law) for the approximately 60% non-lawyers attending the conference .

In discussion, someone commented on Mr. Swire's model, pointing out that he was distinguishing between the physical world and the software world but that a distinction made between mechanism and instances would have been a better approach. Mr. Swire replied that when considering instances, one must often consider the first instance differently than others (since that will often educate the defenders and change the effect of subsequent instances). This led to a discussion of the ability of the law to operate in this complex arena (and the likelihood, or not, of lawyers staying out of the fray). There seemed to be some agreement that we will have some very confused judges, at least for a while.

PANEL

Brief concluding remarks

Jennifer Granick, Stanford CIS; Lauren Gelman, Stanford CIS; Scott Blake, BindView; Greg Schaffer, PricewaterhouseCoopers

No one today has argued against the idea that the market has failed to provide security. Instead of capitalism saving us, we are beginning to conclude that there may be a role for government, a conclusion that many of us find both interesting and disturbing.

There are some interesting (legal) questions to be answered with regard to disclosure, nondisclosure, and liability. What if one can become liable for knowing something and not disclosing it?

Security is about more than fixing "this one bug." It could be about democracy. We don't know enough about security to know that it ought to (or not) be considered differently from other scientific enterprises.

Some people think that the disconnect is about Republicans and Democrats, but it is really about the information-technology and legal communities. Both have well-developed models of their universes and like to be the masters of their respective domains. Neither likes the discomfort of not having a handle on important things that apply to their realms. There are lots of people who have not thought about these problems and won't until there is a crisis, and then the decisions are unlikely to be well-considered and thoughtful. There is a serious need for us to think about these problems in advance, as we have been doing today.

17th Large Installation Systems Administration Conference (LISA '03)

San Diego, California October 26–31, 2003

Keynote Address

Inside eBay.com: The System Administrator's Perspective

Paul Kilmartin, eBay, Inc.

Summarized by Bryan Parno

Kicking off the 17th annual LISA conference, Paul Kilmartin, eBay's director of availability and performance engineering, gave a spirited and engaging tour of the development of eBay's infrastructure, from a single PC in eBay founder Pierre Omidyar's bedroom to the current SAN-based system composed of hundreds of enterprise-level machines. Along the way, eBay's user population exploded from a few hundred in 1995 to over 85 million today.

Throughout the talk, Kilmartin stressed the incredible importance of availability. Since eBay averages $738 of gross merchandise sales every second, the prospect of any prolonged outage is costly indeed. This intense usage also makes eBay the world's 75th largest economic market, falling somewhere between Uzbekistan and the Dominican Republic. Kilmartin repeatedly emphasized how the magnitude of eBay's 85 million user-base impacts virtually every decision the company makes.

In the historical segment of his talk, Kilmartin highlighted eBay's transition from a system based on two-node Veritas clusters to a large-scale SAN. On the plus side, this cut down on the amount of idle hardware, always an important consideration for cost-conscious administrators. It also provided a greater degree of fault minimization and isolation, since the two-node clusters suffered from electrical issues during servicing. Unfortunately, shortly after the migration to the SAN, the co-location company hosting the site announced it would be going out of business. Kilmartin's team of system administrators built an entirely new SAN in three weeks and made the migration with only two hours of downtime in September of 2001. The bankruptcy of the Exodus storage facility in November of 2001 forced yet another move.

Even though the public perceives eBay as an industry leader, Kilmartin repeatedly emphasized his preference for remaining firmly in the mainstream of technology. On several occasions, he urged the audience to forge on ahead and aggressively report problems, so that after a few years of maturation, eBay could adopt the "new" technology. He offered several tips to the audience, encouraging system administrators to doubt everything, to make the system work hundreds of times before trusting it, and to challenge "best procedures" by at least asking for references. He also emphasized the importance of knowing one's role on the team, citing his initial resistance to eBay's foray into the car market (now, he says, a Corvette sells on eBay every 64 minutes). Kilmartin also stressed the need to constantly seek out a better understanding of the customer and how the customer uses the product. Commenting on hiring decisions, he reminded the audience that neither experience nor certification necessarily equates to competence. Concluding with a return to the theme of availability, Kil-

martin asserted the need for vendors to recognize eBay as an active customer, not a cadaver; in other words, the company needs working solutions that can be diagnosed and repaired on the fly, not systems that need to be taken offline and dissected to provide information.

Refereed Papers

Administering Essential Services

Summarized by Ari Pollack

Radmind: The Integration of Filesystem Integrity Checking with Filesystem Management

Wesley D. Craig and Patrick M. McNeal, University of Michigan

Wesley and Patrick introduced radmind, a filesystem management tool designed to replace similar tools, such as Tripwire and cfengine, and overcome some limitations with existing products. Tripwire, for instance, does not scale well or know the difference between unintended changes and OS updates.

Radmind is based on existing work from people in the sysadmin community such as Evard and Anderson, and on features from tried-and-true software. Like Tripwire, it includes integrity. Features in both rsync and radmind include copying of files and comparison to policy, not a live filesystem. Borrowing from cfengine, radmind provides abstract configuration and abstraction of any file set.

Radmind goes further than tripwire; in addition to detecting unwanted changes to the filesystem, it can automatically revert back to a known good state configured in the policy. It only generates reports when something unusual happens. It is easy to understand, has simple setup and configuration, and requires no programming skills for successful use. Radmind is platform-independent; it works on Windows, and it is already in use on MacOS X laptops, Linux and Solaris servers, and supercomputing clusters.

Further Torture: More Testing of Backup and Archive Programs

Elizabeth D. Zwicky, Great Circle Associates

Elizabeth presented the results of her findings from torture-testing various backup tools for UNIX and UNIX-like systems. This is a follow-up to her 1991 paper, which was inspired by frustration at conflicting rumors and vague documentation. The term "backup program" is used loosely; there is no correct term for something that's intended to copy files to another medium for storage (rather than immediate usage).

What she found in 1991 can be summed up as, "don't trust what you've heard, go out and verify." She had heard reports that "cpio doesn't handle too many hard links," so she found out what "too many" meant.

Her latest paper presents a new round of verification of old, out-of-date data. Some of the properties of backups she covers are: file size, devices, strange names, access permissions, holes (numerical representations of nulls on the filesystem), long names, and links. In 1991, every tool died at some point except dump, resulting in core dumps and/or data corruption. Now, nothing handles paths over the maximum path length defined by the operating system, and nothing but restore handled holes absolutely correctly.

Elizabeth says that while backups are difficult, testing backup tools is fun and not that hard. Also, backup programs have different targets and are not consistently useful to everyone. She also presented some conclusions stemming from her research:

- Don't write your own backup program; there are more than enough already.
- Never use old file formats for backups.
- The name of your backup program does not predict its performance in your configuration.
- Long pathnames are an unsolved problem.
- Trust, but verify.
- Backup programs need time to mature.

An Analysis of Database-Driven Mail Servers

Nick Elprin and Bryan Parno, Harvard University

Nick and Bryan took a look at the different kinds of common mail storage formats in use. The three most common are: mbox format, where every email is concatenated into a flat text file; maildirs, where every email is stored in a database file; and databases, where all mail is stored in some kind of structured database.

The two database formats used for testing were Cyrus, which uses Berkeley DB, and their own SQL model using MySQL. Here is what Nick and Bryan found:

- Mbox performs better than Cyrus for a small account in a full-text search.
- Cyrus performs better than maildir and mbox for larger accounts.
- MySQL performs better than the others overall.
- Maildir always performs the worst.

Databases allow better fine-tuning of mail servers and better scalability. File-based solutions perform better on some operations, such as expunging mail. However, performance is usually not the only factor when deciding on a mail format. Maildirs do not suffer from the same locking problems as mbox, and a structured database may require more overhead than is acceptable in some situations.

INFORMATION AND CONTENT MANAGEMENT

Summarized by Kenytt Avery

A Secure and Transparent Firewall Web Proxy

Roger Crandell, James Clifford, and Alexander Kent, Los Alamos National Laboratory

James Clifford describes the LANL Web proxy as a "benevolent man in the middle." In contrast to ordinary Web proxies like Squid, the LANL Web proxy provides access control on incoming rather than outgoing connections. The purpose of the proxy is to allow access to internal Web applications (e.g., Web mail, Nagios network monitoring) from public Internet sites outside the firewall.

The proxy consists of two pieces, the redirection daemon redird, which redirects HTTP requests for internal documents to the equivalent request via HTTPS, and the Web flow daemon wfd, which handles authentication and forwarding requests to the internal network. The external server contains a wildcard SSL certificate for lanl.gov, allowing it to proxy for any internal system.

