conference reports

THANKS TO THE SUMMARIZERS

Alex Boster
Ali R. Butt
Stefan Büttcher
Laura Carriere
Marc Chiarini
Timothy Denehy
Matt Disney
Rik Farrow
Chaos Golubitsky
Kevin Greenan
Roman Valls Guimera
Abhishek Gupta
Jin Liang
KyoungSoo Park
Charles Perkins
Kristal Pollack
Florentina Popovici
Vijayan Prabhakaran
Josh Simon
Aameek Singh
Shafeeq Sinnamohideens
Matthew Wachs

CONTENTS OF SUMMARIES

LISA '05: 19th Large Installation System Administration Conference

San Diego, CA
December 4–9, 2005

Keynote Address

SCALING SEARCH BEYOND THE PUBLIC WEB

Qi Lu, Vice President of Engineering, Yahoo! Inc.

Summarized by Roman Valls Guimera

Qi Lu told us the challenges that Yahoo is facing to adapt their infrastructure to a new search level: personal and social search. These new forms of searching, as opposed to the traditional and well-known public search, are really difficult to scale.Take the example of Yahoo Mail: gigabytes of personal mail that cannot be lost under any circumstances. Hence, a personal space should provide high levels of fault tolerance, replication, and data-partitioning schemes. One thing is clear here: it's really complex to achieve all of them when you have a massive number of users. Without going into details, the Yahoo approach to solving those issues is cleverly simple, analogous to a biological cell: when the data cell grows, it divides and replicates itself throughout the system, keeping the properties we've seen before (redundancy and fault tolerance). Of course this process runs unattended, but it can be monitored in real time.

Without leaving the infrastructure point of view, we need to think about new ways to relate data from different users without losing performance or search quality. As del.icio.us does, a community of friends improves user search, and when there's a critical mass of users, we can improve the quality of results for a lot of users.

You can check Qi Lu's personal 360º Yahoo space for more info: http://360.yahoo.com/profile-dHFl7togcqomOrUGtvI-

CONFIGURATION MANAGEMENT WORKSHOP

Moderator: Paul Anderson

Summarized by Matt Disney

Based on a completely unscientific survey, the odds are high that you do not use a configuration management (confmgt) tool for managing systems. And if you do use a confmgt tool, you probably wrote it yourself (despite the availability of a small number of other confmgt tools) and that nobody else uses it. Why? What are you seeking in a confmgt tool? Are you ready for systematic management of your systems? Is it possible to create a confmgt tool that will be accepted by a majority of system administrators?

The confmgt community asked itself these questions, and many others, at the LISA '05 Configuration Management workshop. The unscientific survey mentioned above was taken at this year's workshop, a gathering of system administrators, researchers, and tool developers interested in the challenge of confmgt.

By some accounts, confmgt problems are characterized by the lack of popular adoption of confmgt tools. Some attendees, while not entirely unconcerned about adoption, are principally concerned with the underlying theory. They believe a solid foundation will yield tools that are attractive and, more important, correct according to certain metrics. Although the differing priorities of these two groups are not necessarily mutually exclusive, the workshop next year will likely be divided into the two categories of tools and theory.

One popular topic this year was the prospect of an OSI-like layered model for confmgt, which could facilitate the progress of tools as well as represent the boundaries between tools so that developers can focus on specific challenges. Such a model emerged from that discussion:

5. Service level goals. Example: .5 second response time for service X.
4. Invariants. Example: port numbers.
3. Services. Example: IMAP service.
2. Configurable elements. Examples: users, groups, resolvable hosts.
1. OS API. Examples: file contents, process memory state.

That definition led to an exploration of related issues, such as the general notion of feedback among the layers and the prospect of suboptimal restrictions potentially inherent in such a framework.

The challenge of federated confmgt was also covered. Existing tools do not reflect the complex political structure of large organizations. Some suggested methods for addressing this included combining abstraction and delegation, separation by infrastructure ownership, and the separation of functional administrative domains.

Andrea Westerinen of Cisco gave a presentation about the Common Information Model (CIM) and helped the attendees frame ways in which it might be used in the context of confmgt. Increased attention to a well-defined and popular, if not technically standard, model for describing system objects could be important and useful to confmgt in the future. Some tools already use CIM to some extent.

Tom Limoncelli also joined the workshop with a presentation from an outsider's perspective. Entitled "What I've Learned from Avoiding Configuration Management," his talk included some tips on how the core confmgt group could do a better job of connecting with the greater system administration community.

While some themes for the workshop recurred this year and will undoubtedly continue to arise on mailing lists and future workshops, there is traction on some new ideas and a continued interest in both confmgt tool development and theory. For detailed workshop notes and general information, see http://homepages.inf.ed.ac.uk/group/lssconf/.

ADVANCED TOPICS WORKSHOP

Moderator: Adam Moskowitz
Summarized by Josh Simon

In answer to the question of how much system administration has changed in the past year, attendees at the Advanced Topics Workshop (businesses, including consultants, outnumbering universities by about 4 to 1), the general consensus was "not much" on a professional level, although various compliance issues (local and federal regulations on IT, including SOX) have affected many. There's an expectation that compliance will take up more of our time and budget. Furthermore, automation is becoming a more obvious necessity to more people; folks are learning that scale, especially with clustering, simply requires it. We also agreed that the so-called soft problems, such as user interaction and customer service, will increase. We noted that many of us seem to be leaving system administration–type roles for networking, security, and, in at least one case, company executive (CIO), and others are losing interest in pure SA-type work.

Next was a quick around-the-room for tools we've seen. Many people said "wiki"; other tools included cfengine and other configuration management tools, Google Earth, IM clients within and across workgroups, Nagios and monitoring tools in general, Ruby, System Installer Suite (SIS), VMware and other virtual machine tools, VNC, ILO, other Lights-Out Management (LOM) software, and Xen. Others mentioned methodologies for development and testing, and code reviews, or hardware tools such as label makers for cables and power-consumption monitoring.

After the morning break, we discussed security and some of the hardware VPN solutions—using security incidents as catalysts for change on both an organizational and a technical level—and when to allow exceptions to your mandated security policy. This segued into a discussion on compliance; two of the points someone stressed were that (1) there's no established case law for SOX, so the auditors get to define what compliance is, and (2) making the collection of reports (or at least data) for the auditors should be both automated and reproducible. This is much like ISO 9000 all over again in some places.

Our next discussion was on scaling and automation. You should never say, "We can do this stuff with less staff," but, rather, "We can do more stuff with the same staff," lest you lose budget. It's essential to plan for growth at the beginning, because you'll rarely get the opportunity to go back and fix it. Many places run homegrown systems (especially configuration management and automation), because there's no off-the-shelf software that does everything we want, and such products as there are tend to have a steep learning curve or cost. Furthermore, getting different single-OS groups to agree on a multi-platform product is hard, and some people fear losing their jobs to automation, as opposed to getting rid of the mundane tasks to focus on the more challenging.

We next discussed personal productivity tools, ranging from changing OS ("Mac OS X"), to documentation (more wikis), to simple command-line tools (vi, grep, glimpse), books, PDA-specific

applications, Web calendaring and sharing tools, sleeping pills (for ourselves, not our customers), unsubscribing from magazines and mailing lists, delegating to others, and even going to the gym.

After lunch, we briefly discussed autonomic computing, and how we as system administrators will interact with these self-modifying systems. In summary, it won't change what we do overnight, there'll be a cost/value tradeoff in outsourcing, and it'll probably be inappropriate for organizations with open-ended problem sets (such as research organizations and other places where the problems the systems are there to solve are not well contained or easily programmed).

Our next discussion was on professionalism and seniority. Some expressed concern that fewer institutions of higher education were offering courses specifically aimed towards system administrators, though others argued that as long as the candidates have thinking and problem-solving skills, that (plus experience as needed) was sufficient. Some concerns were raised about forthcoming regulation of IT personnel as an industry; between compliance issues such as HIPAA and SOX and the issues caused by not patching systems regularly, several people predict that regulation is coming sooner rather than later. The usual analogies were mentioned: are we doctors, are we janitors, or are we on a spectrum like the electricians/electrical engineers?

We next named tools we thought we'd need to learn in the next year or so. Answers included concepts from AJAX to ZFS, along with new operating systems for people (Mac OS X, Solaris 10, and Windows), and the usual suspects (documentation and knowledge management, process management, project planning, and virtualization).

We next discussed storage and efficiency, followed by network versus system administration. Some argue that netadmin is five to ten years behind sysadmin; others argue the reverse. The consensus seems to be somewhere in the middle and depends a lot on how you define your terms. For example, it's harder to have the Internet in your test lab, and routers and switches tend to make changes immediately rather than "when you reboot" or "when you send a signal to a process," as with systems. There was also a discussion about the relative security models for systems (data) and networks (keys to the kingdom).

