software. Another problem was an economic concern: encrypted communications are vital to e-commerce, and without the ability to freely use encryption technologies, American companies would have problems expanding their businesses overseas. In the face of all these problems, the US finally relented and removed their restrictions on the export of encryption technologies. The real question, then, is: how long can the UAE keep up their current policy of encryption control, in the face of these security and economic factors? Even though they are a small, oil-rich country, they can't do so for long, given the difficulties involved.

RUMP SESSION

Summarized by Cody Cutler (ccutler@cs.utah.edu)

- ***A Methodology for Empirical Analysis of Permission-Based Security Models***
  David Barrera, Carleton University

Proposing a new methodology for analyzing how permission-based systems are used in practice, David Barrera et al. designed and implemented an algorithm that takes applications as input. It then determines which permissions they require and generates 2D "unique fingerprints" describing this particular application.

- ***Revisiting the Computation Practicality of Private Information Retrieval***
  Femi Olumofin and Ian Goldberg, University of Waterloo

Femi Olumofin and Ian Goldberg question the results of previous work concerning Private Information Retrieval (PIR): "No conclusion is as efficient as the trivial PIR scheme" in practice for multi-server PIR schemes. The hunch paid off: they discovered that the response times of other schemes are one to three orders of magnitude smaller than the trivial scheme, assuming that realistic computational power and network bandwidths are available.

- ***Somnolescent Cryptanalysis***
  Aniket Kate, University of Waterloo

Aniket Kate blazed a new trail in the security world and pioneered what will no doubt be the most effective brute force technique: brute forcing with Cobb from the movie *Inception*. All that is needed is to descend into the dream-world five or six layers deep, where a few real-life minutes will turn into hundreds of millions of years—plenty of time to brute force the target encryption key. Work is currently being hindered by the search to find Cobb himself, which so far has been unsuccessful.

- ***Security on Memory Deduplication***
  Kuniyasu Suzaki, AIST, Japan

Virtual machine monitors can share identical memory pages between virtual machines, just as an operating system shares identical pages between processes. Kuniyasu Suzaki pointed out that memory peeking can infer information about processes running on other VMs on the same physical system. It can be observed that another VM (but it is unknown *which* VM) shares a page by carefully writing to certain pages and watching for timing latencies which signify the virtual machine monitor had to perform a page copy because of copy-on-write.

- ***RFID-Based Electronic Voting: What Could Possibly Go Wrong?***
  Yossi Orren

Although election procedures used to count votes in Israel were perhaps old-fashioned, they produced excellent results. With participation well above 90% and disqualified votes less than 8%, it is a wonder why it was decided to change to an electronic system. Yossi Orren demonstrated several attacks against the new electronic system, ranging from "unsophisticated" to "slightly sophisticated" where he was able to completely erase all previous votes (thus disqualifying the region) and to change the votes for the already cast ballots to an arbitrary candidate.

- ***Dispatch Loops as Execution Signatures***
  Nathan Taylor, University of British Columbia

Nathan Taylor developed a tool that would watch binary execution, find its main loop, and summarize what changes occur in its address space. His tests on fairly straightforward programs turned out well, and he is now interested in using it on sub-programs and slightly trickier executables. Perhaps someday the tool will be able to analyze malware.

- ***What Is the Name of My Cat?***
  Bart Preneel, Katholieke Universiteit Leuven

Bart Corneal studied secret questions used for password recovery techniques. His data, collected when he was asked to vet questions for an online project, shows that 12% of security questions are clever, 54% are simple with low entropy, and 6% are very strange. He asked that everyone help him in this experiment: you can contribute by sending him your security question and answer so he can continue this interesting research.

- ***Relay Attacks on Passive Keyless Entry and Start Systems in Modern Cars***
  Srdjan Capkun, ETH Zurich

Srdjan Capkun's new car came with a great new feature: using RFID, the car would unlock automatically if you stood within range for a few moments. He demonstrated, though, that if you don't keep your new key in a special, cool-looking aluminum bag you are vulnerable to a relay attack where someone can open your car even if you are a long distance away from it (http://eprint.iacr.org/2010/332.pdf).

- ***Got Privacy?***
  Maritza Johnson, Columbia University

Maritza Johnson is doing a study concerning privacy: are Facebook users able to configure their privacy controls in a way that actually reflects their intents? You can help her by visiting: http://apps.facebook.com/gotprivacy/.

- ***The Case for Open Source Software***
  Jose Fernandez, Polytechnique Montréal

Jose Fernandez delivered a truly beautiful metaphor relating open source to the monks who tried desperately to integrate strawberries into their foreign lands. However, the birds (evil vendors) made it very difficult for the poor monks. The birds like small strawberries, and cast their seeds far and wide. The monks, like most people, prefer larger, sweeter strawberries, but with the birds "'dropping"' seeds everywhere, the monks needed a special environment to breed their larger strawberries.

- ***NoTamper: Automatic Blackbox Detection of Parameter Tampering Opportunities in Web Applications***
  Prithvi Bisht, University of Illinois, Chicago

Prithvi Bisht demonstrated attacks against some online shopping centers concerning client-side verification; he was able to get the shopping cart to pay for itself by having negative quantities of some of the items! You can read more at http://www.cs.uic.edu/~pbisht.

- ***Simple IPSec***
  Steve Bellovin, Columbia University

Because "95% of options are completely irrelevant to 95% of all users" in IPSec configuration files, Dr. Bellovin developed Simple VPN. It makes the right choices for you automatically—his configuration file is 11 lines. You can grab this tool at http://sourceforge.net/projects/simple-vpn.

- ***The Human-Centered Authentication Attack***
  David Harmon, Columbia University

David Harmon led a study where they explored just how safe our passwords in our heads really are. They found that 1 of 10 users will divulge their password if asked nicely while 8 out of 10 will reveal it if they are waterboarded or encouraged in a similar way.

- ***Secure Systems Cannot Be Engineered***
  Anil Somayaji, Carleton University

For a system to be truly secure, it must have a Roman guard. Unfortunately for us, that requires simply way too much infrastructure for the Internet and all our systems. Roman guards do not scale to the Internet.

- ***Pac-Man on the Sequoia AVC-Edge Voting Machine***
  Alex Halderman, University of Michigan

Alex Halderman and colleagues prove that our faith in electronic voting machines is well justified—they installed FreeDOS on a Sequoia AVC-Edge for the purpose of playing Pac-Man (in celebration of its 30th anniversary) in a matter of hours. Not only did they have complete access to the machine's internals after opening it (without ruining the seal), they were also able to correctly guess which pins on the motherboard to jump in order to defeat the 30-second watchguard timer.

- ***The Word***
  Dan Wallach, Rice University

Dan Wallach closed the rump session with a takeoff on Stephen Colbert's "The Word." He reminds us that we shouldn't be too concerned about elections. Making a secure and usable voting system that preserves privacy is indeed quite hard, so let us not even worry about it—everything will be taken care of for us by hard-working politicians who are genuinely concerned for our welfare. The best policy is "'out of sight, out of mind.'"

DISSECTING BUGS

Summarized by Manuel Egele (megele@cs.ucsb.edu)

- ***Toward Automated Detection of Logic Vulnerabilities in Web Applications***
  Viktoria Felmetsger, Ludovico Cavedon, Christopher Kruegel, and Giovanni Vigna, University of California, Santa Barbara

Felmetsger started her presentation by saying that Web applications are omnipresent and come in many different forms. Common vulnerabilities, such as missing input validation or cross-site scripting, can be detected with taint analysis. In contrast, application-specific vulnerabilities are more difficult to detect, and so far have experienced little attention from the research community.

The prevalence of such vulnerabilities is hard to estimate because there is no specification of what constitutes a "'logic vulnerability.'" Information about such vulnerabilities is scattered across different categories, if they are reported at all. Felmetsger showed four examples of application-specific vulnerabilities in Twitter, Facebook (2), and one in myphile that became public only the day before the presentation. Subsequently, Felmetsger presented Waler, a fully automatic approach based on dynamic analysis and model checking to find logic vulnerabilities in servlet-based Web applications. Waler derives an approximation of a program specification by exercising the Web application with "normal" input. Daikon is used to generate likely invariants on the recorded program execution paths. However, many likely invariants found that way do not represent real invariants. Therefore, to assess the validity of an invariant, Waler employs model checking over symbolic input based on Java pathfinder.

Waler is able to detect two different kinds of vulnerabilities: missed checks on a program path, where an invariant supports a check but a different path leading to the same state does not perform such a check, and inconsistencies between state or session variables and database fields. Since Waler has to take all possible entry points to the Web application into account, the accumulated state becomes too big to handle. Therefore, the authors included different techniques that allow them to detect and prune equivalent states. The authors evaluated Waler on four real-world applications and eight applications that were created by software engineering students as lab assignments. The presentation then elaborated on one of the found vulnerabilities, where a missing check led to unauthorized access with administrative privileges in one of the real-world applications.

In her remarks on future work, Felmetsger mentioned recent progress in the development of Waler, such as support for the Struts framework, and their plans for experimenting with new heuristics to further reduce the state space.

The first question was geared at finding out how Waler finds all possible entry points to a Web applications. According to Felmetsger, all the entry points can be extracted from the configuration file. Eric Eide (University of Utah) was wondering whether the same approach could be applied to application domains other than Web applications, and whether the used heuristics would have to be adapted. Felmetsger agreed that some of the heuristics are Web-application specific, whereas others, such as the heuristics that check for a supporting check on a program path, should be applicable also to stand-alone applications. Eide continued by arguing that 300,000 states are not that many for a model-checking approach, and wondered whether Waler is simply keeping too much state. To which Felmetsger replied that the limiting factor was time, as a simple application consisting of only hundreds of lines of code took a very long time to analyze, and that the effort to analyze a big application (e.g., based on the Struts framework) is orders of magnitude higher.

- ***Baaz: A System for Detecting Access Control Misconfigurations***
  Tathagata Das, Ranjita Bhagwan, and Prasad Naldurg, Microsoft Research India

Das presented Baaz as a solution for detecting access control misconfigurations. He stated three properties that such a system has to provide. It should be preventive, should not require a formal specification or documentation of the access control policy, and should provide high performance. Baaz is built around these design goals as an auditing tool to find potential misconfigurations by checking for inconsistent policies. The system provides the desired perfor-

mance by employing scalable algorithms. Das then continued by elaborating on the two classes of misconfigurations Baaz can detect. Security misconfigurations indicate that a user has access to a resource to which she should not have access. Accessibility misconfigurations signify that a user does not have access to a resource to which she should have access.

Baaz does not require a documented security policy. Instead, it relies on a reference data set that needs to be specified as a binary matrix. This reference data set is used as a proxy for a missing security policy. Baaz checks, for each subject, whether it can detect inconsistencies with the reference data set. Das then presented an example for the matrix reduction in Baaz where the basic assumption is that members in a group should have access to the same resources. The presentation then elaborated on a misconfiguration that was detected by Baaz.

The evaluation of Baaz was performed on three different data sets. The presentation then covered the results of the file server data set in detail, where 18 misconfigurations were detected. To assess the ground truth, the authors spent two days manually examining the data set. Das said that the most time-consuming step was the matrix reduction step, whose runtime grows linearly with the size of the matrix.

Someone raised the concern that the access is controlled by a resource owner and asked whether Baaz required input from the owners, such as having synchronized user names. Das replied that Baaz can be run across administrative domains, assuming that the binary matrix is already available. It is not the intent of Baaz to create this matrix.