According to the authors, the chief benefit of the proxy solution is its simplicity, requiring no configuration changes or extra software to be installed on the client beyond an ordinary Web browser. This is in contrast to VPN solutions, which require client software and user training, or to non-transparent proxy servers, which require browsers to be configured to use them.

An important question from the audience concerned the security of potential clients. An untrusted client machine might be running keystroke logging or screen capture software. Clifford responded that the solution has worked well as a stopgap measure until a full VPN can be implemented. In the meantime, efforts have been underway to educate users about the risks of using unknown clients.

URL: *http://www.lanl.gov/orgs/ccn/publications.shtml*

Designing, Developing, and Implementing a Document Repository

Joshua S. Simon, Consultant; Liza Weissler, METI

Josh Simon described a solution to a problem faced by many large sysadmin teams, that of finding documentation. In order to address the constant flow of email asking about various tasks within their consulting company, he and Liza Weissler built a Web-based document management system with the goal of making it easier to find information.

The first problem the authors faced was one of categorization: At one point they identified 52 different types of document. While it is clear that a document management system is considerably more useful when items are separated into categories, users are often unwilling to make the effort to do so as each document is entered. A practical solution was to define a small number of top-level categories (e.g., Customers, Internal, Marketing, Recruiting, Other), each with a small number of subcategories. Categories were assigned single-letter codes, allowing each document to be classified with a two-letter code (e.g., IC for Internal Code).

The other major problem the authors faced was maintaining the metadata about each document once it had been stored in the system. While users submitting documents were encouraged to supply metadata, consultants who were not currently assigned to billable projects were recruited to serve as "librarians," with the ability to edit and update other users' records.

Combining a coarsely grained categorization scheme with constant maintenance by librarians dramatically improved the accessibility of information to employees. The system began with approximately 800 documents and grew to 1200 in its first five months. By that point, only two documents remained in the "Other" class. The system is still in use, and the authors hope to make the code publicly available.

DryDock: A Document Firewall

Deepak Giridharagopal, University of Texas at Austin

Giridharagopal works in a university research lab, a relatively open environment where many autonomous groups share responsibility for publishing content to the Web. The lab's management needed to enforce a policy on publishing information to the Web, ensuring that sensitive or proprietary information is not accidentally made available on the public Web server. Enforcing policy requires oversight and accountability, both of which are addressed by the DryDock system. Until the implementation of DryDock, policy was enforced only when complaints were received.

The DryDock system uses a Web application to manage a Web site. Content is stored in CVS, and document metadata and approvals are stored in a MySQL database. The approach requires two Web servers: an internal staging server located behind the firewall and an external production server located on the DMZ. Authors are free to work with the content on the staging server, using methods such as FTP or WebDAV to access the document root. The production server, however, is stripped of non-essential programs and hardened. DryDock automatically propagates content from the staging server to the production server via SSH once the appropriate approvals have been obtained.

Giridharagopal suggests that one way to look at DryDock is as a tool to shift responsibility for content oversight away from sysadmins and back to management. Sysadmins are responsible for keeping the system running, but in order for any content to appear on the public Web site, DryDock requires it to be approved. Web authors are free to work directly on the staging server, and Dry-

Dock will show the differences between the current contents of the staging server and that of the public Web site. Users are informed when pages have changed, and those with management authority are able to approve publication. DryDock logs the time at which files were approved and which users approved them, and allows content to be rolled back to previous versions when necessary. In use for over a year, the system has resulted in improved Web server security and better management oversight of the publication process.

URL: *http://tools.arlut.utexas.edu/DryDock/*

SYSTEM AND NETWORK MONITORING

Summarized by Venkata Phani Kiran Achanta

Runtime Detection of Heap-based Overflows

William Robertson, Christopher Kruegel, Darren Mutz, and Fredrik Valeur, University of California, Santa Barbara

This paper is about a technique that protects the management information of boundary-tag-based heap managers against malicious or accidental modification. William started out by describing the motivation behind his work, which he mainly attributes to the increasingly common buffer overflow exploits resulting from use of various insecure languages for application development. He reinforced his argument by citing the recent vulnerabilities in OpenSSH, MySQL, etc.

He explained how the buffer overflow exploit occurs and then discussed existing approaches to detect and prevent them, pointing out flaws and describing limitations in existing methods.

Then he introduced his approach, an adaptation of the canary-based stack-protection scheme, where the canaries are seeded with a random number, which a mechanism prevents the intruder from seeing. This detection scheme has been implemented as a patch to the GNU libc library.

William did some micro- and macro-benchmarking and stability evaluation. Later, he discussed techniques to be adopted to handle buffer overflow exploits.

The software can be downloaded from *http://www.cs.ucsb.edu/~rsg/heap*.

Designing a Configuration Monitoring and Reporting Environment

Xev Gittler and Ken Beer, Deutsche Bank

The configuration monitoring and reporting environment (CMRE) is a tool designed to collect and report on the many configuration details of systems within an enterprise. Its goal is to provide a single, complete, up-to-date repository of all system configuration information regardless of platform or use.

Gittler described their operating environment as a conglomeration of diverse systems with different standards and procedures and discussed the potential problems posed by such an environment.

CMRE needs few prerequisites in order to do its job; in fact, the necessary framework for CMRE already exists at their shop. CMRE is modular, flexible, and runs on many different platforms. It is written in a combination of Perl, Korn shell, and PHP and uses proprietary as well as open source software. CMRE currently collects data on thousands of UNIX and Windows systems at Deutsche Bank worldwide.

Gittler showed us some GUIs of CMRE and explained the usefulness of the data it collected. He then described the scenarios where they ran into problems when designing and deploying this system.

Although most of the organizations have this kind of monitoring tool already in use, Gittler advocated the superiority of CMRE, citing the simplicity and non-intrusive nature of the tool and the ease in interpretation of the gathered data.

Contact information: *xev.gittler@db.com*; *ken.beer@db.com*

New NFS Tracing Tools and Techniques for System Analysis

Daniel Ellard and Margo Seltzer, Harvard University

Daniel opened with the background and motivation for doing the paper. He then discussed the usefulness of looking at passive NFS traces over a period of time and talked about the work already done in this arena. He went on to cite some examples of basic and advanced analyses of the gathered data and their relevance to system administration.

The two main tools used for data gathering and analysis were nfsdump and nfsscan. Several related utilities were used in the analysis part. The data was gathered in a university environment, and measures were taken to anonymize the data as much as possible. There is control over anonymity of the data if someone wants to use the tool for real data collection and analysis.

The software and the results can be found at *http://www.eecs.harvard.edu/sos/software/*.

DIFFICULT TASKS MADE EASIER

Summarized by Jarrod Millman

EasyVPN: IPsec Remote Access Made Easy

Mark C. Benvenuto and Angelos D. Keromytis, Columbia University

As a student at Columbia University, Mark developed EasyVPN to integrate an unencrypted, untrusted wireless LAN into the Computer Science Department's LAN and to the Internet. His main design goal was to create a simple and easy-to-use VPN based on IPSec. Unfortunately, as anyone who has tried to do this in a heterogeneous environment knows, setup varies with each

IPSec platform; furthermore, managing certificates is too complicated for users and too time-consuming for administrators. To address these issues, Mark created a solution that leverages the wide availability of Web browsers with SSL/TLS support and the familiarity of users with Web-based interfaces. The Web interface allows the user to create and download the configurations and certificates for their computer without further burdening the system administrator or requiring the user to understand the technical minutiae.

EasyVPN is composed of three main components: the client, the gateway, and the VPN server. The client receives the certificate from the gateway, which serves as the certificate authority (CA). The VPN server trusts the client because it trusts the gateway. Thus, EasyVPN is built on trust and the easy manageability of the CA. To demonstrate the feasibility of such an approach, Mark implemented EasyVPN using Linux FreeS/WAN and Windows clients.

The Yearly Review, or How to Evaluate Your Sys Admin

Carrie Gates and Jason Rouse, Dalhousie University

Many nontechnical managers and employers do not fully understand what a system administrator is or what he or she does. Only recently have there been any publications on the hiring and firing of system administrators. Moreover, there is no clear course of study or career path for becoming a system administrator. Consequently, it comes as no surprise that there is no systematic approach for evaluating the performance or effectiveness of a system administrator. Carrie and Jason presented an approach to evaluating system administrators based on three criteria: achievement of goals, achievement of specified service levels, and general competence. Using these three broad criteria, they developed a quantitative system for evaluating system administrators that is measurable and fair.