Finally, we discussed physical plant issues (power, cooling, weight, and remote access) and social technologies. Most places are using some form of wiki or other documentation and collaboration software; many are using some form of instant-messaging client. One novel approach taken by some places is to use podcasting for information broadcasts.

Technical Sessions

VULNERABILITIES

Summarized by Roman Valls Guimera

- ***GULP: A Unified Logging Architecture for Authentication Data***

  Matt Selsky and Daniel Medina, Columbia University

  GULP (Grand Unified Logging Project), a distributed approach to logging centralization, was born from the difficulties managing logging info at Columbia University: lots of servers saved different logs on local disk with unrelated informationmaking searches and correlation painful.

  GULP aims to solve that problem by applying custom XML templates to the log files and extracting the interesting information from them. When validated, this data is stored on a MySQL database. Now the security team can construct queries to solve their problems: find stolen laptops, missing people, owners of infected machines; confirm stolen accounts; etc.

  http://www.columbia.edu/acis/networks/advanced/gulp

- ***Toward an Automated Vulnerability Comparison of Open Source IMAP Servers***

  Chaos Golubitsky, Carnegie Mellon University

  Awarded Best Student Paper!

  Chaos Golubitsky presented a way to measure the attackability of code. That is, the relation between not commonly accessed code (which the standard user is not supposed to reach) and the code that is accessed under normal circumstances can be expressed in the following weighted formula:

  $$\text{attackability(codebase)} = \sum_f \text{functionsweight}(\text{priv}(f)) \cdot \text{weight}(\text{access}(f))$$

  She applied this to UW IMAP, Cyrus-IMAP, and Courier IMAP. Using a code analysis tool called cflow (http://www.gnu.org/software/cflow/), she managed to split privileged code functions from the user-accessible ones and applied the above weighted formula.

  The winner was Courier-IMAP, because it's designed to have a good privilege separation, while UW and Cyrus were tied.

  If you want more information on this presentation, please see http://www.glassonion.org/projects/imap-attack/slides.pdf.

- ***Fast User-Mode Rootkit Scanner for the Enterprise***

  Ti-Min Wang and Doug Weck, Microsoft Research

  Almost any enterprise or user who uses Microsoft Windows will eventually be infected by malware. Tools such as Ad-Aware perform quite well to wipe out the adware,

trojans, and viruses that infect Windows machines.

Unfortunately, a new form of malware has appeared on the scene: ghostware. Ghostware evades any attempt to clean the system if you use current utilities. It does so by intercepting the API calls, which is just a step away from owning the whole OS. In other words, ghostware cannot be detected from inside the infected machine because it has kidnapped the OS itself, and that "ghost program" responds to the other programs by lying when asked for its presence.

The main concept behind Strider GhostBuster is the cross-view diff approach. Forget about the normal time diff (standard diff) we all know. Cross-view is a diff between what we see *inside* the infected machine, and what we see *outside* of it, so we can see the lie and the truth at the same time. We can then erase the ghost(s): it takes just seconds to see the liar.

http://research.microsoft.com/csm/strider

INVITED TALK

Summarized by Charles Perkins

■ *Network Black Ops: Extracting Unexpected Functionality from Existing Networks*

Dan Kaminsky, DoxPara Research

Introduced as a "white hat" hacker, Dan Kaminsky presented practical and, in many cases, real-time exploits of network and cryptographic protocol weaknesses or unintended behaviors.

The MD5 hash function is broken both in theory and in practice. Dan demonstrated how an unsafe hash (which can be found in about 45 minutes) can be used to create two pages that hash to the same value. Key to the demonstration are that Web pages accept garbage and that you can present Web content programmatically.

Dan then described how for the receiver, keeping track of IP fragments turns a stateless protocol into a stateful one, and that IP fragmentation makes IDS harder. While attention to this has resolved many of the issues, timing attacks remain a problem. When an intrusion protection system operates upstream of a protected host, differences in fragment expiration timing between the host and the IDS can be exploited. A stream of fragments can be created by an attacker such that the IDS will construct a different packet from the fragments than the supposedly protected host will. Dan then described the temporal attack in detail.

Some firewalls, intrusion protection systems, and intrusion detection systems attempt to mask their existence. Dan listed a number of existing packet behaviors, responses, and contents that will reveal the existence of even "transparent" defenses; IPv6 will be even easier to fingerprint, due to encapsulation and reassembly issues.

Dan asserted that IPSes should not insert rules to ban traffic from hosts or networks after receiving invalid, excessive, or anomalous traffic. Simplistic rules will result in banning important services (such as root DNS servers), but, more important, through DNS poisoning an attacker could subvert your infrastructure and use your own rules against you.

Dan next described his project of probing the Internet DNS infrastructure, which he performed using copious bandwidth and novel techniques, including requesting the addresses of dynamically generated names satisfiable only by his DNS servers. Of 9 million nameservers scanned, 2.5 million do recursion; 230,000 forward to Bind8, which is a security problem; and 13,000 have the precise configuration that caused trouble for Google. Dan's resulting data set is quite large, and most interrelationships among nameservers are one hop deep (40,000 are connected graphs that are two hops deep—e.g., ask alice, get a request from bob).

As a result of his study, when the Sony Rootkit was exposed Dan already had a list of all the nameservers in the world and was able to use his tools to get an understanding of the breadth of the rootkit's distribution. It connects to connected.sonymusic.com, and that requires a DNS lookup which goes into the nameserver's cache. Dan performed a scan requesting connected.sonymusic.com of each of the nameservers without recursion. Nameservers that were able to respond with the IP address therefore had already been queried for it. Dan found 556,000 hosts with Sony-linked names. Dan acknowledged the margin of error in the survey due to time-to-live filters, some nodes recursing anyway, etc. Dan was interested to find indications that more nodes were trying to uninstall the rootkit (based on a different Sony domain name) than had gotten the rootkit in the first place.

Dan then showed graphs of the DNS server relationships, animations of router source-destination pairs, and a 65KB/sec video stream encapsulated in and delivered over DNS replies from an outside host.

CONFIGURATION MANAGEMENT THEORY

Summarized by Marc Chiarini

■ *Configuration Tools: Working Together*

Paul Anderson and Edmund Smith, University of Edinburgh

Anderson took a look at the current state of system configuration tools, outlined why there are no clear successes, and made some simple suggestions for improving the technology. Configuration

management needs to be viewed as a continuum, and we are just beginning to understand how to translate from high-level goals to the best low-level network and machine configurations to achieve those goals. This understanding will be facilitated by moving toward a common, generalized framework that represents distinct layers in the continuum and standard means for transforming data between layers.

Anderso focused on generic, semantically unaware operations for the deployment and management of configuration data. First of these are classing operations, which, as implemented in many current tools, cannot easily handle conflicts, such as those that may occur when multiple inheritance is in effect, and do not effectively address cross-cutting concerns. One way in which to shore up the first of these drawbacks is to implement powerful mechanisms for constraining subclasses and prioritizing inherited values. The second type of operation, aggregation, involves the (semi-)automatic creation of server configurations based on the needs of the client. The advantages of aggregation include a reduction both in the time required for manual specification and in the number of configuration errors. Sequencing and planning operations that enforce user-defined invariants in a declarative environment will be integral to any effective configuration tool. Finally, Anderson delivered a convincing argument that delegation and authorization should become multi-valued in order to make meaningful distinctions among required services.

We do not need a common system configuration lexicon or a strictly enforced operational architecture. Rather, we require a data structure for information exchange in the continuum and between independent tools, a "library" of generic operations for configuration data manipulation, and a simple interface for performing these operations. During the Q&A, someone asked about the lack of clear guiding theories, standards, and leaders in the configuration management space, and whether clarity is required to move forward. Anderson replied that arriving at high-level de facto standards will very likely happen naturally.

- ***A Case Study in Configuration Management Deployment***

  Narayan Desai, Rick Bradshaw, Scott Matott, Sandra Bittner, Susan Coghlan, Rémy Evard, Cory Lueninghoener, Ti Leggett, John-Paul Navarro, Gene Rackow, Craig Stacey and Tisha Stacey, Argonne National Library

  Narayan presented a case study based on the rollout of the BCFG2 configuration management tool developed at ANL. The talk focused on the human aspects of CM tool adoption, which have not been extensively researched. Narayan began by stating that CM tools are not widely used and posited a reason: the upside is not well understood. The reason his division wanted to deploy a tool was because they were experiencing serious configuration problems (change propagation issues, patching, etc.) due to many years of ad hoc management. He described a two-year timeline of in-house events that began with the development of BCFG1 (and the eventual realization that it was a miserable failure) and culminated in the successful deployment of BCFG2. Narayan went on to present a retrospective analysis of key discussions within his group and how they arrived at their success.

  Among the many issues addressed by the team, four stood out: tool fitness, group consensus, initial buy-in, and group dynamics. An effective approach was to give admins whitebox access, address their technical questions as quickly as possible, and take their input seriously. Not surprisingly, this also helped in reaching group consensus. Narayan pointed out that this consensus was built by increasing each person's familiarity with BCFG2 and implementing critical features. Communication was hard, since individual assessments of the tool embedded strong personal beliefs, and confidence in the tool varied over time.