- ***Cling: A Memory Allocator to Mitigate Dangling Pointers***
  Periklis Akritidis, Niometrics, Singapore, and University of Cambridge, UK

Dangling pointer vulnerabilities, such as use after free(), are just as dangerous as buffer overflows. Thus they developed Cling as a drop-in replacement for malloc() and new. Similarly to existing approaches, Cling trades memory space for security. Akritidis then presented an example of a vulnerability that Cling is designed to prevent, and two alternatives to prevent such vulnerabilities. The naive solution is never to free any memory, whereas Cling takes the more sophisticated approach, pooling memory and only reusing it for objects of the same type. Cling considers two objects to be the same type if they got allocated by the same instruction (i.e., the address of the call to malloc).

They faced some challenges in implementing Cling. Cling needs to unwind the stack for functions that wrap malloc calls and create different types of objects. Furthermore, as Cling is designed as a drop-in replacement for malloc and new; it does not work out of the box with custom memory allocators. The presentation also included some remarks about limitations of the proposed approach, such as stack-allocated objects and the limited address space on 32-bit architectures. Cling was evaluated on 18 benchmarks by preloading the modified allocator via LD_PRELOAD. The evaluation was performed with regard to memory size and execution time. Furthermore, Cling was evaluated with Firefox, where the requested memory size is almost the same as with the regular system allocator, and only the virtual memory size slightly increased compared to the unmodified version.

Weidong Cui (Microsoft Research) mentioned that memory reusing schemes other than the naive are possible as well (e.g., round robin). Akritidis acknowledged that other schemes exist, but everything that made use of virtual memory so far incurred significant overhead (up to 700%), due to system calls. Furthermore, he said that Cling is designed for production systems, thus protecting the applications, and not for detection purposes, as in other approaches. Robert Watson (University of Cambridge) asked how Cling would behave with C runtime allocations, and closures in particular, where the C library allocates the memory. Akritidis elaborated that the strdup function in the C library exposes exactly this behavior, and Cling treats this function as a wrapper function and is able to resolve the issue by unwinding the stack.

INVITED TALK

- ***Staying Safe on the Web Yesterday, Today, and Tomorrow***
  Sid Stamm, Security & Privacy Nut at Mozilla

  Summarized by Tamara Denning (tdenning@cs.washington.edu)

Sid Stamm began by summarizing the original security tools used by Mozilla: community bug reporting and JavaScript fuzzing. He then described some of the security strategies currently used by Mozilla, as well as strategies that are underway or being considered. Current methods include offering bounties for reporting major security vulnerabilities and fuzz-testing more aspects of the browser. In general, Mozilla is focused on building the browser as a protective agent for the user. Current and contemplated security features include wrapping different browser components, putting plug-ins in their own processes, letting sites specify normal behavior, building in defenses against some CSS attacks, and improving UI indicators of security and trust.

In the future, Mozilla expects a larger attack surface, as the browser harnesses more of a computer's capabilities. The Jetpack system is intended to facilitate add-on security by providing a more compartmentalized system of privileges and APIs. Other goals of interest include a multi-process architecture, an account manager to make a more cohesive registration and login system across sites, better security and trust visualization, something along the lines of a reputation system that reports on the privacy practices of Web sites, reducing browser entropy to increase user anonymity, and associating cookie identity with both its source domain and the page of the domain in which it is displayed.

Many of the audience questions related to either a more effective UI for conveying security and privacy informa-

tion, reputation systems for scoring Web sites, or content security policies. One question addressed Mozilla's stance on the possibility of law enforcement interest in their sync functionality. Stamm replied that the client-side encryption is meant to avoid this kind of scenario. An audience member asked about the security risks associated with a browser account manager. While there are risks, they are outweighed by the security benefits offered to users. How would a Mozilla privacy evaluation system succeed where others (such as P3P) did not gain traction? A reputation system would take the necessary workload away from sites. Stamm also said that the private browsing mode needs to be reevaluated and modified with relevant users and use case scenarios in mind. Tiered sandboxing and a simplified, backward-compatible browser experience are two opportunities for site-side and user-side content security policy.

CRYPTOGRAPHY

Summarized by Ben Ransford (ransford@cs.umass.edu)

- ***ZKPDL: A Language-Based System for Efficient Zero-Knowledge Proofs and Electronic Cash***
  Sarah Meiklejohn, University of California, San Diego; C. Chris Erway and Alptekin Küpçü, Brown University; Theodora Hinkle, University of Wisconsin—Madison; Anna Lysyanskaya, Brown University

Sarah Meiklejohn presented the paper on ZKPDL, a new programming language for the design and implementation of zero-knowledge (ZK) cryptographic protocols. The authors observed an "abyss" between the designers and the implementers of cryptographic protocols: theorists have trouble implementing their schemes at all, and programmers have difficulty implementing these schemes correctly, efficiently, and flexibly. ZKPDL is designed to serve as a lingua franca for both groups, offering theorists a way to express ZK schemes in a concise, familiar way and offering implementers a simple mechanism for incorporating these schemes into their applications.

The authors' cryptography group at Brown had struggled for several months to build an e-cash library for use in a P2P application. The result was messy and difficult to modify, which made changing the details of the underlying ZK scheme difficult. They realized that the ZK scheme could be cleanly separated from their application and reasoned that cryptographers could, if presented with the right interface, code ZK schemes themselves. Their system, ZKPDL, presents both ZK parties—a prover and a verifier—with an identical interpreter and a plaintext ZKPDL program. The prover's goal is to prove knowledge of some fact without revealing anything new about the fact, and the verifier's job is to check the prover's proof. Each party's interpreter loads the program, performs some optimizations where possible, and executes the part of the program corresponding to its role. Meiklejohn showed the interface between ZKPDL and a greatly simplified e-cash library. She demonstrated that ZKPDL programs map cleanly onto the descriptions that cryptographers write in theoretical papers. To demonstrate the efficiency of ZKPDL, she presented performance figures showing various ZK proofs with and without a caching optimization. The authors have made ZKPDL and their e-cash library available at http://github.com/brownie/cashlib.

Peter Neumann asked whether the authors' e-cash library enabled transactions to be traced when it was necessary to do so. Meiklejohn said that e-cash has a basic property that allows cheaters to be de-anonymized. Jeremy Clark asked whether the authors had implemented elliptic-curve primitives; Meiklejohn answered that they had not. Bart Preneel asked whether ZKPDL is designed to be resistant to timing attacks; Meiklejohn answered that it was not.

- ***P4P: Practical Large-Scale Privacy-Preserving Distributed Computation Robust against Malicious Users***
  Yitao Duan, NetEase Youdao, Beijing, China; John Canny, University of California, Berkeley; Justin Zhan, National Center for the Protection of Financial Infrastructure, South Dakota, USA

Yitao Duan introduced P4P, a framework for privacy-preserving distributed computation that supports data-mining operations by decomposing them into vector additions over small fields. P4P aims to address scalability problems that have troubled previous distributed-computation systems; Duan cited several such systems and claimed that their genericity hobbled their scalability. In particular, Duan remarked that existing systems place computationally intensive public-key operations at essential junctures such as simple arithmetic operations, harming performance. P4P's alternative approach is to support only computations that can be decomposed into sequences of so-called private vector additions. This class of computations includes singular value decomposition (SVD), principal component analysis, and a variety of other statistical tools. Duan claimed a run-time improvement of several orders of magnitude for these problems and said that P4P supports operations on up to millions of users or data items.

P4P focuses on a well-known problem: coaxing a group of participants into computing an aggregate function without revealing any node's inputs to any other node. In P4P, participants execute vector additions over small (32- or 64-bit) fields, a fast operation on modern architectures. Duan called these operations private vector additions and said that they are based on verifiable secret sharing. Duan presented an SVD problem as an example: each participant "owns" a row of a matrix A, and the desired computation is the SVD of A. Duan interfaced the popular ARPACK eigensolver with P4P's private vector addition. Each participant's share of the computation is a short sequence of matrix multiplications. To verify the correctness of participants' computations, the coordinating server asks each participant for a projection of its private vectors onto a server-provided random vector; the participant makes a homomorphic commitment to the projection and provides a zero-knowledge proof that the projection is correct. Duan showed run-time figures for a

variety of operations in P4P over a range of numbers of participants. Comparing P4P again to previous systems, Duan pointed out significant (many orders of magnitude) performance improvements for arithmetic operations and statistical computations. The source code for P4P is available at the P4P homepage at http://bid.berkeley.edu/projects/p4p/.

Aniket Kate asked what protections P4P provides against active attacks on the system by cloud servers running the computations. Duan acknowledged that P4P would lose efficiency in the face of such an attack, but he noted that cloud computing providers have no incentive to disrupt computations themselves and typically offer protection against various types of attacks from outsiders.

- ***SEPIA: Privacy-Preserving Aggregation of Multi-Domain Network Events and Statistics***
  Martin Burkhart, Mario Strasser, Dilip Many, and Xenofontas Dimitropoulos, ETH Zurich, Switzerland

Martin Burkhart introduced SEPIA, a system that allows network operators to coordinate defenses against distributed cyberattacks without revealing to each other the identities of their customers or the structure of their networks. Burkhart observed that network providers' dislike of detailed data-sharing has stymied past attempts to address global, coordinated attacks. In SEPIA, each participating network deploys a dedicated SEPIA peer that participates in privacy-preserving computations with other SEPIA peers on other networks. In SEPIA's adversarial model, each network's input data is confidential as long as a majority of these peers are honest. Burkhart echoed Yitao Duan's point that secure multi-party computation frameworks have suffered from speed and scalability problems. SEPIA uses Shamir's secret-sharing scheme but optimizes several important primitives for speed and composes these primitives into protocols designed specifically for aggregation of network statistics and distributed event correlation.

Burkhart pointed out that under a naive implementation of Shamir's secret-sharing scheme, a simple privacy-preserving comparison of two 32-bit IP addresses requires 2592 (81 times 32) distributed multiplications, and each multiplication requires a synchronization round comprising m^2 messages, where m is the number of participants. The authors' novel protocols use parallelization to reduce the number of synchronization rounds; they also apply Fermat's little theorem and use square-and-multiply operations to reduce the number of multiplications for an IP address comparison from 2592 to 34. Burkhart showed an example of distributed anomaly detection in which networks using SEPIA would have received early warning of a Skype outage and assessed privately how much their networks were affected compared to other networks. He suggested some optimizations as future work that would further improve SEPIA's performance. Burkhart finished his talk by claiming that SEPIA makes secure multi-party computation practical for networking applications. SEPIA's Web page is http://www.sepia.ee.ethz.ch/.

In a brief question-and-answer period, Aniket Kate referred to a technique that could reduce exponentiation to two multiplications, and Burkhart thanked him.

INVITED TALK

- ***The Evolution of the Flash Security Model***
  Peleus Uhley, Senior Security Researcher, Adobe

  Summarized by Thomas Moyer (tmmoyer@cse.psu.edu)

Peleus Uhley provided an overview of the Security Product Lifecycle (SPLC) that Adobe uses to develop security for all of their various software platforms, including the near ubiquitous Flash plug-in for browsers. He provided insight into the way in which the Flash security model differs from a stand-alone product, and provided attendees with information regarding Adobe's collaboration with various communities.

Uhley began with a discussion of why it was hard to clearly identify the security model of Flash. Specifically, phrases like "Web browser security model" and the same-origin policy are not clearly and explicitly defined, leading to various interpretations of each. He argued that this has led the Adobe Flash developers to support security features within each browser as each browser develops and evolves. He stressed that Flash, much like other plug-ins, exists within a complex ecosystem. He gave several examples of how this ecosystem has changed over the years, one example being the support of private browsing modes. As each browser added support for private browsing, so did the Flash plug-in.

Next, Uhley talked about the difficulties developers face. Chief among these is the evolution of the users. No longer can developers assume that their users have college degrees. Uhley stated that these problems are not unique to Adobe, but that Adobe Flash has become an increasingly popular target, due to the near-universal deployment of Flash. Adobe has faced several problems with regard to this popularity. Uhley highlighted that even a small percentage of successful attacks on Flash can lead to a large number of exploits, meaning attackers are shifting their focus to Flash, and often these attackers are working in larger and larger groups.