The first criterion, measuring the achievement of stated goals, requires that the manager and administrator work together and provides the manager with an objective assessment of performance. To better understand how an administrator was achieving specified service levels, Carrie and Jason refined this criterion to four components: availability, usability, security, and customer service. General competence was measured by how often the administrator needed to revisit the same problem. Breaking the evaluation into these three criteria provides the manager with an effective tool to isolate the system administrator's strengths and weaknesses. They concluded by describing five different scenarios illustrating how you might deploy this system, what types of scores you might get, and an interpretation of those scores with suggestions for appropriate action. It was emphasized that this system was meant to initiate a wider and more extensive discussion on this important topic.

Peer Certification: Techniques and Tools for Reducing System Admin Support Burdens While Improving Customer Service

Stacy Purcell, Sally Hambridge, David Armstrong, Tod Oace, Matt Baker, and Jeff Sedayao, Intel Corp.

Before peer certification, trouble tickets at Intel Online Services (IOS) were received by help-desk technicians, who would pass them on to the system and network administrators to handle. This caused constant interruptions for the administrators, frustrated the technicians because they weren't able to solve the problems, and impeded customer service due to the lack of direct contact between the customer and the problem solver. IOS wanted a way to allow the technicians to handle the tickets themselves, but needed to ensure that the technicians were qualified to do so. To this end, they created a peer certification process to add qualified troubleshooting personnel.

The certification process divided troubleshooting personnel requirements in two ways – specialty areas and specialty levels. Certification for a specific area and level requires previous-level certification, an oral test, and monitored completion of tasks. Once implemented, peer certification resulted in an increase in the number of staff able to make changes and a reduction in the number of trouble tickets referred to the system administrators.

EMERGING THEORIES OF SYSTEM ADMINISTRATION

Summarized by Kevin Sullivan

ISconf: Theory, Practice, and Beyond

Luke Kanies, Reductive Consulting, LLC

Luke describes his development experiences with a configuration management tool, ISconf. Although ISconf has gone through significant rewrites since the initial version, it still functions by pairing listings of commands with a list of hosts for those commands to be run on. ISconf's use of make satisfies three components of deterministic ordering: state maintenance, failure on error, and consistent ordering. The concept of atomicity is one which ISconf does not currently possess. In many processes, the lack of support for atomicity requires human intervention when an error is encountered. Also, hidden preconditions of a system create situations that ISconf would have difficulty handling. The discussion of these shortcomings will help the development of ISconf and tools like it. ISconf is still a very useful tool and when combined with other configuration management tools these inherent problems can be mitigated.

Seeking Closure in an Open World: A Behavioral Agent Approach to Configuration Management

Alva Couch, John Hart, Elizabeth G. Idhaw, and Dominic Kallas, Tufts University

Alva opened by describing a race between theory and practice in which theory always wins. The main goals of his work are portable validation, where validation occurs once and the results are the same everywhere, and to produce an algebraic model of configuration management. Couch contends that these goals can be achieved through the use of closures and conduits. Closures are like a black-box system that has well-defined inputs and outputs and functions exactly as specified. Conduits are communication channels between closures. The first step in developing a closure is separating internal and external parameters. If it were not for latent preconditions, the composition of closures would be closures themselves. This essentially creates complex services with known functionality and well-defined inputs and outputs. File editing was an initial prototype of this work. A file-editing closure can define all permissible actions to a file in an attempt to reduce errors. Many system administrators are wrapped up in the minutiae of the many systems they manage and have less time to do high-level coordination of services. When these low-level systems are treated as closures and conduits, it becomes easier to focus on more advanced system administration tasks.

Archipelago: A Network Security Analysis Tool

Tuva Stang, Fahimeh Pourbayat, Mark Burgess, Geoffrey Canright, Kenth Engø, and Åsmund Weltzien, Oslo University College

Tuva Stang presented a tool that was intended to visually model interconnected networks. These networks can be physical, social, or knowledge networks. Graph theory was used to show the connections that exist between groups of people, hosts, or other information sources. The most well-connected nodes will become visually apparent. An interesting comparison was drawn between an organizational chart and the charts presented here; in some cases they differ, and the truly connected people are revealed. As a security tool, Archipelago can reveal vulnerable points in a network or even the nodes that should be best secured, due to their importance. The graphs produced by this tool show both the importance and centrality of the nodes.

PRACTICUM: UNUSUAL TECHNIQUES FROM THE FRONT LINES

Summarized by William Reading

Three Practical Ways to Improve Your Network

Kevin Miller, Carnegie Mellon University

First Idea: IP Anycast

IP anycast is the same as shared unicast, in which one IP address is assigned to multiple hosts and the network routing is configured to deliver to one of the many machines that have that IP address configured.

Migrating is not very difficult. For servers that simply use DNS, only an update to DNS is required. In an IP anycast environment, without requiring a configuration change, clients end up using a server that is closer to them than others on the network.

Second Idea: Source Address Verification

Filtering is accomplished by performing source address verification on edge routers using unicast reverse path forwarding. This uses the unicast routing table to make the filtering policy and requires little work compared to traditional filtering with ACLs.

Third Idea: Host Filtering

This builds on the topics mentioned earlier. Essentially, the problem is that there are a large number of hosts that need to be denied access to the network due to viruses and such.

Expect scripts are tedious and can cause problems, so a host route is given, essentially pointing to a sinkhole – which then drops the packets. When the host has been cleaned up, the route is removed.

Tossing Packets Over the Wall Using Transmit-Only Ethernet Cables

Jon Meek and Frank Colosimo, Wyeth

Protecting an internal network while monitoring from remote sites considered to be insecure poses a difficult problem. The talk was loosely organized into the topics of hardware, software, and applications.

On the hardware side, simply snipping the wires does not work, and it is haphazard to do things like soldering a paper clip to an Ethernet card if security is concerned.

However, it is possible to create a circuit that does not permit packets to return over the line. By writing custom software which only relays packets to a specified host on an internal network from the crippled line, security can be maintained.

The Realities of Deploying Desktop Linux

Bevis King, Roger Webb, and Graeme Wilford, University of Surrey

Linux offers a number of benefits for deploying on the desktop, yet a certain degree of Windows compatibility is a must. However, using Linux on the corporate desktop reduces the support time required.

Running Microsoft Windows in a virtual machine has a number of benefits for support because the Windows machines do not have direct access to the network, have abstracted hardware, and are not writable by the end user.

The desktops themselves have greater access to scientific applications that only run on UNIX, and there is a completely

supported X server running to host these applications remotely.

CONFIGURATION MANAGEMENT: TOOLS AND TECHNIQUES

Summarized by Marko Bukovac

STRIDER: A Black-Box, State-based Approach to Change and Configuration Management and Support

Yi-Min Wang, Chad Verbowski, John Dunagan, Yu Chen, Helen J. Wang, Chun Yuan, and Zheng Zhang, Microsoft Research

Yi-Min Wang and Chad Verbowski receiving the Best Paper Award from Æleen Frisch

In a dynamic talked welcomed by administrators who have Microsoft Windows machines on their network, Dr. Wang presented STRIDER, a Windows tool that helps to pinpoint the origin of Windows registry problems. Windows XP has about 200,000 registry entries storing all configuration data, so finding a source of evil is downright impossible without a proper tool. By using white-box data (from support documentation) and black-box testing, STRIDER manages to narrow down the number of possible problems in the registry, making identification fathomable for a human administrator.

Starting with all the registry entries, STRIDER creates a smaller subset by mechanically eliminating entries that are irrelevant to the current problem. It then uses a statistical model to filter out the entries that may be relevant but are most likely not the root of the problem.

Each entry in the smaller subset is then compared to a computer genomics database, a data set obtained from troubleshooting experiences and black-box tests, to potentially pinpoint the solution.

In addition to the published paper, Dr. Wang has a Web page at *http://research.microsoft.com/~ymwang* where one can find more information on STRIDER.

CDSS: Secure Distribution of Software Installation Media Images in a Heterogeneous Environment

Ted Cabeen, Impulse Internet Services; Job Bogan, Consultant

CDSS provides a framework for a distribution of software images over a number of protocols. Software images are stored on an isolated server for every user who is trying to download an image. The user can communicate only with the designated server and can obtain only the requested files. The system does not require any additional setup on the user's side, as CDSS uses standard protocols (HTTP, FTP, SMB, etc.) and a set of shell scripts to access the desired information.

A user who visits a Web page that lists all available software images selects the ones he or she's interested in and provides necessary passwords to access them. At that point, a directory is created for that user, containing only the requested images. At the same time, the servers necessary to allow the user to access the data over the desired protocol are configured and started. By using Linux firewall rules, the user's request is redirected to a non-standard port for each protocol and the data is made available.