  The authors make four recommendations for helping to get a high-impact tool adopted at one's site. First, the tool needs an evangelist. This person consistently touts the prospective benefits of the tool and remains optimistic, but does not ignore complaints. Second, the audience must be shown a short-term payoff. Third, every effort must be made to address the concerns of the users (system administrators), whose instincts are usually correct. Try to incorporate in minor revisions those suggestions that make sense for the tool in the big picture. Lastly, try to keep everyone on the same page whenever possible. This may require sorcerer-like social skills.

  Narayan freely admits that they had several factors working in their favor. Their group already believed that new configuration management techniques were needed; their strongest advocate was also their primary toolsmith; and they had an amicable and highly interactive group from the start. Your mileage may vary.

- ***Reducing Downtime Due to System Maintenance and Upgrades***

  Shaya Potter and Jason Nieh, Columbia University

  Awarded Best Student Paper!

  Shaya Potter mentioned a few well-known reasons why managing computer systems is hard work: software is buggy, hardware suffers from various faults, security can be compromised, and forcing downtime to upgrade or patch for any reason will usually annoy

users. Common approaches to mitigating the impact of such events include the replication of services, OS-based isolation (such as FreeBSD Jail and Solaris Zones), and hardware virtualization using true VMMs like Xen and VMware. The first of these is only useful for shorter-term transactions such as Web requests and is very difficult to implement for longer-term stateful services such as user desktops. OS-based isolation suffers from serious limitations on the types of applications that can be run and may also require extensive non-modular kernel modifications. Lastly, the biggest drawback to heavyweight VMMs is that they still require tight coupling with the underlying OS, making migration costly and inflexible.

The AutoPod system provides secure, virtual private environments (PODs) in which a multitude of processes can execute normally with minor restrictions: a lightweight virtualization layer is installed on a host OS (currently only Linux) via kernel module. The virtualization layer intercepts and potentially rewrites all system-call communication between processes and the real kernel. AutoPod also features a facility to migrate whole PODs across machines or even virtual machines running different OS kernels.

When considering the initial design of AutoPod, the authors identified several hurdles. Most existing applications are not designed to migrate between computers, primarily because their running images are coupled to a specific instance of an OS. Clearly, it is not feasible to rewrite all applications of interest. In conjunction with virtual namespaces, this hurdle is overcome by briefly stopping all POD processes, recording important high-level state information for each process, translating into an efficient intermediate representation, transferring the process state and POD-specific info to a POD on an alternate machine, and restarting the processes where they left off. Another hurdle that needed to be cleared was the isolation of processes for security purposes. In particular, processes running with super privileges are rarely restricted by an underlying OS.

Questions were asked about transferring network state, especially long-lived connections. Potter responded that AutoPod can handle most situations. In some cases, however, such as when a Web server is migrated to another system with a running Web server, an external proxy must be in place to redirect requests to the correct virtual port. Another questioner asked how AutoPod compared to VMware's Vmotion, a migration facility for entire virtual machines. The difference is primarily in the speed with which a migration can be performed (especially for fully loaded VMs) and the limitations on kernel variations.

INVITED TALKS

What Big Sites Can Learn from Little Sites

Tom Limoncelli, Cibernet Corp.
Summarized by Alex Boster

Tom Limoncelli gave a relatively high-level talk about lessons he has learned turning about the IT department of a small site. He began with "why things aren't getting better." Using a pyramid diagram, Tom illustrated the earlier state of IT with a small number of "Good IT" sites at the top and a large number of "Bad IT" sites at the bottom. The state of IT today was illustrated with the same pyramid with a much larger base labeled "Really Bad IT." This was, he asserted, the result of the proliferation of small sites with "small sysadmin" attitude and abilities. However, he asserted that small sites are important because (1) they become big and (2) most big sites are really federations of small sites. These "broken" sites, he said, slack on the fundamentals.

Tom then asked, "Are best practices the solution?" He made an analogy between electricians versus electrical engineers: a construction project stops rather than do something "not up to code." He claimed that what's missing from this analogy in IT is an inspector who signs off on a project. The overall state of best practices is very fragmented: vendor's recommendations, SAGE and LISA publications and tutorials, CMM for sysadmins. Tom made special note of applying Maslow's "hierarchy of need" from the field of psychology to IT users as a good practice.

Finally, he presented his lessons from rebuilding a small site. The first lesson was that, at first, he only had time to deal with the basics, and, furthermore, "being there" crystallized what those basics were. Tom presented his experience in phases. Phase 0, acclimation, was where he learned who the players were and dealt with emergencies. In Phase 1, basic stability, the goals were to make the most important services reliable, establish backup procedures, learn the corporate purchasing process, and replace "accidents of history" design decisions. He emphasized the importance of the email service, meeting with users, a rudimentary documentation repository, and physically labeling everything he touched. Then in Phase 2, he could move on to establish basic IT applications: ticket tracking, network monitoring, documentation wiki, remote access, and automating backups.

Questioners asked about the size of the small company (100 users). In response to a query about backups, Tom stated that he started with rsync and Retrospect and has since moved to Bru apps. This was followed by a back-and-forth about fixing sites that, once stable, can be outsourced.

■ ***Building MIT's Data Center: An IT Perspective***

Garrett Wollman, Infrastructure Group, MIT Computer Science & AI Laboratory

Summarized by Charles Perkins

IT infrastructure was not considered early in the design process for the $300 million CSAIL building, which, at the time of the initial planning for the new building, contained four IT labs with independent IT staff.

Garrett outlined the differences among residential, commercial, and institutional architecture. Institutional architecture usually ends up being one-off construction. This new building had to shelter 1,000 people and meet the needs of 150 faculty, 50 frozen monkeys, four IT organizations, three lecture halls, and three wealthy donors, while reflecting the artistic vision of a well-known architect. Garrett and his team, the Net32 committee representing the computing labs, were brought into the project six years in, well after most of the physical parameters had been set and budget and space had been allocated.

The Net32 committee quickly determined that several misconceptions by management had resulted in a woefully inadequate allocation of space and access for IT infrastructure, including: (1) Computers are smaller and need less space than they used to, never mind that the computing clusters are growing by leaps and bounds. (2) Switches are $50 . . . manageability? What? Why? (3) You can just move the racks, switches, UPSes, power supplies, and all of the rest of the infrastructure over from the old building . . . except that the old system has to stay up and be usable while the move is taking place. (4) The building AC in the ceiling is good enough, and the IT staff doesn't need to monitor the HVAC independently of the facilities people . . . although in the past it has always been the IT staff telling the facilities people that the AC is broken and the computers are overheating. (5) Conventional quad power outlets in the floor will be fine.

The Net32 committee wanted all new racks with room-wide UPS power, under-floor AC with humidity control, power and network pre-wired, SNMP monitoring of the UPS and HVAC, and accessible cable-trays throughout the building for easy network changes.

A compromise was reached: some smaller spaces were coalesced into an irregular larger space and the group got all new racks, roomwide UPS, under-floor AC without humidity control (as the water pipes for chilling had not been designed into the building), power and network partially pre-wired, and separate proprietary UPS and HVAC monitoring.

Lessons learned: You can avoid a great deal of pain by getting involved in the planning early: make sure that management knows what IT costs, get closets, watch your wiring contractors like a hawk, get complete drawings, give complete requirements, think about where office infrastructure goes (printers, etc.), pre-wiring is great, play hardball with vendors, get freebies for naming things after vendors, hold coordination meetings after lunch instead of during lunch, and raised floors outside of machine rooms will make you sad.

CONFIGURATION MANAGEMENT PRACTICE

Summarized by Roman Valls Guimera

■ ***Integration of MacOS X Devices into a Centrally Managed UNIX Environment***

Anton Scultschik, ETH Zürich

Software management has always been complicated, especially on large, shared UNIX environments. Even with the help of package management tools, the admin has to deal with system diversity.

Template tree 2 helps to ease that diversity by providing modularized, self-isolated, meaningful configuration entities. This approach combined with SEPP package manager, which allows on-the-fly software provisioning (using automount), simplifies the daunting task of installing and updating software.

Template tree: http://isg.ee.ethz.ch/tools/tetre2/

SEPP: http://www.sepp.ee.ethz.ch/

■ ***RegColl: Centralized Registry Framework for Infrastructure System Management***

Brent ByungHoon Kang, Vikram Sharma, and Pratik Thanki, University of North Carolina

Managing large networks of Windows clients can be a daunting task: users tend to install their own programs (if they have the privileges to do so), and with those changes eventually comes breakage of their workstation.

Regcoll allows a system administrator to monitor Windows registry changes the same way a revision control system does, but with a real-time feature. If the user complains about a system malfunction, by using regcoll the system administration can revert the offending changes and go back to a state known to be fully operative.