Uhley next described how Adobe's responses have evolved over time, as the attack model has changed, but also as the developers gain more insight into how Flash is being used and what users want to accomplish with Flash. The example Uhley provided described the Flash cross-domain communication policy. Initially, this was introduced to handle cases where Flash content developers assumed that two sub-domains sharing a common suffix (e.g., media.example.org and www.example.org) should be able to communicate, even when the browser treated these as different origins. Uhley indicated that reactions to this policy introduction were mixed, leading to refinements in how the policy was

handled. Uhley then discussed several other enhancements to Flash, such as auto update support, and the addition of socket policies.

Uhley finished his talk with a discussion of where Adobe is headed. He mentioned several projects where Adobe has worked with other industry partners and academia, including WEPAWT and Blitzableiter. Uhley also mentioned the Open Screen project and Adobe's efforts to port Flash to new environments, including mobile devices and televisions. Finally, he described work with OWASP and the publishing of specifications for Adobe file formats.

Several questions were raised at the end of the talk, dealing specifically with Flash security. Perry Metzger complained the Uhley had not described a Flash security model, and after some discussion, agreed to continue offline. Adam Drew of Qualcomm asked about the Flash settings manager, specifically highlighting the fact that the interface was dated and difficult to access. Uhley said that as HTML5 evolves, Adobe will be monitoring how local storage settings are handled and adapt Flash's policies to align with other local storage policies. Dan Boneh asked about several recently reported vulnerabilities, including JIT spraying, and wondered how Adobe was dealing with these issues. Uhley responded that Adobe was currently examining several potential solutions and that he could not discuss any one solution in detail. Finally, several attendees asked about communication between the browser and plug-ins. The first highlighted that it would be helpful for Adobe to provide hooks for introspection in the actual plug-in. Uhley responded that he did not have an answer to such a request at the time. Finally, Helen Wang of MSR asked Uhley about unifying the security policies of all plug-ins and allowing the browser to make security decisions on a global scale, instead of each plug-in implementing its own security policy independently of other plug-ins.

INTERNET SECURITY

Summarized by Zhiqiang Lin (zlin@cs.purdue.edu)

- ***Dude, Where's That IP? Circumventing Measurement-based IP Geolocation***
  Phillipa Gill and Yashar Ganjali, University of Toronto; Bernard Wong, Cornell University; David Lie, University of Toronto

Phillipa Gill began her talk by noting the applications that benefit from geolocating clients, such as online advertising, search engines, and fraud detection. However, geolocated targets have incentive to lie, and current geolocation approaches are susceptible to malicious targets. Then she gave an overview of their contributions. They considered measurement-based geolocation in the presence of an adversary who tries to subvert the techniques into returning a forged result. To this end, they developed two models of adversarial geolocation targets: the first, simple one is the Web client being geolocated, and the second, sophisticated one is the cloud provider, which aims to mislead the geolocation algorithm. They developed two specific attacks based on the two adversary models, and evaluated them on delay and topology-based geolocation.

Next, Gill provided background on geolocation and described two major approaches: a database-based passive approach, which is coarse-grained and slow to update, and a measurement-based geolocation, which leverages several landmark machines with known locations to probe and constrain the geolocation. Gill also showed a delay-based geolocation example to illustrate how measurement-based geolocation works, and then introduced their simple adversary model. In this model, the adversary knows the geolocation algorithm and is able to delay their response to probes. In their sophisticated adversary model, the adversary controls the network the target is located in and constructs network paths to mislead topology-aware geolocation.

Interestingly, in their evaluation, they found that against delay-based techniques the adversary has a clear trade-off between the accuracy and the detectability of an attack. Against the topology-aware techniques, in contrast, they found that more sophisticated topology-aware techniques actually fare worse against an adversary, because these techniques give the adversary more inputs to manipulate through their use of topology and delay information. In their future work, Gill described their eventual goal to develop a provable and practical framework for secure geolocation. One approach is to leverage the existence of a desired location, requiring the target to prove they are in the correct location.

No one asked questions; the session chair, Steve Bellovin, joked that sometimes he does not want to be located.

- ***Idle Port Scanning and Non-interference Analysis of Network Protocol Stacks Using Model Checking***
  Roya Ensafi, Jong Chun Park, Deepak Kapur, and Jedidiah R. Crandall, University of New Mexico

Roya Ensafi began her talk by introducing a peach attack, in which the attacker only climbs the hills (with significant cost) to grab delicious peaches when the peaches in a peach orchard the attacker can see stop disappearing. Similarly, in the port scanning attack, the attacker can also leverage the information from a zombie to infer the status of a victim. That is, the attacker uses side-channel attacks to bounce scans off of a "zombie" host to stealthily scan a victim IP address or infer an IP-based trust relationships between the zombie and victim.

After providing some background, Ensafi presented their techniques. They built a transition system model of a network protocol stack for an attacker, victim, and zombie, and they used model checking to test their model for non-interference properties. Their transition-based network stack model consists of five different hosts (two zombies, two victims, and one attacker). Each host has an IP, three different ports and their status, a TCP RST counter, and SYN cache. The rules in the transition system are (1) the attacker can send any arbitrary sequence of packets; (2) the attacker cannot send packets to the victim with its own return IP;

(3) the attacker never replies to any packet; and (4) zombies and victims reply to packets based on typical Linux or FreeBSD network stack rules. They used SAL as their model checker. Two new methods of idle scans resulted from their modeling effort, based on TCP RST rate limiting and SYN caches, respectively. Their empirical experimental results show that it is possible to scan victims which the attacker is not able to route packets to, meaning that protected networks or ports closed by firewall rules can be scanned. This is not possible with the one currently known method of idle scan in the literature that is based on non-random IPIDs. Through modeling two resources, RST rate limitations and a split SYN cache structure, they also tried to capture the distinction between trusted and untrusted hosts, which will appear in the future design of network protocols. Non-interference for these two resources was verified with symbolic model checking and bounded model checking. They showed that in practice it is possible to infer trust relationships and other IP routing rules between the victim and the zombie.

Someone pointed out that according to the RFC protocol in her second example, when sending SYN-ACK to open port, you will get an RST, because SYN-ACK is out of sequence. Ensafi said their result is from the experiment they did on FreeBSD and Linux, and the OS implementation may not exactly reflect the RFC protocol. Someone else asked why the authors didn't consider Windows and Mac. Ensafi replied that she is a fan of Linux and FreeBSD. A third person speculated whether it is possible to use a VM which has great checkpointing, and inside the VM run a real OS rather than creating the transition and performing model checking. Ensafi answered that the idea is great but may face some new problems, such as security during the transition modeling. Robert Watson asked for recommendations on techniques for isolating different bits in the same cache from each other to further differentiate trusted and untrusted machines. Ensafi replied that it was interesting to think about this. Shawn Hernan asked for thoughts on how an attacker in the real world would locate a suitable zombie. Ensafi answered that she is not clear on how the attacker finds the zombies, but an attacker should have the knowledge to locate them.

- ***Building a Dynamic Reputation System for DNS***
  Manos Antonakakis, Roberto Perdisci, David Dagon, Wenke Lee, and Nick Feamster, Georgia Institute of Technology

Manos Antonakakis presented Notos, a dynamic reputation system for the Domain Name System (DNS). DNS is an essential protocol used by both legitimate Internet applications and cyber attacks. But the problem so far is: (1) malware families utilize a large number of domains for discovering the *up-to-date* C&C address; (2) IP-based blocking technologies have well-known limitations and are very hard to maintain; (3) DNS blacklisting-based technologies cannot keep up with the volume of new domain names used by botnets; and (4) detecting agile botnets cannot be achieved by the current state-of-the-art detection mechanisms. Thus, the authors designed Notos, a dynamic, comprehensive reputation system for DNS.

After briefly describing related work such as passive DNS, IP reputation and blacklisting, and DNS reputation and blacklisting, Antonakakis introduced their techniques. Basically, their techniques use passive DNS query data, and extract from network-based, zone-based, and evidence-based feature vectors. It involves a network modeling step along with two clustering steps: one—coarse-grained—uses the network; the other—fine-grained—uses the zone feature vectors. As such, they are able to characterize unknown domains with known network behaviors (for example, CDN, dynamic DNS, or just popular domains) but also with clusters based upon already labeled domains in close proximity. Their reputation function uses the product of both supervised and unsupervised learning steps to compute a reputation score for a new domain indicative of whether the domain is malicious or legitimate. In their evaluation, they applied Notos in a large ISP's network with DNS traffic from 1.4 million users. Their results show that Notos can identify malicious domains with high accuracy (true positive rate of 96.8%) and low false positive rate (0.38%), and can identify these domains weeks or even months before they appear in public blacklists. Their future work includes targeted detection and combines Notos with spam detection systems for improving accuracy as a primary coarse filter.

Reiner Sailer (IBM) asked about the possibility of an adversary taking advantage of their learning technique. Antonakakis answered that it is impossible, as an adversary cannot evade passive DNS. David Reed was concerned about how the authors label the data, since a classic learning algorithm requires reliable labeling. Antonakakis replied their technique is based on public and private blacklists, which should be trusted. Shawn Hernan first asked about the confidence that Alexa lists are in fact legitimate whitelists from non-malicious domains. Antonakakis answered that the top 500 are definitely true. Hernan's second question concerned whether their heuristics that many domain names pointed to a single IP indicates maliciousness works in an IPv6 world. Antonakakis replied their technique works in IPv6, but they may need to revisit their heuristics. Lucas Ballard asked for thoughts on the case of bad guys affecting domains they do not control. Antonakakis answered that an attacker has to compromise Web servers to achieve their goals.

INVITED TALK

- ***Understanding Scam Victims: Seven Principles for Systems Security***
  Frank Stajano, Senior Lecturer at the University of Cambridge, UK

  Summarized by Thomas Moyer (tmmoyer@cse.psu.edu)

In this talk, Frank Stajano presented work that he has been doing with Paul Wilson, a magician on a television show called "The Real Hustle." In this work, Stajano examines scams that Wilson performs on unsuspecting people. After

the scam is complete, Wilson and his team explain the scam in detail and how human nature has allowed the victim to be scammed. Stajano examines these scams, and in particular the victims, and categorizes the principles used to scam the victim. In outlining these principles he provides insight into how system security engineers can take into account human nature when designing security mechanisms.

Throughout the talk Stajano presented video clips of scams presented in the technical report. After each, he discussed how the principles could be applied to system security. The first scam examined was called the three shells game, where the victim must follow a pea hidden under one of three shells while the con artist moves the shells around. The principles behind this scam include distraction and herd mentality. The distraction principle states that a focused user is distracted from other important things, leaving them vulnerable to attack. The herd principle states that a victim will let their guard down when others around them appear to share the same risks. This applies to multi-user systems, where users are more willing to take risks since there are other users willing to accept the same risks.

The next scam, the lottery scam, illustrates the dishonesty principle, the need and greed principle, and the time principle. In the scam, the victim is asked to buy something for less than face value, and is later told that the value of the object is significantly more than originally thought. In the clip, a lottery ticket is forged that appears to be worth several thousand pounds, but later appears to be worth even more than that. The victim is tricked into giving the con artist money for the ticket, at the original value, thinking that he can cash the ticket in and make a larger profit. The dishonesty principle states that the victim's larceny hooks them into the scam, after which the con artist will use this against them to achieve their goal. The same can be said for placing the victim in an embarrassing situation. The second principle is need and greed, which shows that users are made vulnerable by their desire to fulfill their need or greed. Systems engineers need to be aware of users' needs and work to fulfill them; otherwise an attacker can come along and manipulate the user into thinking that the attacker can fulfill their needs.