CDSS is under a GPL license; more information about it can be found at *http://cdss.sf.net*.

Virtual Appliances for Deploying and Maintaining Software

Constantine Sapuntzakis, David Brumley, Ramesh Chandra, Nickolai Zeldovich, Jim Chow, Monica S. Lam, and Mendel Rosenblum, Stanford University

Computer Appliance is a device, like Tivo, for which the software is installed by the manufacturer (who also provides updates) rather than by the user. Sapuntzakis and fellow researchers took this concept and applied it to virtual appliances, which are just like the physical appliances but without the hardware. Rather than running the appliances on the bare x86 hardware, the authors use the VMware GSX Server.

In the presentation and the demo that followed, Sapuntzakis introduced the basic concepts and presented a prototype model that allows creation, publication, execution, and update of virtual appliances. He argues that using virtual appliances reduces the amount of time needed to administer computers, by having a central management unit control all the software for all the appliance users.

Sapuntzakis et al. also developed a unique configuration language, CVL (collective virtual appliance language), whose syntax is used to describe VAP configurations. Their demo showed the audience sample .cvl files and how to administer the VAPs. More information on Sapuntzakis and the project can be found at *http://suif.stanford.edu/~csapuntz/*.

CONFIGURATION MANAGEMENT: ANALYSIS AND THEORY

Summarized by Aaron Teche

Generating Configuration Files: The Director's Cut

Jon Finke, Rensselaer Polytechnic Institute

At LISA 2000, Jon Finke presented a paper about configuration generation from a relational database. At LISA '03,

he shared his improvements using XML and XSL, with data stored in the relational database for configuration management. While the original system worked very well, it wasn't flexible enough. Any layout changes required a PL/SQL programmer, and the PL/SQL programmer needed presentation skills. In comes XML with XSL transforms. The relational database is still used, but the data goes from the database to XML through an XSL translation to the final output. XML and XSL are platform-independent, which makes this solution vendor-independent. And, finally, the move to an XML/XSL system provides basic consistency checking along the transformation path.

Preventing Wheel Reinvention: The psgconf System Configuration Framework

Mark D. Roth, University of Illinois at Urbana-Champaign

Most configuration management tools are designed monolithically and can't mix and match ideas and functionality. This results in lots of wheel reinvention. Mark Roth presented his solution to this problem, psgconf. While monolithic configuration management tools manage file configs, not abstract ones, psgconf solves this problem with modularity. The psgconf framework is a hierarchy of small, write-once-use-often Perl modules that manage the configuration at a conceptual level. It is intended to know what the data is and to control manipulation of that data according the requirements set by the admin.

SmartFrog Meets LCFG: Autonomous Reconfiguration with Central Policy Control

Paul Anderson, University of Edinburgh; Patrick Goldsack, HP Research Laboratories;Jim Paterson, University of Edinburgh.

LCFG is a config tool that takes a high-level specification and generates a machine profile. LCFG can rebuild an entire site from bare metal, given a central source repository. SmartFrog provides a framework for configuration management of distributed applications. It is a runtime environment which orchestrates the workflow of computers according to configuration. SmartFrog in combination with LCFG can control and maintain a robust service that automatically reallocates machines and services based on demand, including the ability to rebuild around failure.

NETWORK ADMINISTRATION

Summarized by Hernan Laffitte

Distributed Tarpitting: Impeding Spam Across Multiple Servers

Tim Hunter, Paul Terry, and Alan Judge, eircom.net

Tim Hunter and Paul Terry receiving the Best Paper Award from Æleen Frisch

The authors' company, eircom.net, is the biggest ISP in Ireland, with approximately 500,000 users. For them, spam is a big problem: On several occasions they have seen their server outages reported on by the media. To help alleviate this problem, they have configured a tarpitting mechanism.

The method known as "tarpitting" involves inserting a time delay between the moment a message is received by the SMTP server and the moment when the server returns its "250 OK" response. This time delay varies: The goal is for it to be zero for legitimate users and up to 30 seconds per message for spammers. This solution is a reasonable middle ground; there is no need to filter messages based on content, which raises privacy concerns or risks dropping potentially valid messages.

The paper explains how eircom.net implemented a centralized database of messages recently received from each client. A "Theory" section of the paper explains how to set the right parameters so client addresses get tarpitted and untarpitted over time, according to how many messages they send. The "Data" section explains how the method was implemented across eircom.net's various mail servers, using qmail as SMTP server, and IP multicast to share client behavior data, which each machine stores locally on a SQL database.

Finally, a "Tarpitting in Practice" section describes the political problems involved in setting the right parameters for the tarpit and developing policies to follow when a would-be spammer is found in the tarpit. The authors also include data gathered from an actual spamming session, with the spammer trying to navigate around the restrictions posed by the tarpit.

This method has helped eircom.net solve the problem of burst attacks, but some work remains to be done regarding lower-level spamming. In conclusion, tarpitting is a useful addition to the anti-spam toolbox.

Using Service Grammar to Diagnose BGP Configuration Errors

Xiaohu Qie, Princeton University; Sanjai Narain, Telcordia Technologies

It is not uncommon for all routers on a BGP network to be operational and yet route packets incorrectly. This happens because traditional network diagnostic tools can only detect localized errors, such as bad cables or software failures. More automated tools are needed to systematically search through the problem space.

This paper analyzes the use of the Service Grammar technique for diagnosing BGP configuration errors. BGP presents a number of challenges for its imple-

mentation: At the low level, individual routers have to be configured independently, yet the high-level global routing policy of the (sometimes very large) network has to be kept consistent across all routers.

Since BGP is a complex protocol, the manual configuration of routers is a time-consuming and error-prone task. This paper presents a Service Grammar for configuring BGP networks. This Service Grammar consists of a "BGP Requirements Language," which expresses the BGP logical structures; a Configuration Database, which abstracts the different vendor-specific configurations; and a Diagnosis Engine, which is a set of algorithms that validates the configuration database and provides useful information for the debugging process.

The paper includes an example network, where Service Grammar was used to diagnose the configuration of nine Cisco routes, grouped in five ASes.

Splat: A Network Switch/Port Configuration Management Tool

Cary Abrahamson, Michael Blodgett, Adam Kunen, Nathan Mueller, and David Parter, University of Wisconsin, Madison

The old network infrastructure of the University of Wisconsin Computer Science Department consisted of multiple unmanaged Ethernet switches, where people would just plug in their workstations. When the old network was replaced with 50 managed switches using VLANs, the need arose to implement a solution to automate the management of the network infrastructure.

After considering the existing solutions, the authors of the paper decided to implement the Splat tool. This tool provides an easy-to-use interface for configuring the switch ports while enforcing sysadmin best practices.

Using Splat's CLI interface is relatively straightforward; the tool was designed to accommodate relatively inexperienced administrators. For example, to connect a host to a switch port, the only required parameters are the hostname and the label of the data jack on the wall. The tool does the rest: updates the database, computes the new VLAN configuration, and issues the required switch configuration command using the Rancid switch configuration manager. The current configuration data is stored in a PostgreSQL database, which can also be queried using Splat.

The use of the tool is enforced because, without it, the VLAN number is not correctly configured for the switch port, which means the network connection won't work. Also, the tool "locks" the switch port to the MAC address of the workstation. Thus, using Splat is easier than changing all these parameters by hand.

This creates a virtuous cycle: the Splat database is the definitive data source for host/switch-port mapping. And since it's easier to use Splat than to configure the switches by hand, the Splat database is kept current. This way, the sysadmins can easily follow the best practices when managing the switch port configuration.

Guru Sessions

IPSec

Hugh Daniel, Linux FreeS/WAN Project

Summarized by Siddharth Aggarwal

Since this was a guru session, it involved direct questions to the speaker by the audience. Hugh Daniel began by saying that IP networking is antithetical to IPSec. Most system administrators find implementing IPSec problematic because the setup is not done correctly. So the speaker explained a test setup for a Web site in which all the machines are physically kept together.

Daniel clarified some misconceptions about IPSec – for example, that it is technically a transport mechanism and not a technique for authentication or encryption. It is the job of Internet Key Exchange (IKE) to maintain pre-shared secrets and RSA keys. Daniel introduced various ways of deciding if two hosts can talk to each other: pre-shared secrets, RSA keys, X Auth, X.509, etc. Also, a brief introduction about a PDA that runs Linux, called Zaurus, was given.