In addition, regcoll can be used as a monitoring tool and a security analysis and auditing framework. To sum up, regcoll helps you keep your computer park free from unexpected failures caused by third-party software and/or user intervention.

■ ***Herding Cats: Managing a Mobile UNIX Platform***

Wout Mertens and Maarten Thibaut, Cisco Systems, Inc.

Users of laptops behave as if the laptops are their property; they will customize them, install pro-

grams, change default configurations, etc. As a result, the task of keeping those systems updated and clean becomes really difficult for the administrator or help-desk support staff.

Maarten and Wout solved the problem using Mac OS X as the preferred platform (while also supporting others). They use radmind plus their own additions to distribute software updates efficiently, with a pleasant interface on the user side, and, most important, safely (users need their laptops to always be operative). Additionally, they've made backup scripts to keep clients' data safe on a server and configured FileVault (a ciphered file system) properly to ensure users' privacy. (They've also used their own automated scripts to manage the process of issuing client SSL certificates!)

They deployed all these features quite successfully and, more important, usefully and painlessly.

radmind: http://sourceforge.net/projects/radmind

backup software: http://rsug.itd.umich.edu/software/radmind/contrib/LISA05/TacSync.tar.gz

INVITED TALKS

Under 200: Applying IS Best Practices to Small Companies

Strata R. Chalup, Virtual.Net, Inc.

Summarized by Alex Boster

Chalup's talk examined the question, "What of the big company practices can be applied to small companies?" As smaller companies grow to 50–70 people, staff moves on or the junior IS staff does not know how to handle the larger site.

She implored listeners to eschew the term "IT" in favor of "IS," since the ultimate goal of the job is to provide a service, not just the technology itself. This is part of an overall attitude adjustment required of most IT shop patterns.

Chalup's specific recommendations included: control access (widespread root access causes chaos); standardize and modularize everything you touch; have a standard plan for debugging issues; build a knowledge base; make full use of email lists; and use change control everywhere. She also discussed the importance of having written policies published on the intranet. She placed great emphasis on using a ticketing system with built-in metrics for all IS tasks. Proper ticketing system priorities were mentioned.

There was a question about what to do to keep users from walking up to your desk if you don't have a door to close. She stated that she's seen yellow police tape used in place of a door to good effect. A discussion then took place about ticketing systems. Chalup also noted the importance of learning how to get the information you need out of a user.

What's a PKI, Why Would I Want One, and How Should It Be Designed?

Radia Perlman, Sun Microsystems Laboratories

Summarized by Charles Perkins

Radia showed the usefulness of public key–based systems for authentication and authorization, as compared to symmetric key encryption. She described problems with current models (the monopoly of Verisign or oligarchy of self-signed certificates in browsers vs. the anarchy of PGP) and then outlined a model that avoids the concentration of trust inherent in the first two while addressing the scalability issues of the third.

Participants in encryption systems need to get their keys from somewhere. If each participant (n) required a shared secret for each other participant it might need to talk to, n^2 keys would need to be configured. In shared secret systems, such as Kerberos and Windows NT domains, the n^2 requirement is relaxed by using central servers to hold secret keys for participants (e.g., users' workstations and the services that they connect to). The only initial shared secret required is that which allows the participant to talk to the KDC or domain controller.

Public key encryption also requires key distribution, because participants need to get the public keys of their intended destinations from somewhere. The certificate authority is the equivalent of the KDC or domain controller in a Public Key Infrastructure. A certificate authority has significant advantages over its private key equivalent: a KDC is less secure, contains a highly sensitive database, must be online, and must be replicated. The CA, on the other hand, may be offline. Revocation makes CAs harder to implement, however.

Radia asked, "What can I do with PKI?" and answered: establish secure conversation without online introduction service, send encrypted email, send signed email widely, distribute signed content and single sign-on to mutually distrustful sites. Radia doesn't believe we can avoid names in a PKI.

Radia then explored how PKI with access control lists can create a scalable system for revocable granting of permission to resources. The system allows resources to require membership in groups, with the groups nested in hierarchies. On an access attempt the group server will (1) sign a certificate vouching that an identity is a member of that group or (2) require the client to walk up and/or down the tree acquiring proof of membership in sub- and/or super-groups in order to prove membership in the group the resource requires. Proven membership certificates, which may be timestamped, may be cached by the client, and revoca-

tion is provided for by allowing the resource requiring the certificate to accept only recently minted certificates.

There are three models of PKI widely used today:

1. The Monopoly model, whereby Verisign signs all the certificates, which is easy, understandable, vulnerable to monopoly pricing, introduces vulnerabilities getting the certificate from a remote organization, is dependent on Verisign's key never changing, and requires the security of the world to depend on the honesty and competence of one organization forever.

2. The Oligarchy model, used by Web browsers, wherein 80 or so self-signed certificates are implicitly trusted, which allows users to add to or delete from the set of certificates, eliminates monopoly pricing, is less secure (any of the 80 keys may be compromised), and makes it impractical to check the trust anchors.

3. The Anarchy model, used by PGP, wherein anyone may sign a certificate for anyone else; users consciously configure starting keys; proof of identity is inferred from traversing chains of trust, which does not scale as the number of certificates grows and it becomes computationally difficult to find a path; there is no practical way to tell if a path should be trusted; and there is too much work and too many decisions for the user.

Trust in a CA should not be binary; a CA should only be trusted for certain things, and a name-based system seems to make sense.

Radia proposes a bottom-up hierarchical model where each arc in a name tree has a parent cert (up) and child certs (down). The namespace has a CA for each node and lookups don't start at the root—they start at the member's group CA and go up to the least common ancestor. Cross-links are allowed, and this system allows organizations to choose top-level cross-link services. Importantly, the organization can revoke the up-certificate to one cross-linking service and select another if it is unhappy with the service. In intranets, no outside organization is required, inside security is controlled from the inside, and no single compromised key requires massive reconfiguration. A uniform PKI policy across all participants is not required.

Asked why we don't have elliptic curves in all this stuff, Radia replied that the patent situation around elliptic curves is unclear. Also, using the RSI private key is slow, but using the public key is fast. Verifying a certificate using RSI might actually be faster than using elliptic curves.

A concern was raised that fast factoring might make the PKI infrastructure obsolete. Radia conceded that it could happen. However, a fundamental concept of cryptography is to pick a problem mathematicians have been working on for a long time, meaning, hopefully, that it is a hard problem. She predicted that quantum crypto hardware might be able to factor the number "15" in a few years!

She was asked if the bottom-up PKI architecture described in her talk was in the book she co-authored (*Network Security: Private Communication in a Public World*, 2nd ed.). She replied that it was.

■ ***Modern Trends in UNIX and Linux Infrastructure Management***

Andrew Cowie, Operational Dynamics
Summarized by Laura Carriere

Andrew Cowie delivered a thought-provoking session, postulating that the profession of system administration continues to follow numerous divergent paths when solving new problems and does not appear to be converging on a set of standard solutions to these problems. He stated that it was unusual for an industry to fail to converge on standards by this stage in its development.

Cowie observed that system administrators are being asked to solve increasingly complex problems with static or reduced resources and that there are frequently two schools of thought on how to solve these problems. Our profession seems to cycle between the options and often chooses to apply the wrong solution to a given situation.

Cowie gave a number of examples to support his hypothesis. He first addressed the issue of when to scale vertically (using a few powerful systems) and when to scale horizontally (using many small systems), stating there's no consensus within the industry on the criteria to be used when making such decisions. The end result is that many companies choose the wrong solution.

He discussed the related issue of server consolidation versus increasing complexity. A reasonable solution to limited floor space is to consolidate services onto a single UNIX system. However, a conflicting trend is to isolate services on separate servers, which simplifies the administration required to load, deploy, tune, and ghost. The end result is that organizations may be reducing or increasing the number of systems, or, possibly, following both trends at once.

The issue of using multiple blade servers versus moving to virtualization is a similar problem. Multiple small boxes provide plenty of resources but are a management nightmare. Putting multiple virtual systems on one powerful box works well until the virtual systems overuse one resource, thereby creating a bottleneck (which is frequently the I/O system).

Additional conflicting themes discussed by Cowie included Web interfaces without a command line interface, which make it impossible to write management scripts. The irony is that Web interfaces are designed to simplify management but ultimately prevent the best mechanism we have to do that—automation.

Cowie went on to consider desktop deployment. Although vendors have developed tools such as JumpStart and KickStart to automate installation, maintenance is difficult, and vendors are not providing solutions for that, the only exception being RedHat Satellite Servers.

Configuration management (CM) also has two competing approaches—convergence and congruence. Cowie cited cfengine as an example of convergent configuration management, where desired lines are added to the configuration files if they are missing. With a congruent CM system, entire configuration files are regenerated. The industry currently has no guidelines to determine which solution best fits a situation. Cowie briefly discussed the idea of encapsulation, an OO approach to CM that allows the administrator to specify policy (i.e., SwitchToPHP) and let the software do the required configuration.