Another scam examined was the jewelry shop scam, where the con artists pose as a criminal and a law enforcement official. The criminal tries to purchase a high-value item using counterfeit bills, but the "law enforcement official" steps in and prevents the sale. When gathering the "evidence," the high value item is taken as part of the evidence, along with the fake money. The con artists walk out with the money and the item, successfully completing the scam. The premise behind this is that society trains people to not question authority. The con artist leverages this by posing as a higher-ranking official and getting the victim to do what the con artist wants. Systems engineers need to allow users to challenge authority without the risk of being punished, otherwise users will simply be manipulated by the attackers, since the users think that questioning authority will lead to punishments.

In the last scam, the con artist poses as someone in need of help, such as having a flat tire. The con artist is relying on the kindness of the victim. In the scam, while the good Samaritan is changing the tire, the con artist asks if she can warm up in the victim's car. The victim, trying to be nice, gives the con artist the keys to the car so the con artist can turn the heat on. In reality, the con artist is going to steal the car. The car with the flat tire turns out to be owned by some unsuspecting third party, not involved in the scam. The idea behind this scam is that the con artist takes advantage of people's' kindness, which is how many social engineering attacks on systems occur. The attacker poses as someone in need and hopse that the victim will be kind enough to help the con artist.

The only question asked at the end of the talk was about institutional review boards and how Stajano could perform such experiments, as no IRB would approve such experiments. Stajano responded that his partner, Paul Wilson, was the one actually doing the scams and was not subject to IRB approval. Stajano's role is more analysis after the fact.

REAL-WORLD SECURITY

Summarized by Adam J. Aviv (aviv@cis.upenn.edu)

- ***Scantegrity II Municipal Election at Takoma Park: The First E2E Binding Governmental Election with Ballot Privacy***
  Richard Carback, UMBC CDL; David Chaum; Jeremy Clark, University of Waterloo; John Conway, UMBC CDL; Aleksander Essex, University of Waterloo; Paul S. Herrnson, UMCP CAPC; Travis Mayberry, UMBC CDL; Stefan Popoveniuc; Ronald L. Rivest and Emily Shen, MIT CSAIL; Alan T. Sherman, UMBC CDL; Poorvi L. Vora, GW

Richard Carback presented Scantegrity, a voting system designed such that users can confidentially confirm their vote was counted after the election, along with a verifiable tally. It is the first real-world deployment of such a system.

Scantegrity is an electronic voting system with an optical scan reader. The big difference is "invisible ink." A voter uses a special pen when filling out the ballot, and when she marks an oval, a confirmation number appears. These numbers are different on every ballot, and each ballot has a unique identifier. A user is also provided with a confirmation card, where she can write down her ballot's identifier and the confirmation numbers revealed by her "magic marker." After the votes are counted, a voter can go online and enter her ballot number, revealing the official confirmation numbers, and if there are any differences, the voter can challenge the vote.

The most interesting part of the presentation was when Carback discussed some of the real-world pitfalls and successes of the deployment. Tacoma Park had a turnout of 1,723 voters (a good showing), and the "election ran smoothly."

Only 64 voters checked their votes online, and one complained: it turns out that the magic ink "0" looks a lot like an "8." Voter intent issues also arose. Some voters wrote in a candidate already listed on the ballot and did not mark any bubble; others marked the candidate's bubble and wrote the candidate in. These ballots required a hand count. The team also performed an exit poll, and overall voters responded very well to Scantegrity. In conclusion, Carback claimed, "This stuff is ready to go. We did it!"

Peter Neumann asked about overvotes, and Carback explained that overvotes required hand counting. Someone asked whether someone else could determine who you voted for, and Carback responded that the mix hides this information. Abe Singer wondered how their system handled blind voters, and Carback said Takoma Park fell under federal standards (DC) and thus didn't need to support blind voters. Someone asked about absentee ballots, and Carback said they could have sent them the special pens, but they didn't because of the expense. They did design their own ink for the ballots, fill printer cartridges with the special ink, print ballots, and fill pens with the solution that reveals the hidden numbers in the bubbles that are used to verify votes.

- ***Acoustic Side-Channel Attacks on Printers***
  Michael Backes, Saarland University and Max Planck Institute for Software Systems (MPI-SWS); Markus Dürmuth, Sebastian Gerling, Manfred Pinkal, and Caroline Sporleder, Saarland University

After Michael Backes approached the podium to present his work, the first sound the audience heard was the unmistakable beep and whir of a dot matrix printer, producing a chuckle. Backes then asked, "What was just printed?" The crux of his paper, with co-authors Markus Dürmuth, Manfred Pinkal, and Caroline Sporleder, is to answer that exact question.

Dot matrix printers print documents using one to two rows of needles that strike a page through an ink ribbon, producing dots on the paper that form letters and symbols. Printing different letters produces different sounds, and this was known as early as 1991. However, no one has actually produced an end-to-end attack based on this information. One is likely to ask, "'Aren't dot matrix printers obsolete and not used anymore?" Not true. Backes lists a number of examples where dot matrix printers are the norm, including doctor offices for printing prescriptions. Not mentioned by the presenter, but known to the author of this summary, is that these printers are also shipped with many voting machines to produce verifiable paper trails. The relevance of this attack is broader than one may think.

The attack consists of recording the sounds of the printer as it prints a document. The recording is then passed through a recognition phase to produce a set of initial candidates which are pruned down. Language-based improvements are made using standard Markov modeling and n-grams. The results of the attack are striking: 69% of a message can be recovered without a strong straining corpus, and with a good corpus they saw decoding results as high as 95%. They even performed a real-world attack at a doctor's office (on a fake prescription) and were successful in decoding the document.

Ian Goldberg noted that some printers print left-to-right then come back and print right-to-left. He wondered whether the authors could take advantage of this to improve their attack. Backes replied that other noises they would produce might be helpful. An audience member wondered if this could be used to fingerprint a printer or the language being printed. Backes didn't perform that experiment, but he said that getting the printer model is likely possible and recognizing the language should work well. Another audience member was interested in defenses based on adding additional white noise, such as playing fake printer recordings. Backes acknowledged that this might work, but, again it was not tested. One defense that was tested was placing the printer in a box. It didn't work.

- ***Security and Privacy Vulnerabilities of In-Car Wireless Networks: A Tire Pressure Monitoring System Case Study***
  Ishtiaq Rouf, University of South Carolina, Columbia; Rob Miller, Rutgers University; Hossen Mustafa and Travis Taylor, University of South Carolina, Columbia; Sangho Oh, Rutgers University; Wenyuan Xu, University of South Carolina, Columbia; Marco Gruteser, Wade Trappe, and Ivan Seskar, Rutgers University

The last talk of the session was presented by Wenyuan Xu, an assistant professor at the University of South Carolina at Columbia. She began by noting that there are an increasing number of wireless devices in cars, and computers are integrated deeply into the mechanical systems and display units of newer model cars. These systems were not designed with security in mind, and in this work, Xu and her co-authors exploited the tire pressure monitoring system (TPMS) in demonstrating poor security and privacy design.

TPMS are wirelessly connected sensors placed in the rims of the tires. They become active when the car is moving faster than 25 mph, and then report regularly to an electronic control unit (ECU) connected to the dash-board display. Fortunately, Xu had a car with such a system, and the team set out recording the wireless signal used. Immediately it became clear that it was Manchester encoding, and they were able to decipher the packet format in less than half a day.

There was no encryption, and they also recognized that each packet had a unique ID. This implies that a TPMS message can be used to fingerprint a vehicle and driver, and potentially track them as the car moves about. The next logical question is: What is the signal range? While the car is parked, without an amplified antenna, they were able to record packets at a range of 10 meters. With an amplifier, this increased to 40 meters. They even tested it on the highway while the car was in motion, and again the signal was easily recorded. Xu joked that she will only claim to have

tested the car at 70 mph and no faster so as not to admit to exceeding any speed limits.

The team also investigated transmitting false TPMS packets to fool the driver into pulling over. Xu described highway robbers in Italy who set up road blocks, but with TPMS hacking they only need to send a "flat-tire" signal to get the car to pull over. This was a very effective attack. Even if the ECU received eight good "inflated-tire" packets and one bad "flat-tire" packet, the on-screen display would warn of a flat tire.

As fortunate as Xu was in having a car outfitted with TPMS technology, she was unfortunate in that it was the guinea pig in all the experiments. In fact, she sent so many forged "flat-tire" packets to the ECU that her system died. Kevin Fu asked about bringing the car back to the dealer after she had crashed the computer: "What excuse did you give the dealer?" Xu replied that she was quite reluctant to reveal the exact reason for the failure. "Just reset the computer. The hardware is fine. Trust me."

Brian Rosenberg (Qualcomm) suggested that the reason the computer would signal a flat tire even while receiving four "good" tire signals was the risk in not reporting a flat tire is much greater than in reporting a flat tire that is not really flat. Dan Wallach asked if they had talked to manufacturers and Xu said they did, attempting to find out how sensors are associated with a car. All the manufacturers would say is that a dealer must install replacement tires for the system to work.

INVITED TALK

- ***Vulnerable Compliance***
  Dan Geer, In-Q-Tel

  Summarized by Ben Ransford (ransford@cs.umass.edu)

Dan Geer posed a series of provocative questions about the following topic: what should be done when a vulnerability is found in a specification rather than an implementation? When such a vulnerability has been disclosed, how do we detect and repair the systems that implement the specification and therefore exhibit the vulnerability? Should protocol designers assume that security flaws will be found in their work and design accordingly? Geer's talk featured an interlude with guest Marsh Ray and a lively question-and-answer session.

Geer pointed to historical examples of so-called vulnerable compliance. In each case, the system's wide install base prevented vulnerabilities from completely disappearing for a long time. Mistakes in the Kerberos Version 4 protocol, introduced in 1988 and retired 16 years later (but still undoubtedly in use), were the first example of full compliance with a specification begetting a vulnerability, according to Geer. Predictable TCP sequence numbers were proved vulnerable in 1985 but not corrected in an RFC until 1996. The wireless networking protocol WEP was not reviewed by cryptographers, has gaping vulnerabilities, and is still widespread today. Further examples include a recent DNS cache-poisoning vulnerability, vendors' hurried implementations of IKE with Xauth, and the proliferation of the flawed sign-then-encrypt (and vice versa) paradigm. Geer observed that, in many cases, these vulnerabilities went unfixed until the disclosure of a working exploit, sometimes years after the vulnerability disclosure. (See the article on p. 26.)

A lesson Geer offered to protocol designers is that if you produce a reference implementation, designers of compatible or derivative systems are afraid to diverge from it—especially for complex protocols. Geer gave Kerberos version 4, SNMP version 1, and ASN.1 as examples of specifications that contained vulnerabilities but were sufficiently complicated that most implementers simply followed the reference implementations. The problem with such close adherence to reference implementations is the loss of implementation diversity, a key principle in the design of the Internet. If merely complying with a specification requires substantial implementation effort, little room remains for critical thinking about the specification's flaws.

Geer gave the floor to Marsh Ray, a security researcher known for his recent discovery of a flaw in the renegotiation phase of the TLS protocol. Ray related the long history of a flaw in the way Windows forwards login credentials. Windows uses a protocol called NTLM to store and transmit password-based credentials. Ray noted that CVE reports are still being issued today for a trivial man-in-the-middle vulnerability that has plagued versions of Windows since 1996. He showed a matrix representing the vulnerability's attack surface over combinations of protocols that use NTLM and said that there were still many opportunities to exploit the vulnerability. The vendor has begun fixing the problem but has not made the repaired behavior the default, because they do not want to break backward compatibility. Ray drew several lessons from the saga. If breaking backward compatibility is painful, do it once and comprehensively fix the problem. Highly visible attacks that focus on one facet of a system can distract from potentially more severe underlying vulnerabilities. Protocol designers may find their work burned into silicon, complicating repairs. Encrypted communications can hide the existence of underlying flaws or disguise attacks.