Daniel then introduced the Wavesec technology, which uses a combination of opportunistic encryption (OE), dynamic DNS, and DHCP. OE enables you to set up IPSec tunnels without coordinating with another site administrator and without hand-configuring each tunnel. He also explained the goal of Free S/WAN, which is to provide a host-to-host or network-to-network privacy environment via a distributed database of DNS entries and keys. He explained why FreeS/WAN emphasizes an anti-NAT (Network Address Translation). IPSec fails when packets go through a NAPT (network address and port translation) box, because NAPT mangles the packets.

The session concluded with some links to useful resources:
http://www.freeswan.ca
http://www.wavesec.org
http://www.freeswan.org/talks/lisa-2003

AFS

Esther Filderman, The OpenAFS Project; Garry Zacheiss, MIT

Summarized by Venkata Phani Kiran Achanta

The AFS guru session consisted of questions about large file size support, read-write replication functionality, status of disconnected AFS, back-up strategies, and many other topics as well.

Some people asked whether there were plans to make read-write replication of volumes. Esther said the Coda filesystem does RW replication of volumes (there is no notion of cell in Coda yet), but they were not sure whether it would be available in AFS or not. Garry added that Coda is entirely a research project and is not for use in a production environment.

Regarding disconnected AFS status, Garry said that there was an initial verbal commitment from the University of Michigan to incorporate disconnected AFS functionality into OpenAFS code, but they later backed out because they are heavily into OpenBSD research.

Alf Wachsmann from Stanford Linear Accelerator Center made an announcement about an OpenAFS best-practices workshop being held at SLAC in February.

People were curious to know how MIT and PSC were doing backups. Garry said they were using butc with a bunch of self-written Perl scripts, which need no human interaction. Esther said that they would do a vos dump locally and, with HSM support, would migrate that dump to a repository. She added that there used to be an add-on to Legato a while back. The most popular backup solution for AFS is TSM. Cornell University is working to tie AFS into Amanda.

There were some people interested in using OpenAFS in a grid computing environment, but lack of file support for files greater than 2GB seems to be a limitation for them.

A newbie asked about recovery when an RW volume is lost. Esther said they can always do a vos dump of the existing RO copy of the volume as an RW volume and start using it as if nothing had happened.

Somebody asked whether the 22-character limit in the naming size of volumes would be increased in future releases of OpenAFS. Garry said that there are no plans to increase it, but that there is a workaround using MD5 hashing. Esther added that if they did increase the limit, the old AFS clients would be confused.

Answering a question on ideal client cache size, Esther said that it would mostly depend on the chunk size at their site. Someone asked whether to restart the file server once a week if clients are using it 24/7. Garry said there is no necessity to restart.

There was discussion about MRAFS, which is heavily used by Naval Research Labs; different authentication techniques; and why AFS uses Kerberos.

Like any other open source project, OpenAFS also seems to suffer from lack of "more" volunteer time. The gurus were optimistic about the future of OpenAFS and said that if more volunteers were willing to contribute to the OpenAFS project, there would be much more functionality that could be incorporated into OpenAFS.

MBAs for SysAdmins

Brent Chapman, Great Circle Associates

Summarized by Carrie Gates

Why should a system administrator pursue an MBA? There are two answers to this. The first is the marketing-type answer, which is that it will, on average, add 25–40% to your current salary. The second answer is that it provides a better understanding of the entire business environment, such as finance and personnel, which in turn will allow you to better relate to the concerns of those who work in these other departments.

There are three paths to an MBA: standard full-time courses, part-time courses, and the executive-level MBA. Although the full-time MBA allows a student to complete the degree more quickly, the part-time MBA enables one to keep working while obtaining the degree. Unfortunately, the part-time students often miss out on many of the opportunities available to the full-time students. Conversely, the part-time students tend to be older and have more business experience, and so the full-time students often miss out on learning from discussions with them. The executive MBA is a combination of the two approaches, but is more expensive and is geared toward senior managers (where their company is paying for tuition). Typical courses consist largely of case studies, with a single case study taking up 5–25 pages of scenario. These case studies are used to generate and guide discussion.

The bottom line is that you will get out of an MBA what you put into it. MBAs offer a wealth of learning opportunities, both in the classroom and outside of it, as well as providing the opportunity for considerable networking within the business field. For those who are interested in pursuing a management path, it can also provide an extra credential when applying for management positions. Beyond this, it can provide someone who has a technical background with the confidence to pursue career paths such as CIO or CTO.

PKI/Cryptography

Greg Rose, QUALCOMM, Inc.

Summarized by der.hans

Rose mentioned that there are two types of cyphers, symmetric and asymmetric. Symmetric cyphers, such as DES and Rijndael (accepted as AES), are the traditional type of cypher and there is evidence they were used as far back as 2000 BC. Symmetric cyphers use the same key both ways. Asymmetric cyphers, a.k.a. public key cyphers, such as RSA, use different keys for encryption and decryption.

All the old cell phone cryptography was broken. Rose was first to break some of the algorithms. The new 3G cell networks use different but equivalent ciphers. All use 128-bit keys. One of the problems with the old algorithms is that they were created behind closed doors. Review of the algorithm and the code is important to be certain an implementation is secure.

Rose gave several examples of cryptography that was weak due to shortcomings in the algorithms or errors in the implementation. He mentioned that most Web server administrators know that most of the CPU is used in putting the padlock on the browser, not in transmit-

ting the data. For instance, small keys take constant time because they fit 32-bit CPUs, but large keys have to be broken up and done "longhand." Going from 1024 bits to 2048 bits cubes the time needed to generate the key. Computational time equals lost battery life for cell phones.

Linux

Bdale Garbee, HP Linux and Open Source Lab/Debian

Summarized by Hernan Laffitte

Topics such as the SCO lawsuit and the end-of-life announcement from RedHat figured prominently in the first segment of the talk. Mr. Garbee explained that HP's first concern is supporting its customers, many of whom run SCO and RedHat, and also promoting the use of open and free standards.

Another important issue facing Linux developers is that a number of independent software vendors (ISVs), such as Oracle, and hardware manufacturers, such as HP, will only certify their products against a small number of commercial Linux distributions. This is a result of the economic realities of setting up QA and support, and the fact that no two Linux distributions seem to use the same kernel.

Setting up standards for Linux distributions will help alleviate this problem, and Linux 2.6 will have a feature set closer to what many ISVs want. Other companies, however, will want to add different features to the kernel. And there is always the issue, even if everybody agrees on the current standard, of negotiating which features will go into the next one.

Economic realities also conspire against selling Linux to the general (read: non-techie) public. For example, putting a line of Linux-powered machines on the shelves of a computer store involves a lot of expenses: printing a different set of manuals, different packaging, tracking a different part/model number from the factory down to the store in Kalamazoo . . . it's all expensive, even if the OS is free.

Actually, the money involved in the OS licensing is not as much as many believe. It's simply a question of demand. The demand is growing, but it's still not there. The marketing people would say, "Come back next year if you have 10 times the current volume." Also, Mr. Garbee commented jokingly, whatever distribution you choose to sell, the rest of the Linux users will hate you.

The juggernaut is rolling in the right direction, though. For example, HP recently released a BIOS patch for one of its systems specifically to improve Linux compatibility, and is working to improve Linux compatibility in general.

Mr. Garbee also talked about his experience in porting Linux to the Itanium platform. A big percentage of the Itanium 2 systems shipping in the first quarter of production were Linux, and the trend increased in the second quarter. Linux is also very popular in Itanium workstations, and HP-UX customers like having the possibility of replacing old PA-RISC machines with Itanium without having to make any changes to the software.

In addition to his work on UNIX and Linux, Mr. Garbee is a prominent member of the amateur satellite community. The talk touched briefly on the issues of Linux in space (it was used in an experiment on the shuttle, and will also be used in the amateur radio experiment on the international space station). Mr. Garbee also discussed the technology of amateur satellites. He stressed that there is a constant need to simplify the hardware requirements. The 1802 processor used on many satellites, for example, runs at 100 KIPS (kilo instructions per second). Things don't happen very fast in space, so there is no need for lots of processing power. And amateur satellites are a fun hobby in part because the types of problems faced when working on 8-bit micro-controllers are quite different from the ones encountered when working on Linux for Itanium systems at HP.

Automated System Administration/Infrastructure

Paul Anderson, University of Edinburgh; Steve Traugott, Infrastructures.Org

Summarized by Kevin Sullivan

Configuration management seemed to be a central theme of this year's conference, and it took center stage at this guru session. A packed room gathered to hear Paul Anderson and Steve Traugott give their opinions on the state of automated system administration. The hour-and-a-half session was very informative, with a great discussion of theory interspersed with various tools administrators are using today.