Cowie concluded with a warning that Grid computing is coming and will radically change the industry. Again there are two competing approaches, a tightly linked cluster with shared memory, such as an SGI predicting the weather, and an aggregate of individually maintained systems, such as the systems that comprise SETI@home. He expressed his concern that Grid computing will drive the development of effective management tools and that this will threaten the livelihood of the junior sysadmin who enjoys repetitive tasks. During the Q&A period, Cowie expanded on this, saying that change is good and more evolutionary solutions free us to do more interesting work.

- ***Incident Command for IT: What We Can Learn from the Fire Department***

  Brent Chapman, Great Circle Associates

  Summarized by Marc Chiarini

  Brent Chapman, a California Civil Air Patrol incident commander and local fire department volunteer, gave a talk about applying the principles of incident command in IT departments. An IC system is used by various public safety organizations (Coast Guard, local fire and police departments, FEMA) to coordinate themselves and communicate with other agencies in an efficient manner during major unplanned incidents. Often, many different individuals and organizations are involved, and there needs to be a structure to determine who is in charge and exactly what needs to be done. Brent gave several real-world examples (car accident, raging wildfires, total data-center power outage) to help the listeners understand the scale of situations that occur. He also stressed that IC can be applied to nonemergency situations, such as facility moves and major system/network upgrades.

  A typical ICS follows nine key principles:

  1. Maintain a modular and scalable organizational structure. There may be five "sections" or groups responsible for different tasks: a Command Section with a capable IC (incident commander) must always be available; a mandatory Operations Section executes plans to achieve command objectives and worries about the now; a Planning/Status Section collects and evaluates information needed to prepare action plans and tracks progress; a Logistics Section is responsible for obtaining all resources required to deal with an incident; an Admin/Finance Section, necessary for the largest and longest-running incidents, will track costs and administer procurements.

  2. Maintain a manageable span of control. Limit section sizes and grow the hierarchy as necessary.

  3. Maintain unity of command. A strict tree structure (each person has only one boss) facilitates communication and reduces freelancing.

  4. Transfers of responsibility must be explicit.

  5. Maintain clear, expedited communication. Use no shorthand or codes and speak directly to resources when possible.

  6. Keep action plans consolidated. Command maintains the top-level (preferably written) plan for the current operational period (hour, shift, day, etc.).

  7. Manage by objective. Tell subordinates what to do, not how to do it.

  8. Maintain comprehensive resource management. Track all assets and personnel. Establish a sign-in process and "report-to" site.

  9. Use designated incident facilities. Must always identify a Command Post (CP).

  Brent went on to give a compelling example of using ICS in the IT world. He presented the timeline of an IC response to a data center failure, including the creation of subgroups in Operations, an explicit transfer of responsibility, assignment of a liaison, and ongoing organizational restructuring.

  The talk ended with some important tips for implementing ICS effectively: initiate incident response as soon as possible, use ICS as a toolbox, keep things simple, and practice all the time with routine and pre-planned events.

More info can be found at http://www.greatcircle.com/blog.

During the Q&A, David Blank-Edelman asked how people stay updated in the field. Brent recommended wikis, bulletin boards, top-down word-of-mouth, and whiteboards and Post-Its for areas without power. John Millard mentioned having standardized ICS kits ready for immediate use. I asked whether there are any standard metrics for judging the efficiency of a response. Brent replied that a good way to do this is follow the paper trail and do not get emotional when reviewing performance.

THEORY

Summarized by Marc Chiarini

■ *Toward a Cost Model for System Administration*

Alva Couch, Ning Wu and Hengky Susanto, Tufts University

Awarded Best Paper!

Alva Couch presented a novel first step in approximating the costs of system administration. System administration incurs both tangible and intangible costs; the former, as described in Patterson's cost model (LISA '02), tend to result in financial or productivity losses. The latter are much more difficult to measure, but an appropriate model would allow organizations to assess and improve their current processes. To arrive at such a model, Couch's team combined queuing theory, risk analysis, and simulation with an analysis of 400+ days of request ticket data (obtained from Tufts' EECS support group).

At first glance, measuring time spent waiting seems like a daunting task. It is, however, possible to view it as a function of certain parameters (request arrival rate, service rate, number of workers, etc.). This naturally leads one into queuing theory. Couch demonstrated how viewing request arrivals from the appropriate height, removing outliers from the ticket data, and adjusting for daily work cycles can ultimately reveal Poisson processes. To estimate the expected service rate, it is possible to apply risk analysis to the decision trees used by system administrators to resolve requests.

After examining real data, the authors chose to simulate a trouble-ticketing environment with non-product behaviors. As Couch explained, the motivation behind this was to account for phenomena that cannot be analyzed effectively via queuing theory. The team found that running a system near absolute capacity will cause chaotic and utterly unpredictable increases in service wait times. The important point is that in order to be useful, the new cost model cannot be applied to networks on the edge of steady state. When the capacity to resolve standard requests comfortably exceeds load, however, estimating the cost of administrative practice by indirect methods such as risk analysis can be made much more accurate.

Some interesting points were clarified during the Q&A session. Mark Burgess asked whether the data had been overly massaged. Couch responded that it was within reasonable limits for obtaining a decent model of steady-state behavior and extracting inhomogeneous trends. On the service side, non-product (realistic) systems could be approximated by introducing interruptions into an ideal system and analyzed via perturbation theory. When Couch mentioned that the study of realistic systems suffered from lack of data, someone suggested that SAGE or LOPSA could volunteer data sets. Couch was ecstatic about this prospect and stressed the importance of anonymized submissions.

■ *Voluntary Cooperation in Pervasive Computing Services*

Mark Burgess and Kyrre Begnum, Oslo University College

Mark Burgess spoke of a worldwide move toward pervasive computing, with multiple decentralized services provided by individual actors implementing autonomous policies. The authors believe strongly that the sysadmin tasks of tomorrow must integrate ideas about this explosion of autonomy. Mark's "promise theory" provides a different risk model for service provision. Whereas modern services are driven by demand and the server and client trust each other almost implicitly, this new approach takes an individualistic view of how an actor protects its own resources and acquires those it needs. In a future with very limited resources, client demand will no longer be the governing factor; clients and servers will have to cooperate voluntarily to keep things humming. The focus of every transaction in promise theory is on minimizing the risk of the involved parties.

The authors demonstrate the strengths of their approach by implementing a proof-of-concept voluntary RPC mechanism in cfengine. They observe that cooperative agreements now become the key to eliminating unpredictability. As opposed to traditional services, the protocol does not enforce reliability. Actors learn over time the probability that their peers will deliver on their promises, and then fall into stable patterns. The protocol itself was analyzed and verified for correctness using Maude, a programming language for reasoning about temporal logic and proving certain properties. Combined with the POC, this analysis revealed several limitations: the mechanism for initial agreement is made out-of-band; there is no current means of reprisal for uncooperative actors;

and the protocol does not easily provide a HA environment.

An interesting question was asked by Alva Couch about the quandary of having to put a file system into the pervasive network. Mark answered that there does need to be an addressable superblock out there.

INVITED TALK

Automatic PC Desktop Management with Virtualization Technology

Monica Lam, Stanford University/ SkyBlue Technologies

Summarized by Alex Boster

Monica Lam's talk was about a new x86 PC virtualization system in its pre-alpha stage (details are available on itCasting.org). She started by describing their team's motivation: to allow end users to turn over management of their desktops to professionals by breaking old assumptions. Their solution, called itPlayer, solves issues of mobility, management, and security.

The itPlayer software is built on a small, bootable Linux system and VMware Player. itPlayer is placed on any bootable storage device, such as an SD card, micro drive, or iPod. The whole VM resides at a known place on the network but is cached locally—similar to the way virtual memory works. Changes can be written back over the network, giving the user an online backup of the system. The system can also run in disconnected mode, provided the local storage device is large enough to hold the entire image (e.g., a hard drive, but probably not an SD card).

According to Lam, itPlayer is fast if the local cache is good; is as easy to use as a television ("just turn it on"); cannot be lost—just grab a new copy from the network; has disconnected operation; and has low virtualization overhead. It's limited by what Linux device drivers are available, having no virtualization of advanced graphics, and the fact that the desktop must be USB-bootable.

This new system results in new assumptions: that the state of the computer is always backed up, and that hardware is interchangeable. Lam then compared this system to other ways of doing desktop management: stand-alone PCs, mainframes, and thin clients.

Lam addressed the issue of updates by pointing out that the image provider (an IT department, for example) can update an image. Upon reboot, the users of that image will simply swap in the new image blocks from the network and run the new image. She said that currently desktop customization is easy, and standardization is hard. Lam asserted that the itPlayer system reverses that arrangement.

The talk ended with a demo of itPlayer. A Windows XP SP1 image was booted, the backing store image was replaced with an updated image running SP2, and the itPlayer restarted into SP2 upon reboot.