Geer discussed remediation strategies used in the past. He gave examples of top-down remediations, such as when AT&T enlisted the help of legislators to punish phreakers while they invested in more secure protocols. A similar approach to vulnerable Internet nodes might treat such a node as an "attractive nuisance" akin to an unprotected backyard pool. Another possible remediation strategy would be for protocol designers to issue an expiration date for new protocols. He mused that inflection points such as Y2K are an appealing juncture for protocol switchovers. He closed his talk by wondering whether sound defensive strategies might

lead implementers away from Postel's famous Robustness Principle: systems should become conservative in what they produce *and* in what they accept.

Ray showed a video (by Liam Schneider) of credential-forwarding attacks on NTLM while the audience responded to the speakers. An audience member pointed out that many countermeasures, such as patches to DNS servers and TCP implementations, import the notion of security into systems that were designed without security in mind, which seems like a mistake. Ray commented that DNSSEC in particular draws attention away from a need to reform the badly broken PKI infrastructure for SSL. Geer noted that we must sometimes deploy imperfect solutions. Wietse Venema asked whether Geer thought SMTP should have been expired; after all, it lacks authentication and authorization. Geer compared email to financial markets, which have taught us that we can build systems that are too complex to operate, and suggested that perhaps SMTP ought to be made modular, with parts that can be swapped out if they are found to be vulnerable.

David Reed pointed out that users have a need to assign blame for security problems, but that standards committees cannot simply be held liable; he remarked that security is a societal process rather than a property. Ray and Geer agreed; Geer suggested that organizations should allocate resources to deal with security failures. Another audience member wondered why there are no insurance groups that estimate the costs of security failures; Geer said that insurance is enormously complicated but that some organizations have been thinking about it, but only to insure users against failures rather than insure designers against flaws. An engineer in the audience noted that civil engineers, for example, can be held accountable for flaws in the physical artifacts they design (e.g., bridges), but noted that a comparable notion of accountability on the decentralized Internet would unduly hinder development. Geer offered the maxim "Freedom, security, convenience—choose two," and suggested that perhaps the people who deploy, rather than design, systems should be held accountable for failures. As the allotted time drew to a close, David Reed pointed out that although every software company warns against using their software in critical systems, most, with a smile and a wink, upsell their wares into just such systems.

POSTER SESSION & HAPPY HOUR

Summarized by Sandra Rueda (ruedarod@cse.psu.edu) and Rick Carback (rick.carback@gmail.com)

[Editor's note: This session was so popular that it wasn't possible to interview all the poster presenters: it was too noisy, and having good food and drinks made things more difficult. Nevertheless, it was a lot of fun. We are just sorry that not all poster presentations could be covered.]

- ***GuardRails: A (Nearly) Painless Solution to Insecure Web Applications***
  Jonathan Burket, Patrick Mutchler, Michael Weaver, and Muzzammil Zaveri, University of Virginia

GuardRails is a Ruby security tool. It is designed to help developers avoid common Web application security vulnerabilities using annotations instead of explicit security checking code. It provides support for data input sanitization and access controls, and also avoids information disclosure vulnerabilities when access denied errors occur.

- ***Tools for Tracking and Understanding Keyword-Based Internet Censorship***
  Antonio M. Espinoza, Ronald J. Garduño, Leif A. Guillermo, Veronika Strnadova, and Jedidiah R. Crandall, University of New Mexico

This is a probe for detecting words and phrases that trigger Chinese Internet censorship actions. It uses character similarities, named entity extraction, and latent semantic analysis to create the list of censored topics. It can potentially be used as a data source for the Concept Doppler system (ConceptDoppler.org).

- ***Advancing the Science of Cyber Security Experimentation and Test***
  Jelena Mirkovic, USC Information Sciences Institute

DETER is a project that aims to provide a public repository of experiments in the area of computer and network security for educators and students to analyze in related college courses. The experiments include exercises about intrusion attacks and detection, firewall management, spoofing, forensics, denial-of-service attacks, and worm behavior. More information at: http://www.isi.edu/deter/.

- ***MedVault: Health Professional Access to Source-Verifiable Patient-Centric PHR Repository.***
  Mustaque Ahamad, Douglas Blough, Ling Liu, David Bauer, Apurva Mohan, Daisuke Mashima, Bhuvan Bamba, Balaji Palanisamy, Ramkumar Krishnan, Italo Dacosta, and Ketan Kalgaonkar, Georgia Institute of Technology.

MedVault is a project to develop new techniques for the storage, maintenance, and sharing of health records while protecting such records from unauthorized use and disclosure. MedVault uses Merkle hash trees to provide minimal disclosure of information and integrity verification at the same time. The patient only needs to authorize the release of a specific piece of data and the hash codes associated with the remaining branches of the tree for the reader to be able to verify the integrity of the data. MedVault also uses attribute-based policies to release information. Attribute-based policies enable patients to make fine-grained decisions about data sharing.

- ***Redacting PHI in Neurological Images using XNAT***
  Alex Barclay, Laureate Institute for Brain Research and Institute of Bioinformatics and Computational Biology, University

of Tulsa; Nakeisha Schimke and John Hale, Institute of Bioinformatics and Computational Biology, University of Tulsa

XNAT is an open source platform designed to handle medical imaging and data. XNAT uses the DICOM standard (Digital Imaging and Communications in Medicine standard) for handling medical images. The problem is that the DICOM standard includes Protected Health Information (PHI), that is, information that can be used to identify an individual. Furthermore, the image itself may include information that can be used to identify an individual. This poster highlights the need for a tool to redact the entire PHI data stack, including DICOM headers, text, and the image byte stream, to ensure privacy of the data.

- ***Embedded Firmware Diversity for Smart Electric Meters***
  Stephen McLaughlin, Dmitry Podkuiko, Adam Delozier, Sergei Miadzvezhanka, and Patrick McDaniel, Pennsylvania State University

Current smart meters belong to a category that is known as monoculture, meaning that a large percentage of these meters have the same hardware and software. From a security point of view, monocultures represent a high risk since attacks that succeed on one of the elements can be repeated on all of them without much additional effort. Traditionally, software diversity techniques have been used to mitigate attacks on monocultures. However, the techniques that can be used on smart meters are limited because of the hardware requirements associated with many of them and the hardware limitations of the smart meters.

This poster presents redundant address encryption to provide "lightweight control flow integrity" to prevent random errors after an exploit attempt. Redundant encryption using different keys to protect return addresses provides reasonable guarantees to protect the smart meters.

- ***Process Firewalls: Mechanism and Utility***
  Hayawardh Vijayakumar, Sandra Rueda, Divya Muthukumaran, Joshua Schiffman, and Trent Jaeger, Pennsylvania State University

Current operating systems support access control policies at the granularity of a program and cannot enforce finer-grained access control policies. Therefore, an operating system's ability to enforce a policy depends on what interface a program is using to access a given OS resource. This project proposes a Process Firewall mechanism to enforce policies with a finer granularity that would allow access based on what interface a program is using to access a given OS resource. This behavior is analogous to a regular firewall's behavior that enforces policies for a given host based on network features such as a port number.

- ***Graph Cuts Can Be Used to Solve Security Problems***
  Divya Muthukumaran, Dave King, and Trent Jaeger, Pennsylvania State University

This poster proposes that security problems arising from information flow errors can be modeled as a graph cut problem. A cut solution to the graph cut problem is a solution for the security problem. This kind of problem includes mediation placement in programs (placement of declassifiers and endorsers), privilege separation (since we want to split the code), and policy error resolution (errors indicate illegal information flows and thus a cut suggests where to mediate the flow). The challenges to model the problem include identifying sources and sinks, and converting cuts to the appropriate security solutions.

- ***Securing End-to-End Provenance: A Systems and Storage Perspective***
  Kevin Butler, University of Oregon; Patrick McDaniel, Stephen McLaughlin, and Devin Pohly, Pennsylvania State University; Radu Sion and Erez Zadok, Stony Brook University; Marianne Winslett, University of Illinois

This paper presents a mechanism, Kells, that enables a USB device to evaluate the integrity of the host it is being connected to, before releasing any of the information it stores.

Since Kells can identify the machine that it is plugged into, it is possible to build a provenance chain at the block level based on reads and writes from a given machine. Once the host is validated by the device, it can be considered to be within the TCB, so requests are trusted. At the block level there is no concept of users per se, but the device can consider users through other means, such as biometrics on the USB drive.

- ***Verifying Cloud Integrity: Making the Cloud Do the Dirty Work***
  Joshua Schiffman, Thomas Moyer, Hayawardh Vijayakumar, Trent Jaeger, and Patrick McDaniel, Pennsylvania State University

This work addresses two questions: (1) How do we ensure the integrity of the results produced in a cloud environment? (2) How can customers verify integrity?

This project designed and implemented a cloud verifier (CV) to answer these questions. The cloud verifier is a component in the cloud that can verify the integrity of the virtual machine monitors (VMMs) in the cloud. It does so based on an integrity criterion that is shared with the customers. Customers decide if the CV criterion meets their own. The CV also provides an IPSec key that customers can use to establish trusted sessions with their own VMs (for instance, to send keys to access encrypted data stored in the cloud). This key is generated by a VMM for the VMs it is hosting. Since the CV has verified the VMM's integrity, it signs the key and sends it to the customer.

- ***tNAC: Trusted Network Access Control***
  Ingo Bente, Josef von Helden, and Joerg Vieweg, Trust@FHH Research Group, University of Applied Sciences and Arts, Germany; Marian Jungbauer and Norbert Pohlmann, Institute for Internet Security, University of Applied Sciences, Germany

The tNAC project aims to develop a trustworthy Network Access Control solution. tNAC builds upon Turaya, the

secure operating platform, and TNC@FHH, the open source Trusted Network Connect implementation. tNAC ensures that by integrating the capabilities of Turaya, which are rooted in the Trusted Platform Module, and TNC@FHH, which gathers security relevant information about each endpoint, only those endpoints that match the policy of the provider will be allowed to access the network. Endpoints that try to lie about their current security state will be detected. For further information about tNAC, please visit www.tnac-project.org.

- ***Moving from Logical Sharing of Guest OS to Physical Sharing of Deduplication on Virtual Machine***
  Kuniyasu Suzaki, Toshiki Yagi, Kengo Iijima, Nguyen Anh Quynh, and Cyrille Artho, National Institute of Advanced Industrial Science and Technology; Yoshihito Watanebe, Alpha Systems, Inc.

This is a proposal to use memory- and storage-deduplication to increase security. Application binaries are translated by pseudo-static converter (for example, "statifier" in Linux). The binaries share necessary libraries and prevent search path replacement attack, GOT (Global Offset Table) overwrite attack, Dependency Hell, etc. They require more storage and memory, but deduplication techniques reduce the increase.

WEB SECURITY

Summarized by Manuel Egele (megele@cs.ucsb.edu)

- ***VEX: Vetting Browser Extensions for Security ulnerabilities***
  Sruthi Bandhakavi, Samuel T. King, P. Madhusudan, and Marianne Winslett, University of Illinois at Urbana-Champaign

  Awarded Best Paper!

Firefox currently has around 25% market share, and 150 million plug-ins (i.e., extensions) are in use. Firefox extensions are written in JavaScript and executed in the same context as the chrome, the browser's frame and controls. Extensions run as part of the browser and thus have access to everything you do with your browser. After giving a brief overview of the current submit process for Firefox extensions and its weaknesses, Sruthi Bandhakavi elaborated on the idea and the threat model behind VEX.