The discussion quickly turned to "push" vs. "pull" systems in configuration management. Steve and Paul contended that many people who think they have a "push" system actually have a "pull" system. Steve said that a "pull" system is advantageous because it reduces the threat of divergence, since a machine will properly configure itself before it offers any services. Paul added that "pull" systems don't require any knowledge about the state of the machine at configuration time, so offline hosts will not be missed.

Paul went on to describe a configuration fabric consisting of hardware, software, specifications, and policies. Soon the room was buzzing about the tools used to build and maintain this fabric. Each tool employed a different paradigm: Anderson's tool, LCFG, tells a host what it wants to look like, while Traugott's ISConf – originally a quick fix aimed at building up an infrastructure – tells a host what to do.

Also discussed was "The Test," in which you imagine taking a random machine that has never been backed up, destroy

it, and then have its services recovered within 10 minutes. Both Paul and Steve note that their infrastructure management systems pass The Test.

Professional Growth and Development

David Parter, University of Wisconsin, Madison

Summarized by Marko Bukovac

David Parter led an excellent free-flowing discussion covering several topics of interest to system administrators in industry and academia. The first topic came from mid-level administrators who were interested in knowing how to mentor their students and colleagues. Senior administrators recommended that students develop a range of technical skills, including "people skills," which is a big part of the job. In addition, mentors should always treat system administration as a legitimate profession (it is not always seen as such by users). Students should be encouraged to communicate with their mentors (who should set some time aside to work with students) and ask questions using SAGE online resources, such as the Web site, IRC channels (*irc.sage-members.org #sage-members*), and the mailing list (*sage-members@sage.org*).

Mid-level admins mentioned that logical thinking and thorough knowledge of the fundamentals (though it can sometimes be hard to define what fundamentals really are) are perhaps the most highly valued skills in the field. Some admins mentioned that students' fear of "breaking things" slows their growth and that they should be encouraged to experiment, only not on the main servers. A debate about the relative importance of depth cersus breadth concluded that they are equally important.

To keep their job fun and interesting, some administrators would like their jobs to change with time and include more research. While there is no overall solution to this, as it is company-dependent, some senior admins recommended books, such as O'Reilly's *Love Your Job*, and some recommended writing and sharing tools, which can then lead to more communication between companies and to more research on the subject. Many recommended getting books and taking classes on time management, since this is a skill that many admins (especially younger ones) lack. Giving small group tutorials and then expanding might lead to giving a tutorial at a LISA conference.

Many of the admins wondered how to take control of their careers. Senior admins saw themselves in the position of having to join the management and abandon technical duties, to the dismay of most of them. The main suggestion in this case was to check with HR (even before getting hired) to ask about job growth and possible future duties. Some admins considered switching from university environments to the "real world" but feared that they were not ready for it (myth: work at the university is not as important and difficult as work at the corporation). All of them were encouraged by the corporate admins, who said that academia is not at all different from the corporate world.

The session concluded with a discussion about personal career plans. Everyone should have a personal career plan and an idea of what their dream job would be. One should not be afraid to ask the employer about future plans and how the job will evolve. In their work, admins need to manage users, systems, and management, and many find it very tricky to manage all three successfully. Some senior admins suggested taking nonsystem administration courses, such as management, as well as documenting all political decisions (resources, time, budget) made by their supervisors. Managing management is a vital part of the job, and senior admins recommended learning this skill.

Invited Talks

Outsourcing: Common Problems and Current Trends in the Outsourcing Industry

John Nicholson, Shaw Pittman LLP

Summarized by Emma Buneci

Outsourcing has been a hot topic over the past few years, and John Nicholson presented an excellent overview of the topic. Outsourcing is defined as the long-term contracting of an information system or business process to an external service provider in order to achieve strategic business results.

The top-tier providers are IBM, CSD, EDS, and ACS, while in the second tier we find Perot Systems, Accentrue, CGI, Unisys, and Lockheed Martin Siemens, as well as other consulting firms. As an interesting change, the hardware providers, such as Dell, Compaq, and HP, have all been moving into providing services for their clients. As offshore providers, there are typically the larger Indian companies, such as the Tata Group. IBM is the dominant player in the global market and was able to maintain this position by drawing on its own strengths and taking advantage of the leadership and accounting problems at other companies.

After outlining the seven major trends in the outsourcing industry – mid-sized markets; outsourcing of IT, business process, and business transformation; offshore outsourcing; shareholder influence; renegotiation of existing agreements; piecemeal deals; and the changing nature of IT departments – Nicholson discussed problems with outsourcing. The three major issues seem to be timing, customer perspective, and perceived poor customer service.

Rushed negotiations, differing expectations, and poor communication with end users lead to a very unhappy relationship. In order to minimize problems, any outsourcing deal must be treated with the same care and planning as buying a used car. It makes sense to

talk to multiple vendors because talking to only one vendor will undercut negotiating leverage. The customers must be clear about document scope, service levels, and cost. Pricing must be clearly specified before signing the deal. Assumptions and dependencies must be avoided: If there is any assumption or dependency written in a deal, it must be specified how it will imply a change in the price.

Using an independent deal consultant is highly recommended; in the same way that a car mechanic is crucial to buying a used car, a consultant will know how to look for and evaluate problems that you might not see. The final piece of advice: "Communicate, communicate, communicate!"

A Case Study in Internet Pathology: Flawed Routers Flood University's Network

Dave Plonka, University of Wisconsin, Madison

Summarized by Jason Rouse

Dave Plonka gave an enlightening talk on the story behind the flooding of the University of Wisconsin's public NTP server. On May 14, 2002, Dave was reviewing network logs. He was quite surprised to find a nearly 90,000 packet-per-second forwarding rate through one of the university's public NTP servers. Seeing that the source port was fixed and IP addresses associated with the flows were random, Dave's first guess was a distributed denial of service. To combat this, he placed university-local blocks on the ingress routers.

A month later, however, Dave was surprised to find the access control lists dropping over 250,000 packets per second, all with the same IP profile! This time, Dave decided to escalate the investigative procedure. He chose the two top talkers and emailed them directly, received immediate responses, and found the commonality was a Netgear product. After searching for the model number, Dave located a few references to the product, one such reference, to ICSA Labs, mentioning that the Netgear router did not include a battery-backed clock.

Plonka's next step was to examine the hardware and software directly. He downloaded the firmware available from the Netgear Web site. After a cursory examination, he found that the Netgear firmware included the IP address of one of the university's NTP servers. As soon as he made this discovery, Plonka contacted Netgear directly via their help desk and customer service channels. After a number of days without response, Plonka phoned a Netgear executive directly.

Plonka then guided the formation of a team consisting of Netgear employees, university employees, and independent experts. This key step ensured that the problem could be addressed in a way that was fair to the university, the company, and the Internet community as a whole. The initial response was to point users to an "Instant Code" update, available from the Netgear Web site. Interestingly, this code had been available for some time, but had not been widely advertised or adopted by the product community.

Understanding the difficulties involved in communicating to such a diverse user group, the review team pursued other options in order to mediate the large amount of incoming NTP traffic. Finally, the team concluded that the implementation of an anycast NTP time service at the Wisconsin site could successfully handle such a traffic load. As of this writing, Netgear and the University of Wisconsin have undertaken a project to provide this anycast deployment.

Plonka's experiences were summed up in two pieces of sage advice. First, involve all parties in any dialogue when searching for a solution. Second, recognize that the Internet is a shared resource based on the good citizenship of many, many users, and act accordingly.

Organizational Maturity Models: Achieving Success and Happiness in Modern IT Environments

Geoff Halprin, The SysAdmin Group

Summarized by Jason Rouse

Geoff Halprin has the courage to say what we've all been thinking: Sysadmins have a hard work life. What with the economic downturn since the dot-com bomb, the reactionary posture we have to assume in order to meet fluid business goals, and the organic nature of system and software development, sysadmins truly have a difficult juggling act in front of them.

Halprin described system administration as a constant quest for reliability, availability, and serviceability. As a part of this quest, system administrators must combat the often organic growth of systems and software, engineering fixes in order to maintain systemic improvements. Halprin also mentioned the distinct lack of recognition for systemic improvements, leading to a lack of work in this area. This cycle of low reward and organic growth leads to systems that age badly, requiring more and more work to maintain them as time passes.

Halprin also understands that system administrators must deal with constant change. Systems creep toward states of increased entropy, and Halprin shows how system administrators can combat this gradual degradation. By having an exact worst-case cost associated with downtime, Halprin believes that system administrators can communicate more effectively with management, achieving management buy-in. Management buy-in improves overall workflow management, thus lightening the workload on the system administrator. Management buy-in also allows a larger measure of root-cause analysis, so often missing in highly dynamic workplaces.