Questions focused on licensing issues, which Lam addressed mostly by pointing out that there is lots of freely available software. This was followed by a discussion of practical difficulties in customizing itPlayer environments per user in a corporate setting.

NETWORK VISUALIZATION

Summarized by Charles Perkins

Visualizing NetFlows for Security at Line Speed: The SIFT Tool Suite

William Yurcik, NCSA

William Yurcik demonstrated Security Incident Fusion Tools, which leverages human ability to discern patterns in visual displays.

CANINE provides NetFlows interoperability by converting and anonymizing NetFlow events from many commercial formats. It performs multi-dimensional anonymization of fields to facilitate secure data sharing and it reads both Cisco unidirectional NetFlows and Argus bi-directional NetFlows (see http://security.ncsa.uiuc.edu/distribution/CanineDownLoad.html).

NVisionIP shows the user the state of the IP address space, with default configuration for a class-B range, in a single screen. Activity is displayed by address in a pixilated matrix, with subnets across the top and station addresses down the side. It provides for drilling down to graphical views of activity on subnets, sets of hosts, and a single machine (http://security.ncsa.uiuc.edu/distribution/NVisionIPDownLoad.html).

VisFlowConnect-IP shows who is connected to whom on the network in a parallel axis chart with an inside view and an inside/outside view of network traffic. One-to-many, many-to-one, scanning activity, and unusual connection behavior can be observed in real time on the parallel-axis views, and both drill-down functionality and a filter language are provided: http://security.ncsa.uiuc.edu/distribution/VisFlowConnectDownLoad.html).

Yurick completed his talk by pointing interested parties to the VizSEC community at http://www.ncassr.org/projects/sift/vizsec/ and http://www.ncassr.org/projects/sift/.

Question: How do the tools scale above a class B network? Answer: One would open different windows, one for each class B. Question: How much trouble is it to make the software handle different data sources? Answer: It takes hard work, some "bribing," and a clear understanding of the protocols and formats. Also, the software is going open source.

■ ***Interactive Traffic Analysis and Visualization with Wisconsin Netpy***

Cristian Estan and Garret Magin, University of Wisconsin, Madison

Cristian Estan described adding interactive drill-down and flexible analysis to real-time traffic monitoring of network traffic. The Hierarchical Heavy Hitter approach reports traffic that exceeds a threshold and can use subnets, ports, and routing table prefixes, as well as user-defined groupings as hierarchies with ACL-like rules.

Cristian demonstrated the advantage of real-time interactive drill-down to determine the cause of anomalous network behavior, with "heatmap" charts of sender/receiver pairs making network traffic hotspots visually apparent.

Analysis may be conducted through text, time series plots, bar charts, and bi-dimensional reports across hierarchies. The user can select the time interval, bytes, packets or flows, and filters to be applied. The software handles router sampling and can use a database or files.

The software will be open source and more information can be found at the Netpy home page, http://wail.cs.wisc.edu/netpy/.

■ ***NetViewer: A Network Traffic Visualization and Analysis Tool***

Seong Soo Kim and A.L. Narasimha Reddy, Texas A&M University

Seong Soo Kim presented the paper, demonstrating, producing, and analyzing video from captured packet header information in order to detect DoS, DDoS, and worm behavior in the network. He asserted that DDoS flows look like any other flow and require aggregate analysis.

NetViewer aggregates seconds of traffic header information in a concise data structure in order to compare sequential frames with image-processing algorithms. Variations in pixel intensity and movement indicate DoS, DDoS, and worms. He displayed representative sequences and showed characteristic visual patterns produced by network attacks.

NetViewer has been run on several university and ISP connections, and they found things that snort did not. NetViewer is not looking for known attacks, is generic, is real-time with latencies of a few seconds, is simple enough to be implemented inline, and has a Windows and a UNIX GUI.

Email seongsoo1.kim@samsung.com or reddy@ece.tamu.edu for more information.

INVITED TALKS

■ ***Internet Counter-Intelligence: Offense and Defense***

Lance Cottrell, Founder, President, and Chief Scientist, Anonymizer, Inc.

Summarized by Alex Boster

Lance Cottrell began by describing his company, Anonymizer, Inc., and their history, products, and services. He described some of the basic problems in intelligence analysis, pointing out that simple log file analysis is still the most common method. He also noted that tech companies are far from the only ones doing this.

However, whenever you have exposed IP addresses, Cottrell claims, you are leaking information about your business out to the world. Even if you engage in IP blocking (which people can see you do) or IP spoofing (having different versions of Web sites for different visitors), you are still "hemorrhaging" data out. For example, competitors can read your whitepapers, product listings, press releases, and so forth to discover your business and research profile.

Cottrell then cited a number of examples: that prior art is a huge intellectual property issue, and if you have visited a competitor's Web site, you may be exposed; Cisco employees who surfed to a competitor's Web site were presented with a job offer; European hackers who would launch automatic DDoS attacks against visitors to their Web site who were seen to be running Microsoft IE and coming from a Washington, D.C., IP address.

One solution to conducting this kind of intelligence analysis is to anonymize traffic by routing it through another network and rewriting the headers. However, Cottrell pointed out, it is tricky to do this without introducing inconsistencies (e.g., traffic made to look as though it originated in Hong Kong, but the time zone was PST). Further examples of intelligence analysis were given: airlines scraping all their competitors' fares; retailers profiling users both on their buying habits and on their geographical location.

Next, Cottrell moved on to examples of counter-intelligence. Less aggressive companies can monitor their traffic closely, for example, for a 3 sigma change in interest in whitepapers. Companies in a bidding war might bug the investor section of their Web site.

A questioner asked if companies block Anonymizer. The answer was, yes, they try, but they cannot do so effectively, due to Anonymizer's large, scattered, frequently changing IP address space. Another question was about ethical boundaries of Anonymizer. Cottrell said that they try to detect and reject attacks, spam, IP floods, and the like. Their policy, he said, was that they would block activities that are illegal in the U.S.—however, all other uses by enterprises were permitted after a committee review. He also stressed the importance of Anonymizer ensuring privacy by never, ever keeping logs. Other questions dealt with: issues of ISP trust (Anonymizer must engage in long discussions when

buying IP blocks from a new ISP); working with law enforcement (Cottrell said they are usually respectful and that Anonymizer cooperates where appropriate); how many companies engage in dynamic customer profiling, for example, offering different prices to different people (he said it was "very widely used" and most big sites did it).

■ ***Preventing Child Neglect in DNSSECbis Using Lookaside Validation (DLV)***

Paul Vixie, Internet Systems Consortium, Inc.

Summarized by Chaos Golubitsky

In this talk, Paul Vixie proposed DNSSEC Lookaside Validation (DLV) as a means of overcoming the road blocks which currently prevent deployment of Secure DNS. He justified the need for such a solution with some history. First deployed in 1987, DNS was not designed to enable authentication of name data. The IETF has been working on Secure DNS since 1994, but it has still not been deployed at any sites.

The current Secure DNS proposal, DNSSECbis, works by introducing a new set of DNS RR types, most of which are used by a zone to enable authentication of its own DNS data using public key cryptography.

DNS is hierarchical by design: just as DNS validators hard-code the locations of the root nameservers, DNSSECbis validators will hard-code the root nameserver DNSKEY. The effect is that no zone can deploy DNSSECbis until the zone's parent has deployed it. In particular, DNSSECbis cannot be meaningfully deployed until it is present in the root and .com zones. Since parties higher on the DNS tree see more of the costs of DNSSECbis and fewer of the benefits, this may never happen.

DLV is designed to allow zones to deploy Secure DNS even if their parents have not deployed it. It introduces a DLV resource record, which is functionally similar to the DS (Delegation Signer) record. It also introduces DLV namespaces, zones which have offered to serve DLV data for all or part of the DNS space. A validator looking for Secure DNS data for a given zone must first look for a DS record at the zone's parent. If none is found, the validator may then look for entries within any DLV namespaces it knows. For example, if dlv.isc.org is a DLV namespace and there is no DS entry for vix.com, then a DLV entry can be stored at vix.com.dlv.isc.org. Therefore, vix.com can deploy DNSSECbis even if none of its parents have done so.

DLV is intended as a temporary solution, which should be shut down either when deployment of DNSSECbis reaches critical levels or when it becomes clear that DNSSECbis will fail. As a result, the DLV namespace should be introduced by a public benefit corporation which uses a cost-based fee structure. Vixie identified his employer, ISC, as committed to this model. BIND 9.4.0, to be released soon, will contain support for DLV, and ISC will operate a DLV registry using BIND9. For further information, search for "ieice vixie dlv" to find Vixie's 2004 paper introducing DLV.