Extensions are assumed to be benign and vulnerabilities to be the effects of buggy extension code. Vulnerabilities can be exploited by a malicious Web site. To protect from these threats, VEX employs static analysis to check for explicit information flows that bridge the two trust domains for JavaScript in the Firefox browser: the chrome and content contexts.

VEX identified a vulnerability in an RSS reader extension. Bandhakavi prepared a demo exploit to attack this vulnerability and demonstrated the effects. VEX uses abstract heap data structures for objects, methods, and properties to compute precise flows between objects. Currently, VEX contains three different flow patterns, and the authors were able to identify six vulnerabilities in 2452 extensions they analyzed with VEX.

The presentation concluded with a glance at future work: Bandhakavi said that they want to study and classify known vulnerabilities, and employ a constraint solver to improve VEX. The project Web site can be found at http://www.cs.illinois.edu/~sbandha2/VEX/.

Peter Neumann asked about the limitations of the employed flow analysis and how we can get out of the unfortunate situation that we have untrusted operating systems, browsers, and browser plug-ins. Bandhakavi answered that the limitations for static analysis apply to VEX too. However, VEX was designed as a bug finding tool and thus is not able to state the absence of bugs. More effort should be put into designing languages that can be analyzed reliably. Someone asked about false positive and false negative evaluation and where the ground truth comes from. Bandhakavi replied that VEX did not detect all known vulnerabilities. For example eval constructs still pose a limitation to VEX. Two undergrads worked to systematically create attacks employing fuzzing techniques, but it was really tough to create such a tool, because each extension is unique in what inputs it accepts. She emphasized the need for tools like VEX that could at least point to the presence of an attackable flow in order to test the extensions manually. The flows detected in the extensions could eventually turn out to be not attackable for various reasons outlined in the paper and therefore become false positives.

Collin Jackson (CMU) wondered how many extensions loaded content that got executed in the chrome context from HTTPS-secured URLs instead of regular HTTP. He asked why one would ever allow content from nonsecure sources to be passed to the eval statement. Bandhakavi felt that only allowing HTTPS sources might be too restrictive.

Helen Wang asked how VEX compared to inline monitor approaches that are built into the browser. Bandhakavi clarified that VEX is intended to help extension editors to vet extensions before they get approved, and thus is able to find vulnerabilities before they get deployed to the browser.

- ***Securing Script-Based Extensibility in Web Browsers***
  Vladan Djeric and Ashvin Goel, University of Toronto

Vladan Djeric presented their work to provide protection against privilege escalation vulnerabilities in script-based browser plug-ins. Djeric started his presentation with a brief overview of the Firefox architecture. One of their design principles was to implement their approach with no modification to existing extensions. Djeric then divided existing vulnerabilities into three classes: code compilation vulnerabilities, luring vulnerabilities, and reference leaks. The threat model assumes benign extensions and untrusted data being executed as privileged code. They added a dynamic taint propagation engine to the Firefox browser.

Existing security measures in Firefox advocate the use of a taint propagation scheme. For example, name space separation in the browser creates a natural boundary for taint labels, and privileged scripts usually handle untrusted data with care.

Based on the taint propagation scheme, the authors implemented techniques to detect vulnerabilities in all of the three vulnerability classes. The authors implemented and evaluated a prototype of their technique in Firefox 1.0.0. The reason to choose this rather old version is that the published security bulletins are very detailed. Out of 14 advisories their approach was able to detect 13 vulnerabilities. The only vulnerability that was not detected results from an incomplete implementation. More precisely, the authors did not implement the taint propagation throughout the HTML parsing engine. To evaluate the false positive rate, the system was exercised by a human Web surfer for five hours, resulting in one false alert. Also, an automated crawler visited the top 200 Web sites of the Alexa Web site ranking, also resulting in one false positive. The performance evaluation showed slowdowns up to 30% in microbenchmarks.

Ian Goldberg (University of Waterloo) wondered how the system handles JavaScript closures. According to Djeric, using closures to interact between trusted and untrusted content is not common. Peter Neumann wondered whether their approach could benefit from a more fine-grained interpretation of taint, as opposed to the binary tainted/not-tainted scheme. Djeric responded that he prefers to err on the side of caution. Venkat Venkatakrishnan (University of Illinois, Chicago) compared this work with the previous talk on VEX and asked whether static or dynamic analysis techniques are better suited to protect the user from vulnerable extensions. Djeric stated that their approach does not only detect vulnerabilities in extension but also in the browser itself, if, for example, vulnerable wrappers are present. Sruthi Bandhakavi, the presenter of the previous talk, described a problem with dynamic analysis: once a problem is detected, the user has to make a decision on how to proceed (i.e.., ignore warning and continue or terminate the execution).

David Wagner (University of California, Berkeley) wondered about the methodology that was used to measure the 30% performance impact. Djeric agreed that this slowdown is not negligible but said that it's too little to be perceived in day-to-day browsing. Niels Provos (Google) said that the user cannot trust extensions. Djeric agreed and reiterated that their work was aimed to protect the user from vulnerabilities in benign extensions.

- ***AdJail: Practical Enforcement of Confidentiality and Integrity Policies on Web Advertisements***
  Mike Ter Louw, Karthik Thotta Ganesh, and V.N. Venkatakrishnan, University of Illinois at Chicago

Mike Ter Louw presented AdJail. He introduced a running example of the Yahoo Webmail interface that he would use throughout the presentation and discussed the issue of a context-sensitive ad on Facebook that would fetch the profile pictures of a user's friends and use them in dating service advertisements. The specific example suggested that the user might be advised to date his own wife through this dating service. The presentation continued by stating five design goals for AdJail. These goals are confidentiality and integrity of sensitive page data, a consistent user experience, support for ad scripts that perform contextual advertisement, transparency towards the ad-networks, and support for all major Web browsers.

AdJail creates a shadow page for each real page that contains the unmodified ad script. Access to content of the real page is mediated by two JavaScript components embedded in these two pages, the real and shadow pages. These components employ DOM interposition and are responsible for mediating access and forwarding events. Furthermore, AdJail defines a policy language to annotate read and write properties for certain content areas pertaining to ad scripts. The AdJail prototype was evaluated with six ad networks and was integrated with the Roundcube Webmail application. Their prototype implementation resulted in a slowdown of around 200ms for rendering the advertisements.

Niels Provos (Google) wondered how this approach relates to confidentiality breaches, where ad scripts steal browser history, and how this work relates to Caja. Ter Louw replied that such attacks are outside the scope of their work. Also, Caja has to transform the ad script before it is delivered, which is not necessary for AdJail and is undesirable, as it may raise false positives in ad networks' click-fraud detection mechanisms. Lucas Ballard (Google) asked how Flash advertisements are handled. In AdJail, Flash advertisements cannot interact with JavaScript. Algis Rudys asked whether AdJail allows the publisher to limit the write access to areas where context-sensitive ads will be placed (e.g., an ad should only be able to add content, such as links for keywords, but not be able to rewrite the whole content). Once a region is marked as writeable, the ad can perform any modification to the area, including a complete rewrite.

Matt Jones (Facebook) wondered whether the amount of data that is transmitted to the ad network can be limited, or if an ad script could send the whole email content to the ad network. Ter Louw answered that, commonly, only keywords are extracted and transmitted, but in general it would be hard confining such behavior. The last question was geared at finding out how the ad script and the AdJail scripts communicate and whether a malicious ad script could talk to the AdJail script in the original page directly, bypassing the protection. Ter Louw responded that the

AdJail script on the real page does not trust anything from the shadow page, and all policies are enforced in the AdJail component on the real page.

INVITED TALK

- ***How Cyber Attacks Will Be Used in International Conflicts***
  Scott Borg, Chief Economist, US Cyber Consequences Unit

  Summarized by Sunjeet Singh (sstattla@gmail.com)

Scott Borg, an expert in the area of cyber warfare, assesses cyber security risks to the US and closely studies ongoing cyber conflicts internationally. Borg discussed various recent real-world examples to draw a line between the true potential of cyber attacks and the actual extent to which cyber attacks play out today. He then presented specific statistics that explain the strategic implications of such cyber attacks and said that cyber attacks are set to become the major form of warfare in the future. (During his talk he repeatedly cited his summary on the recent conflict between Russia and Georgia: search for US-CCU-Georgia-Cyber-Campaign-Overview.pdf.

Cyber attacks offer many unique advantages over physical attacks, mainly in that they can be anonymous, highly targeted, overwhelming in impact, and, at the same time, reversible. The reliance of any nation on information technology makes it a prospective target for cyber attack. Apart from the critical infrastructure, many modern weapon systems in use today use IT, and this makes cyber security all the more crucial. Cyber wars have been witnessed at several levels in recent conflicts all over the world, with each successive conflict increasingly sophisticated.

In the recent conflict between Russia and Georgia, there was high strategic coordination between cyber and physical attacks. Although there is no firm evidence that Russia was behind the cyber attacks that took out Georgian government Web sites, media communications, and power infrastructure during that period, all these events were so highly synchronized with on-ground military advances that it seems implausible that a third entity could have been behind the cyber attacks. It is believed that the Russian cyber attackers had control over much more of Georgia's critical infrastructure than they exercised, which would go to show that the attack was highly organized and disciplined. Georgia in turn came up with a counterattack by releasing malware on social networking Web sites using the Russian language, thus targeting Russian users. The suffering of Georgia from this war has left behind bitter traces in the minds of Georgian people, which suggests to Borg that future attacks might not be as controlled as the Russian attack was.

In attacks less controlled than Russia's, it is likely that a local conflict could lead to a global impact. In a recent staged government experiment, hackers were able to seize control of a US power grid generator and caused it to self-destruct. Having established that critical infrastructure can be attacked and that physical damage can be inflicted by cyber attacks, it is reasonable to assume that for higher-value targets such as pipelines and refineries, the damage would be severe. For example, a disruption in electronic supply or oil and gas chains in Asia would cause global repercussions.

Given the potential impact, unlike many specialists in this field who believe that cyber warfare will supplement conventional warfare and act merely as a force-multiplier, Borg argued that cyber techniques will govern physical techniques to become the major weapon in future. The purpose of any war is to establish control over the adversary, and cyber warfare provides the means to do it in an effective manner.

At this point, the audience had questions on the practicality of large-scale cyber attacks, e.g., on a nation's complete power grid, on how well such attacks can be controlled, and on how asymmetric the attacking and defending sides can be. To these, Borg's reply was that the whole world is high-tech today. Low-launch attacks from minimal infrastructure and from any part of the world can potentially cause great impact. Although it is not easy to take advantage of an attack in a controlled fashion, it is much easier to inject malware to cause damage.

SECURING SYSTEMS

Summarized by Andres Molina-Markham (amolina@cs.umass.edu)

- ***Realization of RF Distance Bounding***
  Kasper Bonne Rasmussen and Srdjan Capkun, ETH Zurich

Kasper Rasmussen presented a way to realize a distance bounding protocol using RF communication. Distance bounding protocols are run between two entities, the verifier and the prover. The prover's goal is to prove to the verifier, using a challenge response protocol, that he is within a given physical distance from the verifier. More precisely, in a model where the verifier is trusted and the prover is untrusted, the prover cannot pretend to be closer than he really is. Also, after the protocol is run, the verifier has proof that the prover is within a certain distance.

Rasmussen noted that robustness in a distance bounding protocol comes from requiring that an attacker must take essentially zero processing time to respond to challenges. The authors propose the use of Challenge Reflection with Channel Selection (CRCS) in distance bounding protocols instead of bounding protocols that require the prover to interpret the received bit before replying to it. Not only is interpreting unnecessary, but it is the reason why alternatives are slow. Rasmussen explained that even alternatives that implement this using XOR are inadequate, not because XOR itself is slow, but because protocols require that full symbols be received before processing them, and receiving a

symbol can take microseconds. The fastest known approach relying on XOR has a processing time of 300 ns, which translates into an error in distance bounding of 50 meters. In contrast, the proposed solution that uses CRCS is well suited for distance bounding because it does not require the interpretation of the bit received before replying. This allows the prover to receive, process, and send a challenge in less than one nanosecond, which translates into an error in distance bounding of about 15 centimeters.