Finally, given that systems will break, how do system administrators minimize or control failures? Halprin's answer is to

ensure that system administrators continuously move toward a proactive stance, constantly re-evaluating their workflows and incident handling.

Network Telescopes: Tracking Denial-of-Service Attacks and Internet Worms Around the Globe

David Moore, CAIDA (Cooperative Association for Internet Data Analysis)

Summarized by Carrie Gates

David Moore described network telescopes, what they are and how they can be used. The basic premise is to take a chunk of IP address space that receives little or no legitimate traffic (or receives traffic that can easily be filtered) and analyze the traffic that it receives. All of the traffic seen by that space (other than any known, legitimate traffic that has been filtered) represents some unusual network event.

For example, network telescopes can be used to examine the presence of spoofed-IP denial-of-service attacks on the Internet. Say you have a /8 network that you can use as a network telescope. This address space represents 1/256 of the Internet. If an attacker is DoSing some target using spoofed IP addresses that have been randomly chosen, then the telescope should see approximately 1/256 of the response traffic, as that is the likelihood that an IP address in the telescope address space has been chosen. By analyzing this information, we can infer the number of DoS attacks occurring on the network, as well as information about the attack itself. Over the past two years, for example, there have been approximately 40 DoS attacks against /24 networks per hour. The majority of these consisted of SYN floods against HTTP services.

Network telescopes can also be used to study the spread of Internet worms. Assuming that there are no biases (or bugs!) in choosing the next IP address to infect (that is, any target IP address has been chosen randomly across the entire Internet address space), a network telescope can expect to see 1/256 of the scanning traffic generated by any one instance of the worm. It was seen with Code Red that the majority of the infections were ISPs providing home and small-business connectivity. Within 10 hours, Code Red had infected 360,000 hosts, indicating that there was no effective patch response to the spreading infection. Additionally, Code Red remained inactive for 12 days and then became active again. It was well known that the worm would reactivate on August 1, and so there was a lot of media coverage. Despite this, the majority of previously infected machines were not patched until August 2, after being reinfected.

For users interested in building their own network telescope, all that is required is a globally accessible network address space that can be monitored. Suggested tools for analyzing the captured data include FlowScan (for analyzing flows), CoralReef (for analyzing packets), and AutoFocus (which analyzes both flows and packets). The effectiveness of the network telescope will depend largely on the amount of address space that can be monitored. The larger the address space, the more traffic it will be able to analyze. For example, a /8 network represents 1/256 of the Internet, but a /16 will only see 1/65536 of the Internet and so will have considerably less chance of seeing any traffic that has been randomly addressed.

Network telescopes, especially when deployed across a large address space, can provide significant insight into non-local network events.

Internet Governance Reloaded

Paul Vixie, Internet Software Consortium

Summarized by der.hans

[Note: Due to the fires in Southern California, Paul Vixie was unable to attend LISA '03, so kc claffy substituted for him on short notice and used his slides.]

kc claffy

kc explained that governance is needed for such shared resources as IP addresses, domain names, AS numbers, and protocol numbers. Governance means that those who are affected by a decision get to help make that decision. Stakeholders are those who hold/own/use/control the resources and those who allocate the resources.

The first example of shared resources kc mentioned is global routable IP. Demand appears to be higher than scale allows. ARIN/RIPE/APNIC/LACNIC are constantly searching for an equilibrium between routing table size and minimum allocation size.

The next example was Verisign's typosquatting with SiteFinder. While the talk wasn't specifically about Verisign, SiteFinder became the primary topic, with lots of input from the audience.

Verisign doesn't see itself as the steward of public resources; it sees itself as the owner of those public resources. Unfortunately, the contract with Verisign apparently doesn't specify which view is correct. Both kc and Vixie were in Washington, D.C., for the first ICANN security meeting about the Verisign typosquatting. kc pointed out that ICANN responded with impressive speed and integrity with regard to Verisign's typosquatting, which was turned off 19 days after Verisign instituted it.

Responding to customer requests, ISC created a patch for BIND9 to block SiteFinder. China opted out of Site-

Finder by null-routing Verisign's IP for SiteFinder. kc described SiteFinder, ISC's BIND9 patches, and China's blocking of SiteFinder as examples of cybernetic warlordism.

Several times, kc suggested getting involved, emphasizing how close we are to the action. This is Internet policy being made right before our eyes, and we can participate. She reminded everyone to be courteous, mature, and professional. We can help make the rules.

Vixie says SiteFinder's losers are registrars, domain registrants, spam victims, Web surfers, other typosquatters, users of non-Web protocols, and the Internet governance trust model. He challenges Verisign to provide diverse and specific examples of entities other than Verisign that benefit from SiteFinder.

Vixie predicts lawsuits and countersuits before the SiteFinder and stewardship vs. ownership issues are resolved.

Many members of the audience mentioned that the governance organizations need to be non-national and specifically non-USA.

Resources:

http://www.icann.org/tlds/agreements/verisign/
http://www.icann.org/announcements/announcement-17sep03.htm
http://www.icann.org/correspondence/twomey-to-tonkin-20oct03.pdf
http://secsac.icann.org/
http://www.icannwatch.org/
http://www.isoc.org/
http://www.ntia.doc.gov/
http://www.stanford.edu/class/ee380/Abstracts/031001.html

High Risk Information: Safe Handling for System Administrators

Lance Hayden, Advanced Services for Network Security (ASNS)

Summarized by Jason Rouse

Lance Hayden began by explaining that most information, if viewed in the proper context, could be damaging and, therefore, high risk. Examples of such information could be names, addresses, credit card numbers, or phone numbers. Since system administrators are often tasked with securing and maintaining systems on which this data is stored, Hayden believes that it is in the best interest of system administrators to make themselves aware of the ongoing work in regulatory legislation and practices.

Hayden gave an excellent overview of current and future legislation and interpretations, focusing on their impact on system administrators. He produced a world map, showing the increase in data privacy legislation across the globe, and then outlined a six-step iterative process to enable system administrators to educate themselves about the high-risk information they might handle and then inventory and build a strategy for dealing with that information. Review and alignment of IT with core business goals is a key factor in this process.

Summing up, Hayden introduced the "true" OSI model – one where the "financial" and "political" layers heap upon the application layer. In this environment, Hayden argues, system administrators must be aware not only of their place in the legal and social infrastructure but of their potential liability and methods to mitigate this risk.

Panel: Myth or Reality: Studies of System Administrators

Moderators: Jeff R. Allen, Tellme Networks, Inc.; Eser Kandogan, IBM Research

Panelists: Nancy Mann, Sun Microsystems; Paul Maglio, IBM Research; Kristyn Greenwood, Oracle; Cynthia DuVal, IBM Software

Summarized by Kevin Sullivan

This session assembled three researchers from major corporations, each of whom studies the actions and responsibilities of system administrators. For some it was surprising to learn that there is a lot of research devoted to usability within the system administration community. It was quickly suggested that "system administration is a misunderstood profession, both from inside and out." The session focussed on how usability experts can study what system administrators do, and how system administrators can employ usability research tools to improve how they do their jobs.

The panel suggested that there are four aspects to system administration: psychological, technological, cognitive, and social. These aspects can be studied in various ways, including diaries, lab studies, questionnaires, and observation.

Kristyn Greenwood discussed how she conducts usability studies known as "DBAs in the Wild." This was a naturalistic observation of DBAs and SAs where the researchers recorded every action of the user. The primary aim was to provide this information to product development teams so that they could improve their products based on the feedback from these sessions. Interestingly, Kristyn found that SAs spent 18% of their time on group coordination compared to 27% on actual troubleshooting.

Paul Maglio spoke on his study of internal Web administrators at IBM. His focus was on the methods of communication used in problem solving, namely, phone or instant messaging. Paul also noted that large portions of time are spent on collaboration and communication. He suggested that tool development focus on collaborating and allowing the user to shift effortlessly between systems. A particularly insightful comment was that command line interfaces do not provide the situational awareness that is important to many complex tasks.

Nancy Mann spoke about her study, "Who Manages Sun Systems?" This study aimed to develop a profile of a system administrator, including experience, tasks, goals, motivators, and tools. Infor-

mation gathered in the process will also be provided to software design teams to improve the overall experience for system administrators.

It is quite apparent that system administrators need well-designed tools just as much as novice users. This panel showed that there are people devoted to improving the computing experience for all types of users. Usability as it applies to system administrators is very different, but just as important.

SPAM MINI-SYMPOSIUM

Summarized by Steve Wormley

The first part of the LISA '03 Spam Mini-Symposium consisted of two presentations.