Attendees asked how individuals can convince their employers to roll out DLV, and how ISC plans to authenticate DLV registrants. First, the announcement of BIND 9.4.0 will announce DLV, since many sites will deploy as soon as possible. Second, Vixie is compiling a set of marketing whitepapers to advertise DLV. Authentication of registrants involves liability risk for ISC; the exact mechanism has not been determined. Possibilities include: initially registering DLV records only for people with whom ISC has an existing business relationship; charging a fee to cover the cost of verifying registrants' identities; obtaining identity information from existing registrars; or using a web-of-trust scheme, starting with existing ISC business partners.

PLENARY SESSION

■ ***Picking Locks with Cryptography***

Matt Blaze, University of Pennsylvania

Summarized by Alex Boster

Matt Blaze did not, in fact, give a talk on lock picking using cryptanalysis. Instead, he talked about his more recent research into wiretap eavesdropping and applying computer and network security techniques to wiretap systems. Blaze pointed out that there are important legal implications to vulnerabilities in wiretap systems that might cast doubt on the reliability of the tap.

Blaze then described the two basic types of wiretaps: pen registers, which record the numbers dialed but not the audio, and full audio taps, which have greater legal restrictions. A description of basic telephone and wiretap terminology and functions followed. Blaze's research focused not on the many ways one could do wiretaps but, rather, on how law enforcement agencies actually do them.

Various types of wiretap equipment were then presented. Blaze pointed out that wiretaps do not perform exactly the same as the phone company's central office (CO) equipment—and that opens up some vulnerabilities. He was able to reverse-engineer the signals used by wiretap systems. Taking advantage of differences in tolerance (the phone tap equipment is more sensitive to the on-hook signal than the actual CO equipment), he was able to play two recordings of the same phone conversation: a short one where the wiretap had been fooled into halt-

ing recording, and the full version recorded directly from the line.

Questioners asked if audio and call detail logs are correlated. Blaze replied that they were not standard operating procedure. Blaze was also asked about parallels between the talk he gave about wiretaps and his research on lock picking and cryptography. He said that parallels included understanding the limits to mechanical devices, noting that we tend to upgrade them to electronic devices, and that reducing the problem to software might not be a good idea.

INVITED TALKS

- ***How Sysadmins Can Protect Free Speech and Privacy on the Electronic Frontier***

  Kevin Bankston, Electronic Frontier Foundation Staff Attorney

  Summarized by Rik Farrow

  Bankston began with a history of U.S. laws relating to wiretapping. Until a Supreme Court decision in 1967, U.S. citizens could expect almost no privacy from surveillance via taps installed on telephone lines. The Wiretap Act of 1968 placed federal law in line with the court decision, but the law and later court decisions still permitted pen-traps, collection of call log information. In 1986, the Electronic Communications Privacy Act attempted to modernize the law. In 1996, CALEA forced telephone providers to include mechanisms for install taps and/or pen-traps via phone switches, in support of law enforcement armed with judicial permissions.

  The Patriot Act changed much of the landscape, making it possible for a tap to be installed and the target never informed of it, unlike earlier laws. NSLs (National Security Letters) issued directly by the FBI can also not be challenged or made public, ever, and an article in the *Washington Post* suggests that these letters are being used for surveillance of domestic opposition to the current administration.

  What can sysadmins do to protect the privacy of their users? Bankston had a series of suggestions:

  - Minimize logfiles; storing logs forever is more likely to cause problems than to help you.
  - Have a clear policy about how long you keep log files, and follow it.
  - Negotiate to keep the government software and hardware out; you don't have to redesign your networks—yet.
  - Lobby for legal challenges (you can call a lawyer).
  - Give notice whenever possible.

  If you are asked to do surveillance, do check on the law. Contact EFF, even if you get a supersecret order, or you can go to a lawyer (ask your boss). You often do have the power to inform people if their info has been subpoenaed. Yahoo has done this.

  You can also join the EFF (eff.org).

- ***Wireless Security***

  Michael H. Warfield, Internet Security Systems, Inc.

  Summarized by Chaos Golubitsky

  Michael Warfield provided an overview of the current state of wireless security. The focus of the talk was classification of methods of attacking networks, outcomes of successful attacks, and available means of protection.

  While war driving for insecure access points is the best-known exploit of wireless networks, others are also in use. Attackers can run their own APs, either to opportunistically snoop on any machine with an open wireless configuration (inverse war driving) or with a specifically chosen SSID to mirror a legitimate network (evil twin attack). In a hotspot battle, an attacker launches a denial of service attack on a specific wireless network by interfering with the channel used by that network.

  Once a network has been exploited, the attacker's target may be the network itself (simple bandwidth theft, denial of service), the contents of machines using the network (information theft, extortion), or the use of the network to anonymize illegal activity (spam, visiting illegal Web sites). Warfield noted that arp cache poisoning can be used to redirect interesting traffic from adjacent wired networks, and that owners of wireless networks may face liability or reputation problems due to illegal activity on their networks.

  The last portion of the talk focused on the benefits and shortcomings of common wireless network defenses. Warfield stated that MAC address control is not very valuable—the administrative overhead of maintaining tables is high, and guessing a valid address can be trivial. Since tools such as Kismet can easily probe silent access points, turning off SSID broadcasting is not a good security measure either. In general, WPA should be preferred to WEP. However, both protocols have a history of weak implementations, and a modern WEP network may require more traffic in order to break a key than a broken WPA network. Virtual Private Networks should be used, but they provide no protection against poorly configured legitimate machines. To the extent possible, wireless networks should be protected against physical threats—for instance, by placing APs in the interior of a building rather than near the outside.

  Warfield repeatedly made the point that it is useful to classify attacks according to whether they are opportunistic or targeted. Evil twin attacks and hotspot battles necessarily explicitly target the

network being attacked, while others may be indiscriminate attacks against any nearby network, or may even be accidents. Similarly, weak countermeasures may have value because they prove intent. WEP is easy to crack, but it cannot be cracked accidentally, so an intruder on a WEP-protected network can be assumed to be launching a deliberate attack on that network.

The full slides for the presentation are available at http://www.wittsend.com/mhw/2005/Wireless-Security-LISA.

ACCESS CONTROL

Summarized by Chaos Golubitsky

■ *Towards a Deep-Packet-Filter Toolkit for Securing Legacy Resources*

James Deverick and Phil Kearns, The College of William and Mary

The goal of this project is to provide a toolkit for authenticating access to non-secured legacy resources through a firewall. The toolkit should consist of a central library of solutions which can secure many network services with minimal per-service coding, and should not require that the protected software be altered in any way. Jim Deverick presented a proof-of-concept implementation which used the Linux netfilter packet filter to authenticate NFS mount and umount requests and LPR printing.

Both services are wrapped using a netfilter rule set which captures packets representing new requests and holds these packets while they are examined by user-space code on the firewall. The firewall code performs an external authentication step, generally by contacting a daemon on the client system with a challenge/response request. If authentication is successful, the connection request is forwarded to the server. If not, the toolkit cleans up any loose TCP connections created on the server.

As implemented, the toolkit secures only NFS mount and umount requests and initial LPR connections. No authentication is required in order to submit packets to a connection already in progress, and, in the NFS case, no authentication is required in order to perform NFS operations on a mounted file system. Since netfilter operates on TCP packets, authorization could be provided at the granularity of source and destination IP/port pairs, although the current implementation authorizes the entire source host to send packets to the target port. In the future, the authors hope to improve the implementation so that wrappers can be added and modified more easily.

■ *Administering Access Control in Dynamic Coalitions*

Rakesh Bobba and Himanshu Khurana, NCSA and University of Illinois at Urbana-Champaign; Serban Gavrila, VDG Inc.; Virgil Gligor and Radostina Koleva, University of Maryland

Radostina Koleva introduced a prototype of a set of tools for administering dynamic coalitions. A dynamic coalition is a set of independent organizations (domains) that share resources for use in a joint project. The example given was that of a pharmaceutical company, an FDA review board, and a research hospital working together on a new drug. For a coalition to form, each domain must have an incentive to bring private resources to the table. A flexible framework is needed to control other domains' access to these resources. The coalition may create shared resources, which will be owned and administered by consensus among domains. In addition, new domains may join an existing dynamic coalition for certain projects, and previous member domains may leave.

Negotiating a coherent access policy is a challenge, as is implementing a formal policy specification. The tool set presented here can help negotiate coalition policies in a semi-automated fashion, allow consensus-based administration of joint resources, distribute and revoke privileges efficiently, and provide each member organization with tools to assess current and proposed policies.

The tool set is implemented over a Windows 2000 server and consists of the Common Access State, a formal specification of the access policy implemented using an RBAC tool and Active Directory; policy management tools for domain administrators; three types of certificate authorities, for authenticating users within each domain, for authorizing access to resources belonging to each domain, and for authorizing access to joint resources using a shared-RSA cryptosystem; and a secure communication framework allowing trusted communication between domains.

An attendee asked how the coalition verifies that the domains are not passing shared information to outside parties. Koleva replied that confidentiality would need to be enforced using a non-technological mechanism such as a legal agreement.