The main idea of this approach is that, using two channels, the prover reflects a challenge back to the verifier without interpreting it. The use of one channel would encode a 1 and using the other would encode a 0. Thus the prover's choice of a channel would encode a bit of knowledge of a nonce. A distance bounding protocol would, in addition, rely on cryptographic signatures and the integrity of a challenge to protect against two attacks, distance fraud and mafia fraud. Rasmussen described a wired implementation and referred interested members of the audience to the paper for a wireless implementation.

Ian Goldberg (University of Waterloo) noted that a prover could collude with an external attacker that is closer to the verifier to prove that the prover is as close as the attacker. Rasmussen responded that in that case the attacker becomes the prover, and it is just a matter of preventing the prover from sharing his credentials. Avishai Wool from Tel Aviv University noted that in the wired implementation described, high frequencies (~3.5 GHz) were used, but that some important applications, e.g., contact-less cards, work at low frequencies (~13 MHz) and with slow processors. He asked if the proposed solution would still apply in these cases. Rasmussen responded that in theory the approach should still be valid but that it would be an engineering challenge to deal with such low frequencies. Another member of the audience asked if the mixer in the proposed approach could be replaced by switching a modulation on and off to encode a bit, for example. Rasmussen responded that indeed other approaches are possible, as long as they avoid symbol processing and interpretation before replying.

- ***The Case for Ubiquitous Transport-Level Encryption***
  Andrea Bittau and Michael Hamburg, Stanford; Mark Handley, UCL; David Mazières and Dan Boneh, Stanford

Andrea Bittau presented tcpcrypt, a TCP extension that would enable end-to-end encryption of TCP traffic by default. He started by listing the three main requirements for a solution that would encrypt the vast majority of TCP traffic: performance, endpoint authentication, and compatibility with existing networks and legacy applications. He then said that no existing solution achieves all three.

Bittau provided examples in which tcpcrypt would improve the security guarantees on sites like CNN, Amazon.com, Facebook, or Bank of America. He hinted that this could be done while also improving overall performance. Next, he listed some advantages and disadvantages of providing security at an application layer with SSL or at the network layer with IPSec. In particular, he mentioned that while IPSec could work with all applications, it could break NAT and would not be able to leverage user authentication. These claims about IPSec would later be challenged by a member of the audience.

The authors claimed that tcpcrypt would provide high server performance by pushing complexity to the clients, would allow applications to authenticate endpoints, and would provide backwards compatibility with all TCP applications, networks, and authentication settings. Performance is achieved because encryption and decryption operations in RSA are not equally expensive. Thus, it is possible to design a protocol in which the cheap operations are on the server side. Doing so would allow servers 36 times better performance than SSL. However, this would require a different approach to authentication, using session IDs. Additionally, tcpcrypt would use existing SSL infrastructures to batch-sign session IDs and thus amortize the cost of RSA operations. In order to provide compatibility, tcpcrypt would modify the initial SYN-TCP with a SYN-CRYPT to probe for tcpcrypt support. If the server ignores the probe, the client would fall back to regular TCP. However, if the server supports tcpcrypt, then both parties would continue with a tcpcrypt negotiation encoded in TCP options.

After going over various protocol and implementation details, Bittau explained that even though better performance can be achieved with tcpcrypt than with SSL, performance gains would vary according to various ways of providing authentication. Bittau referred the audience to http://tcpcrypt.org to obtain a copy of tcpcrypt and install it in their systems. The authors offer tcpcrypt in a Linux kernel implementation and a userspace implementation that runs on Windows, Mac OS, Linux, and FreeBSD. Bittau concluded his talk by demoing tcpcrypt on a Web application that allows clients using tcpcrypt to post messages into the tcpcrypt Hall of Fame.

David Reed pointed out that piggy-backing on the SYN packet may allow DoS attacks. Bittau responded that tcpcrypt is implemented using mini-sockets requiring one single bit, and thus the server state on the SYN is cheap. Another member of the audience said that by using tcpcrypt instead of IPSec, one would lose protection on other transport layer protocols such as UDP or SCTP. He also challenged Bittau's claims about IPSec not being able to provide individual authentication and not being able to play with NAT. Bittau responded that, indeed, the authors had restricted their attention to TCP traffic, which is the majority of the Internet traffic. As for his previous claims, Bittau stood by them and invited the member to continue the discussion offline.

- ***Automatic Generation of Remediation Procedures for Malware Infections***
  Roberto Paleari, Università degli Studi di Milano; Lorenzo Martignoni, Università degli Studi di Udine; Emanuele Passerini,

Università degli Studi di Milano; Drew Davidson and Matt Fredrikson, University of Wisconsin; Jon Giffin, Georgia Institute of Technology; Somesh Jha, University of Wisconsin

Lorenzo Martignoni proposed an architecture that can be used to automatically generate procedures to repair a system after it has been infected with malware. Martignoni made the case that preventing an infection is not always feasible and that current malware detection software does not always leave systems in a stable and safe state after repairing them. The authors showed that their approach was able to revert 98% of the activities performed by 200 pieces of malware, in comparison to the 82% achieved by the best leading commercial solution.

Martignoni described the challenges of generating remediation procedures. One complication is that malware code is usually obfuscated and, therefore, hard to analyze. Moreover, the behavior of this type of software is typically non-deterministic, and remediation usually takes place only after an infection has been detected, so the previous state of the system is not completely known. Next, Martignoni described their approach, which consists of three steps: (1) they construct "infection relations" by extracting generalized patterns of clusters on behavior graphs obtained by running the malware in diverse controlled systems; (2) infection relations are then used to construct remediation procedures; and (3) these procedures are performed in the infected system to revert the effects described by the infection relations. The major limitation of this approach is that attackers could increase the behavior generalization of their malware, thereby decreasing the ability for this system to obtain complete results. Also, only a subset of modified resources can be properly restored. In particular, deleted files or user files cannot be restored.

Katsunari Yoshioka (Yokohama National University) asked about the malware samples used for the evaluation part of the paper. Katsunari explained that in his experience these are hard to analyze because they are often not self-contained, so parts of their code may be obtained from remote locations. Martignoni agreed and added that, in fact, some pieces of malware may simply crash and stop working. However, these were not considered in the paper.

INVITED TALK

- ***Grid, PhD: Smart Grid, Cyber Security, and the Future of Keeping the Lights On***
  Kelly Ziegler, Chief Operating Officer, National Board of Information Security Examiners

  Summarized by Leif Guillermo (laag@unm.edu)

Kelly Ziegler explained that the talk would be kept at a high level for a policy perspective and would explain how the electric grid works and how the smart grid came to be. She hoped this would provide a useful background for understanding some of the issues we are facing now related to cyber security and other security-related issues. She also mentioned that at the end of the talk she would speak about the regulatory framework surrounding the power grid.

There were three main areas of focus on the power grid: power generation, transmission, and distribution. There are roughly 5000 power plants with roughly 160,000 miles of power lines distributed over one million square miles. The North American power grid can be broken down into three interconnections. These interconnections are described as the eastern connection, the western connection, and ERCOT, which is located in Texas. These connections can be thought of as the largest machines in the world, because they are all synchronized.

Between supply and demand, energy output must meet energy demand at every instant. There are three main energy supplies, and a supplementary supply. For the base load of energy demand—large consumers of electricity such as factories and commerce—coal power is generally used. The intermediate energy load requires gas units. Finally, for peak loads and supplementary supply, natural gas is used. Peak loads generally occur between 3:00 and 6:00 p.m.

There is an imaginary barrier known as the "Chinese Wall" which separates bulk power system policies from distribution policies. The bulk policies are regulated at the federal level, and these policies deal with power plants and power generation, whereas distribution is regulated at the state level and deals with how power is transferred to consumers. Due to this barrier, regulating demand can be problematic. The smart grid is a temporary solution to gain control over demand. It implements a variety of solutions: automatic meter reading, distribution automation and generation, demand response, and supervisory control and data acquisition (SCADA) control systems and sensing. Automatic meter reading was the first temporary solution, deployed earliest on major industrial and commercial locations. This method provided a more detailed hourly and time of use billing. It also helped to cut down on the number of meter readers and allowed for various different configurations depending on the needs of the user. Distribution and transmission system automation allows operators greater control and management. Distributed generation allows for smaller generating units to serve the energy load locally. Demand response is a technique to flatten out the peak of energy consumption. One implementation of this method is for people to opt to reduce their basic energy rate, but when it's really hot outside and the energy load is very high, the rate is increased.

In the 1990s, deregulation occurred in the electricity sector, which continues to allow people to trade electricity. Since then, there have been huge amounts of growth in the energy sector. Eighty-five percent of relays are now digital. Originally security was not a big design requirement for the power grid, but after the terrorist attacks of September 11, 2001, we started realizing that security is actually a big issue for us. Since the original design of the grids wasn't

implemented with security integration in mind, the issue of security has become a very troublesome obstacle to tackle.

The greatest threat is the potential for an attacker to attack multiple key nodes on a system. Both physical security and cyber security are enormous issues, and new vulnerabilities arise all the time. Before addressing security, however, many business issues need to be addressed in order to be sure that the security issues are feasible. Managing the risk of implementing security measures seems to be the most important piece in keeping the power grids safe.

There are nine critical infrastructure protection standards designated by the North American Electric Reliability Corporation (NERC), which reports to the Federal Energy Regulatory Commission. NERC is self-regulatory and is governed by the utility companies. NERC's argument is that if they don't make certain requirements critical, the utilities don't have to comply with those requirements, so this is another roadblock in the way of security. An important idea to take away from the talk is that the current regulatory structures set in place to address cyber security in smart grid technologies are inadequate, in part because of the complexity of the whole smart grid system and the fact that the smart grid wasn't designed with a high level of security in mind.

Many questions focused on attacks that destroyed transformers and on the resiliency of the existing system. Evo Dismet pointed out that destroying a substation could take out a city for a year. Ziegler responded that taking out three or four substations would cut off DC from power. Some substations use very large custom-designed transformers and take over 18 months to build. Cathy Jenks of Sun/Oracle asked if the US has agreements with the countries who manufacture this equipment, and Ziegler pointed out that Aviva, in France, would likely replace transformers in France before they would help other countries. Jessica Smith from MITRE wondered about communication and load balancing, asking if it made sense to connect the east and west networks. Ziegler said that it didn't, although it had been considered for better use of renewables. But things are quite reliable as they are, and connecting the two networks might create more unreliability.

Steve McLaughlin of Penn State asked about spare equipment at substations to prevent cascading failures. Storm restoration is something utilities do all the time. But transformers are hugely heavy and very difficult to move around, although mobile transformers do exist. But if a cyber attack occurred that took out many nodes, recovery could take years.

USING HUMANS

Summarized by Femi Olumofin (fgolumof@cs.uwaterloo.ca)

- ***Re: CAPTCHAs—Understanding CAPTCHA-Solving Services in an Economic Context***
  Marti Motoyama, Kirill Levchenko, Chris Kanich, Damon McCoy, Geoffrey M. Voelker, and Stefan Savage, University of California, San Diego

Marti Motoyama began this talk by describing the goal of the paper, which is to evaluate CAPTCHAs as a security mechanism by looking at CAPTCHAs-solving ecosystems. CAPTCHAs, or Reverse Turing tests, are first-line defense mechanisms against large-scale, automated exploitation of Web resources. In their most common form, CAPTCHAs consist of alphanumeric characters distorted in some ways and are presented as visual challenges to the user. CAPTCHAs are easily solved by humans, are easily generated and automated, but are hard to solve by computers.