■ *Manage People, Not Userids*

Jon Finke, Rensselaer Polytechnic Institute

Jon Finke contends that it is possible to maintain a single source of data about the people at your institution, and that the system administration group is well placed to run such a system. In this talk, he discussed details and strategies for such a database, using the implementation he oversaw at RPI as an example.

The driving principle is that every person in the system should have a status ("student," "faculty," "staff," "guest") and that a reasonable provider should maintain data related to each status. For instance, Human Resources should maintain staff data, while

the registrar handles students. This system appeals to prospective data providers because they can be given total authority over their data. Consumers can use the database to group people accurately based on status. For instance, the library can set different book check-out intervals for professors and for students.

Finke then discussed technical details of the implementation, including the types of information stored in the database for each class of users. He discussed the maintenance of guests, which is complicated because universities have a large number of types of guests (e.g., visiting professors, dependents of other people in the system). In order to manage guests more easily, he requires that someone be responsible for each guest's data (the hosting department for visitors, the employee for dependents), and that guests expire from the database unless their data is explicitly renewed.

One attendee asked about problems encountered when correlating multiple sources of data. Finke replied that his group attempts to ensure that each person has only one database entry, but it does not always succeed. Once data providers are using the system, getting them to maintain their data is not hard, since users now know where to complain if their information is inaccurate.

INVITED TALKS

- ***Wikis, Weblogs, and RSS for System Administrators***

  Dr. Jonas Luster, Socialtext, Inc.

  Summarized by Laura Carriere

  Luster began his highly entertaining talk by acknowledging that wiki and blog technologies have been around for a number of years now and are well established. Sociologists believe that the strongest human drives are to communicate and to make sense of communication; Luster stated that everyone is a sender but pointed out that there is no way to filter or roll back the data once it has been sent. It is the receiver's job to filter the data stream. Weblogs are an example of sending without filtering. RSS is an example of the receiver filtering the data. To emphasize his point, Luster observed that as the speaker he could choose to moon the audience and we would be unable to stop him, only to try to filter the image.

  Wikis, as opposed to Weblogs, give permission to the receiver to participate. This makes them collaborative and creates fertile ground for communication.

  Luster went on to describe the Pastures Theory, which explains that areas with the greenest grass attract the most cows. These cows then fertilize these areas, and this promotes the growth of more green grass, which attracts more cows. He compared this process to a busy wiki such as Wikipedia. Luster proposed that adding syndication to Weblogs, although it adds value by providing filtering, decreases the opportunities for fertilization and leads to empty pastures and deserted Weblogs. Luster then cautioned the audience to resist the temptation to compare our users to cows processing grass, but many of us were stuck with this image.

  Luster presented survey results which found that there are 486 Weblog projects and 198 wiki projects currently available, and he suggested that we'd be better off with more wiki software and less Weblog software. He also reported that there were 16 million Weblogs in November 2005 and 13,000 contributors to Wikipedia. The average user is comfortable with this technology and users reported that their coding and HTML skills improved with Weblog development, although he expressed some skepticism about this result, based on his observations of many Weblogs.

  Luster offered his view that the future holds tighter integration of video, audio, text, and collaboration and that these technologies may converge. He acknowledged that the required increase in complexity will increase the burden on the software maintainers. He also expressed concern that legal issues related to freedom of speech may soon come into play but suggested that the technical people leave this to the lawyers.

  During the Q&A period, Adele Shakal, Caltech, asked for advice on social engineering strategies to deal with outdated content. Luster offered two recommendations: tie it to the user's paycheck by making it standard company practice, and automate a congratulations email after every 1,000 visitors, to encourage voluntary page maintenance.

  At the conclusion of the talk the author expressed his pleasure at being able to share both the cows and the mooning images with us and then performed a live blog update rather than a live moon. The audience was profoundly grateful for his discretion.

- ***Using Your Body for Authentication: A Biometrics Guide for System Administrators***

  Michael R. Crusoe

  Summarized by Josh Simon

  Michael Crusoe, a recent escapee from the biometrics industry, spoke about using biometrics from a sysadmin point of view. It was a high-level overview of the major biometric modalities, or methods of using body parts for identification. Techniques included:

  - Facial recognition, which are error-prone in two dimensions due to changes in position and lighting.
  - Fingerprinting, which can use the actual image, and the minutiae or the changes and breaks in ridges;

real-world testing shows that errors, both false positives and false negatives, decrease as the number of fingers examined increases.

- Hand geometry readers, the largest-deployed technology today.
- Iris recognition, which is the most accurate, due to the large amount of data available in a small space (striations, positioning, etc.), but which is very expensive to calculate; only one vendor is in this space (with soon-to-expire patents, so this may change).
- Speaker recognition, or voice-response.

Other modalities were mentioned, including vein recognition (using the pattern of the veins in the hand) and dynamic signature recognition (specifying the location, pressure, and velocity of the pen). Efforts are made to ensure that the body part is live (either by prompted motion, such as smiling or blinking on cue, or by scanning for temperature or motion).

WORK-IN-PROGRESS REPORTS

Summarized by Charles Perkins

- ***Bedework Open Source Institutional Calendar System***

Jon Finke, Rensselaer Polytechnic Institute

An open source standards-conforming calendar system designed to meet institutional needs, Bedework presents a Web interface, supports subscriptions, and presents a calDAV interface. iCal and skins are supported. Oracle is not used. Bedework is written in Java. For more information, see www.bedework.org.

- ***DeSPAC-SE: Delegated Administration Framework for SELinux***

Ryan Spring, Herbey Zepeda, Eric Freudenthal, and Luc Longpre, UTEP; Nick West, Stanford University

Eric Freudenthal presented a delegated administration framework for SELinux. DeSPAC-SE uses Mandatory Access Control to create security domains, and an active classifier with human intervention creates security tables of program types and allowed behavior. Security classification can be delegated and is amortized over many systems.

- ***Deployment of BladeLogic for Access Control Restriction, Change Tracking, and Packaged Software Distribution Primary to Ensuring Sarbanes-Oxley Compliance***

Michael Mraz

Developed for Solaris on SPARC as well as RedHat and SUSE x86 Linux, the software enables logging and auditing from development, through QA, and into production of complete software systems.

- ***VNC Manger: A Software Thin Client Using Perl, VNC, and SSH***

Wout Mertens

Mertens showed a brief live demo of Perl + TK software for managing multiple sessions of VNC over SSH with load sharing. The software thin client works on any UNIX, and special attention has been paid to server-side Solaris. Wout's presentation tied for best WiP. For more information, see http://sf.net/projects/vncmgr.

- ***An Exoskeleton for Nagios: Scalable Data Collection Architecture***

Carson Gaspar

Gaspar shows how to solve limitations of Nagios by adding a queueing server, a modular client agent, a config-file generator, an rrd-based trending server, and a ping agent. Multiple Nagios servers in passive pipe mode display and act on queued data.

- ***A Brief Look at RSA Moduli***

James Smith, Texas A&M

In his presentation, subtitled "What an English Major Learned in Class," James took the audience on a quick spin through the set of mathematical knowns and unknowns when narrowing the search space for finding factors of an RSA key.

- ***Mail Backup***

Dan McQueen, Cisco

Designed by Dan and coded by Ed Miller, this Sendmail/procmail backup system makes local copies of incoming mail automatically and allows users to initiate restoration of messages that might be lost due to user action before the nightly filesystem backup occurs. Text- and GUI-based restore tools are provided. Retention periods can be set. Restoration is a resend. Docs are forthcoming, and there are plans for open source. For more information, email dmcqueen@cisco.com.

- ***What I Did on My LISA Vacation***

Dave Nolan, CMU Network Services

Dave described the network architecture set up for the LISA conference. He addressed problems with network performance, reliability, and transparency, suggesting that for success one should "clone Tony" and spend money. Good results were had for LISA '05 because of a hotel-link upgrade, donated hardware, and excellent volunteer staff. Monitoring was done with the cricket collector, drraw drawing engine, argus network flow analysis tool, and mon nagios.

- ***Pretty Network Pictures***

Dan Kaminsky, DoxPara Research

In his presentation, subtitled "I Like Big Graphs and I Cannot Lie," Dan explained that while visual displays allow a human to absorb more complexity than text, animation encodes even more complexity. He then demonstrated real-time tcpdump data piped through OpenGL and displayed as video. With this codebase, Dan asserts that "OpenGL does the graphing, Boost does the layout, the programmer gets to be lazy." Tied for best WiP. For more information, email dan@doxpara.com.

■ ***How to Ask Questions the Right Way***

Cat Okita

Cat promoted asking better questions of those seeking technical help, including: What do you want to do? What have you tried to do? What happened? A little more detail please . . Got any ideas?

■ ***Portable Cluster Computers and Infiniband Clusters***

Mitch Williams, Sandia National Labs

Mitch described his work with clustered computers from the extremely small (one foot tall and 6x6 inches wide) to the Thunderbird system, which is #5 in the supercomputer list. For more information, see eri.ca.sandia.gov /clustermatic.org.