To help attackers circumvent the defenses posed by CAPTCHAs on targeted Web sites, commercial CAPTCHA-solving services have emerged consisting of automated software solvers and third-party human solving services. Some of the identified limitations for software solvers are the requirement for skilled programmers, difficulties in achieving high accuracy, and the ease with which defenders (i.e., designers of CAPTCHAs) can adapt and defeat solving algorithms using better obfuscation of their CAPTCHA challenges. Marti said that it does not make sense to invest in software solvers. Even the popular Xrumer solver has been adapted recently to leverage human-based CAPTCHA-solving services.

The solving market is globalized because of several factors, including cheap Internet access, the commodity nature of CAPTCHAs nowadays, and the non-specialized skill requirements for solving CAPTCHAs. It is easy for these service providers to aggregate on-demand CAPTCHA-solving requests and outsource them to workers recruited from some of the lowest-paid labor markets around the world. Many of these services are able to solve CAPTCHAs for retail prices as low as $1 per thousand. Wholesale and retail prices are declining in this demand-limited market.

In this study, the authors tried to understand the security of CAPTCHAs by asking economics questions that compare the cost of solving CAPTCHAs, using either of the two approaches, to the cost of the assets that CAPTCHAs protect. Essentially, CAPTCHAs add friction to the business models of attackers and should be evaluated in terms of how efficiently they can undermine attackers' profitability. Some of the findings from the study were validated in an interaction with the owner of a successful CAPTCHA-solving service.

Stephen Jenbecky from MITRE suggested the use of culturally dependent CAPTCHAs, such as ones that pose visual challenges that depend on a geographical area. Such

CAPTCHAs can help reduce the effectiveness of foreign human laborers used by CAPTCHA-solving services. Jeremy Epstein (SRI International) asked how many times Klingon (Star Trek) CAPTCHAs were tried, and Motoyama said 222. Epstein commented that human solvers could learn from examples. Cody Cutler asked about the legitimacy of CAPTCHA-solving services, and whether or not such services pay their workers. Motoyama said they did pay their workers.

- ***Chipping Away at Censorship Firewalls with User-Generated Content***
  Sam Burnett, Nick Feamster, and Santosh Vempala, Georgia Tech

Sam Burnett described Internet censorship as a global problem not limited to oppressive regimes alone but including democratic governments as well. Existing solutions to defeat censorship and surveillance of network communications rely on helpers (e.g., proxies) to relay communications between users in a censored regime and those outside the censored regime. Commonly used anti-censorship systems, such as Tor, have three shortcomings. First, it is easy for censors to block proxies if the proxy list is public. Second, a user in a censored regime can often not deny participating in a communication. Third, the success of such systems relies on the benevolence of volunteers outside the censored regime to provide a network of proxies (i.e., requires dedicated infrastructure).

Burnett called their solution Collage, which is a method for bypassing censorship firewalls by hiding messages inside user-generated content such as photos on Flickr, tweets on Twitter, and videos on YouTube. The vast amounts of user-generated content on many Web sites provides an unlimited amount of cover traffic that makes it difficult for censors to block all possible sources (i.e., no dedicated infrastructure to block). Burnett said that they have developed tools to store censored data in user-generated content using such techniques as steganography and watermarking. Unlike Tor, where a user is easily implicated by merely contacting a relay, Collage provides its users with some level of deniability, since they can hide their messages inside harmless-looking messages (e.g., photos, videos, etc.).

Sending a message with Collage requires the sender to obtain the message and pick a message identifier for the message, which should only be known to the intended recipient. Then the sender obtains cover media such as personal photos and embeds the message in the cover media. Next, the sender uploads the user-generated content to some hosts. The receiver can then find and download the user-generated content from the hosts and extract the message from it. Embedding a message into cover media consists of two steps: (1) encrypt the message with the message identifier; (2) split the ciphertext into many chunks using erasure coding. Each erasure-encoded chunk corresponds to a task, and the ciphertext can be reconstructed from any k-subset (i.e., offers robustness). Another problem addressed is how message receivers can identify the locations of message vectors without having to crawl the entire user-generated content on a host, and without any immediate communication with senders. Their solution was to use task mapping to map message identifiers to these locations. Senders publish message vectors so that receivers can get the vectors when they perform tasks. For example, a task may be for the receiver to search YouTube or Flickr with a particular keyword.

The performance metrics for Collage include sender and receiver traffic overhead, sender and receiver transfer time, and the storage required on content hosts. These metrics vary a lot depending on the content host and type of tasks that receivers need to perform in order to retrieve message vectors. Burnett described a case study on sending a news article and covert tweets using Flickr and Twitter as content hosts. The message sizes were 30KB and 140 bytes, receiving times were two minutes and half a minute, and storage needed on hosts 600KB and 4KB, respectively. Sam also ran a demo of a Collage application, which is available for download at http://gtnoise.net/collage.

The presentation ended with highlights of some areas for further research, such as statistical deniability against traffic analysis, learning timing behavior from users, and Tor bridge discovery.

- ***Fighting Coercion Attacks in Key Generation using Skin Conductance***
  Payas Gupta and Debin Gao, Singapore Management University

Payas Gupta began this talk by saying that many techniques have been proposed to generate strong cryptographic keys. While some of these techniques—biometrics, for example—possess desirable security properties such as ease of use, unforgettability, unforgeability, and high entropy of the keys, they cannot resist coercion attacks. In this attack, the adversary forces the user to reveal the key. The focus is on finding ways that would make the user incapable of generating correct keys when he or she is coerced. They assumed that the adversary knows about the coercion-resistant property; otherwise the user's inability to generate a correct key might be interpreted as stubbornness, and that could endanger the life of the user.

Gupta described their proposed solution to achieve coercion resistance, which is to incorporate users' emotional status or arousal state, through the measure of skin conductance, into the process of key generation. They extended a previously proposed biometric key generation technique that relies on voice, to use both voice and an emotional response parameter of the user's skin conductance. Key generation follows a look-up approach based on the original biometric key generation technique. Their reason for choosing skin conductance over other physiological signals (e.g., heart rate, skin temperature) was because skin conductance is cheap to measure and the deviation in measurements is small.

Gupta described a user study to evaluate their solution consisting of 39 participants (22 male and 17 female) who were undergraduate and graduate students aged between 18

and 30. They ran two experiments to capture the emotional response of participants, using skin conductance sensors attached to their fingers, when they were in a calm condition and when they were stressed. Each user generated a cryptographic key in each state. The approach used to stress participants was by showing them a frightening horror movie. The result of the study shows that different cryptographic keys were generated for the two experiments and the approach has moderate false positive and false negative rates.

Someone asked whether the authors obtained internal ethics approval before conducting the user study. Gupta confirmed that they did. The same person was concerned about why they had to put participants in such a high-stress situation and questioned the validity of their result because many variables might be going into the result without them knowing. Another person commented that skin conductance might depend on the climate of the room where the person is located. The same person said that skin conductance is a measure of stress, which may be unrelated to whether or not a person is coerced. Lucas Ballard (Google) commented that sometimes it might be difficult to detect why authentication failed even in a non-stressed situation, due to high variability in biometric measurements (i.e., voice and/or skin conductance). The failure of either or both of these could be due to other factors such as noise in the environment, illness, or tiredness.

INVITED TALK

- ***End-to-End Arguments: The Internet and Beyond***
  David P. Reed, MIT Media Laboratory

  Summarized by Joshua Schiffman (jschiffm@cse.psu.edu)

David Reed started his talk by providing a historical background that led to the publishing of his original End-to-End (E2E) argument paper, which he notes is one of the most cited papers and least understood ideas. Originally, Reed and his advisor Saltzer had been collecting design principles from security experts from the NSA and IBM, but stressed that no one understood computer security at that time. In 1976, he shifted his focus to networking protocols and how they could be factored into layers, which itself generated much argument as to which features should go into each layer. Reed mentioned an early paper of Clark's, "The Design Philosophy of the DARPA Internet Protocols," which stressed the technique of multiplexing *existing* interconnected networks as a major design goal. Another paper Reed and Clark published, "An Introduction to Local Area Networks," also emphasized that a technological innovation is utilized in two stages. In the first, the innovation is used to improve the performance of what was already being done; the second stage is the discovery of new applications not conceived of beforehand. Finally, Saltzer, Clark, and Reed published the E2E paper, which identified a non-intuitive structure of some systems and presented an argument of what not to put in the core of the communication network.

Reed defined the argument abstractly and then said, more concretely, that secure message delivery can only be done at the endpoints, despite what networking companies tout as a secure network. He then said that a major confusion point is deciding what constitutes a function F and what constitutes an endpoint. Some examples include traffic management and capacity reservation, which could be done entirely in the network. F is a quality, property, or attribute of the network that is emergent, but not necessarily a property of all the parts. Security and reliability were identified as emergent because a system may be reliable despite an individual piece being insecure or unreliable. Reed believes that the E2E argument should really have been called End-to-End Argumentation, to carefully define such functions and avoid confusing them with techniques that designers want in their networks or products.

The talk then moved to some earlier publications that picked up the E2E idea. One notable example was Lessig's article in *The New Republic* that placed the E2E argument into a legal domain and introduced new concepts like "network neutrality." Reed also described how the E2E argument was similar to the financial theory term, Real Options, which suggests one should delay making decisions that limit options, thus preserving those options for the future. He then noted that this introduces a design trade-off of preserving options versus optimizing. Leaving a system unoptimized introduces uncertainty, but is not a problem if it is built into the design. Security for example, deals with uncertainty as much as it does threats.

Reed then touched on some of the controversies around the E2E argument. In *The Future of the Internet and How to Stop It*, Zittrain calls for abandoning or modifying E2E arguments if the Internet is to be secure, robust, and safe; E2E lets the users control the Internet, and the unity of the network enables real-time sensing and dissemination of users' information. Reed notes that these arguments have a compelling meaning to them, but they are not compelling enough to change the design principle. In response to Clark and Blumenthal's "Rethinking the Design of the Internet," which says that policy requirements that employ CALEA-like rules and spam blocking are not compatible with E2E, Reed questioned whether the techniques used to address the issue were right in the first place.

In closing, Reed reiterated that design principles survive because they make use of clear systematic reasoning. Such principles are neither gospel nor prime directive, but a pattern to reason by. He repeated that the E2E principle helps to manage uncertainties by dealing with how functions should be implemented and that we should not confuse functions with the techniques, features, or capabilities for achieving that function.

Ron Rivest from MIT noted that the E2E idea presupposes that one can implement things correctly, but most people cannot implement security right. If there are attacks on incorrectly built communication networks, where should the complexity be designed? Reed replied by questioning the wisdom of modularity, saying that we often confuse ideal properties with a module itself or the specification with the chip implementing it. Thus, the problem is a logical issue, by which we map the model to the object, and is not an issue with the E2E principle. Ben Norrik from Google asked Reed to define an endpoint; Reed answered that it is inherent in the design of what you are building.

Another audience member asked what Reed thought of nation states that dislike the Internet's inability to be controlled precisely because that function is not in the network. Reed mentioned that some aspects of the network came from the need for a globally addressable scheme for all participants. What these nations do is form their own private Internets, much as companies create private networks. An attendee pointed out that Reed suggested that security is not something to build into the network and asked whether Reed felt putting ACLs into an OS kernel was a design error. Reed said he disagreed with his co-authors that it was practical to design a secure kernel. They had originally been tasked by the military to build a kernel that functioned like a network, which passes messages from process to process and respected a multi-level security lattice. However, such a kernel was of no military value, because military operators frequently declassify messages in the field and thus break their own requirements to be practical. Ultimately, they learned that the specification was extremely flawed and had they applied the E2E argument to kernels, they would have realized they could not build what they needed into it.