Emerging Spam-Fighting Techniques

Robert Haskins, Computer Net Works and Rob Kolstad, SAGE

The authors started with a quick survey of the audience which found that most receive over 30 spam messages per day. The first point mentioned was that one of the problems with spam is the definition. The end users know spam when they see it, the ISP knows it uses resources, and the spammer knows it makes them money. Yet, spam is hard to define. A second problem is that bulk email is cheap for the sender. Of course, the spammers say "Just hit delete," but we all know it's not that easy for the recipient. Bandwidth costs continue to increase and the consumer bears the cost of the email. For one example, Rob Kolstad apparently receives 400 spam messages per day.

One interesting point that was made is that spam is fraud. Spam has misleading subject lines and advertises fraudulent products. Also, opt-out in spam isn't a way to escape, and opt-in is a joke. And finally, spam almost always hides its sites and sources. More spam problems include that spam is hard to winnow, it overloads mailboxes, and the messages themselves are annoying. And sending spam is easy there are fairly low barriers to entry.

The presenters believed that most spam is already covered by existing laws: fraud is already covered, as is trespass. New laws for other email will be expensive and difficult to pursue. In addition, the issues of free versus regulated speech versus privacy will be difficult to balance going forward. And the root of the issue is that spammers spam because people

The Spam Mini-Symposium

buy stuff from spam: at least one survey said 7% of recipients have ordered from unsolicited e-mail.

How spammers are still sending mail varies. There are still open relays spammers can use. More these days are also hijacking PCs to send their spam. Some service providers also allow spam via "pink contracts," allowing them to avoid typical terms of service. The presenters mentioned that even the smallest service providers should be able to block most outgoing spam should they choose to.

Spam turns out to be an arms race. Spam is not easy to stop because most spam comes from forged sources, hijacked systems, drive-by spamming from wireless, gypsy accounts (set up, spam, and leave), and the content (what the spam points to) is often not traceable.

The practical solutions consist of education, technical solutions, legal solutions, or social solutions. Education is such things as getting people to shut down open relays, which is often an issue in developing countries, and having people secure their home PCs. One of the better legal, social, and economic methods is to enforce existing laws.

On the technical side, it is fairly easy to handle outbound spam: simply require authentication of the user sending the mail. The inbound side of spam is where the problem is. The first recommendation is to replace RFC 822. Other ideas are things like blacklists, whitelists, distributed collaborative filters, onetime or limited-use addresses, challenge response, forcing the sender to compute something, filtering services, scoring and rating products(SpamAssassin), enterprise plug-ins, and Bayesian filtering. Bayesian filtering uses probability theory to perform its spam checks; CRM 114 looks at 16 observations for each word and works fairly well. Blacklists are good for providers. Reporting spam is important so that things can get fixed where possible.

Adaptive Filtering: One Year On

John Graham-Cumming, ActiveState

John's presentation emphasized the fact that the best way to control spam was to increase barriers to entry. One way to do this is with filtering. Products such as POP file use adaptive filtering to gauge the level of spamminess of an email.

One of the reasons spam filtering is a big issue is the "Grandma Problem": now that Grandma is starting to get spams, filtering them is becoming more important. Many filters exist today both in open source and commercial products. John expects that by 2004 every mail client will have adaptive filtering.

The primary adaptive filtering issues are the man-in-the-street usability issues, false positives, overtraining, oneman spam, and internationalization. Things such as integration into the mail client, auto whitelisting, and the filter guarding against false positives help. However, overtraining needs to be handled by the user, who may click the "spam" button on far too many messages, causing the system to think everything is spam. For

internationalization the filter system needs to understand how languages work and how punctuation and tokenization should be handled.

Most spammers are trying to overwhelm filters with good words which are then hidden using various HTML tricks such as comments and invisible ink. As the arms race progresses, the spammers try more things, and the anti-spammers sometimes get more fingerprints. The question is, do filters make spam more effective, since at least one spammer has claimed that filters helped him by reducing complaints.

Panel Discussion: Current Best Practices and Forthcoming Advances

Part 2 of the Symposium was moderated by Dan Klein, with the presenters from the first spam session and three additional participants.

First there was a brief presentation by Ken Schneider of Brightmail. Brightmail provides a spam filtering package with service and products. They estimate that over 50% of email is spam now. The majority of the spam messages advertise products, and another large category of spam is adult advertisement. Brightmail uses a set of decoy accounts on client systems to collect spam, which their operations center then classifies, and they creates rules which are sent back to the clients.

Other panel members were Laura Atkins, president of the Spamcon Foundation, which is working to keep mail usable, reduce false positives, assist with legal fees for anti-spammers and file suits against spammers; and Daniel Quinland, the author of SpamAssassin. SpamAssassin is an open source product which uses anything that works to stop spam. He also encouraged everyone to implement SPF, at *http://spf.pobox.com/*.

Who writes the software for spammers? The general consensus was that it was commercial organizations, some software often shipped with anti-spam software to test the spam before it's sent.

One of the more contentious issues which came up in the round table was the issue of challenge response. The consensus from the panel was that none of them thought it was a good idea. Some of the issues included fake challenges from spammers, spammers faking a known good address, spammers using a sweatshop to accept all the challenges, and the general annoyance to people who send you email for legitimate reasons.

The panel then was asked about blocking customers with viruses by ISPs. They felt it was useful for customer ISPs but not necessarily co-location facilities. There was also some concern that it could affect the common carrier status of an ISP.

How do people handle users who report spam that is actually requested email? Brightmail in this case requires a minimum threshold for something to be classified as spam.

What about the spam program writers? Apparently in many cases the programs are legitimate bulk mail tools for various companies. Rob Kolstad pointed out that programmers cannot be responsible for content.

Is spam legislation needed? Rob Kolstad felt that the main things was that spammers should not be able to say what they are doing is legal. Laura Atkins responded that the DMA (Direct Marketing Association) is in the pockets of the people on Capitol Hill. The DMA does not want opt-in for email. They also don't want this to become the requirement for future marketing.

Coping with the Disappearance of Network Boundaries

Peyton Engel, Berbee

Summarized by Jason Rouse

Peyton Engel highlighted the advancement of technologies such as VPNs, distributed computing, and load-balancing boxes and how the introduction of these technologies has blurred the boundaries of traditional IT roles and network demarcation points.

When using these technologies, one has to ask questions about liability and due diligence. If a distributed computing cluster is compromised and is used to scan or compromise other networks, who is responsible? Since VPN technology effectively extends network boundaries to arbitrary limits, how do we handle cybersecurity threats in this new environment? This, Engel argues, is the world into which we will be heading in the coming months.

As organizations begin to incorporate these new technologies, Engel believes that security is frequently overlooked, or existing security solutions are trusted to operate in environments for which they were never designed. Engel dealt with these questions and more, citing the need for competent, well-rounded security practitioners and the defense-in-depth strategy of multi-level, multi-vector infrastructure and employee protection. Engel also noted the growing fluidity of administrative domains, for example merging two corporate networks.

Engel believes that this new environment will provide both challenges and insights into tomorrow's best practices, and that these issues will become the groundwork for system, network, and security administrator approaches in the coming years.

Security vs. Science: Changing the Security Culture of a National Lab

Rémy Evard, Argonne National Laboratory

Summarized by Carrie Gates

Rémy Evard gave a presentation on changing the culture of a research science lab to incorporate secure practices. Such a change in culture requires several stages, starting with reaction mode and

then moving through project mode and institutionalize mode before achieving an ongoing program.

The reaction mode, in which they started, consisted of a climate where there were no policies or support for security. For example, there were no policies restricting the use of cleartext passwords. The result was a number of intrusions, and poor results from security auditors. The problem was the culture – the belief was that effective security would keep users from being able to do what they wanted to do, and so there was no support for security, which translated into no funding and no direction.

The catalyst for change, causing them to enter the project mode, was a new director who took security more seriously and asked for an internal report. The report's recommendation was for the development of a security policy committee. This committee was formed with the goal of fixing everything (!), followed by passing another audit. A key part of attaining this goal was the development of policies. And a key part of drafting acceptable policies was holding general discussions of the policy in town hall meetings with the entire lab. This helped to alleviate the fear that people would not be able to perform their work, and helped to create the buy-in required to have the policies work. By the end of this stage, an internal risk assessment had been performed, ongoing internal scanning for vulnerabilities was being performed, and firewalls had been deployed.

There was a gradual move into the institutional mode after this. Here the goals were to reduce the effort required to achieve effective security (while still keeping up the energy for it) and to prepare for the next audit. The technical activities consisted of improving both consistency and integration and deploying practical solutions. During this stage, an intrusion detection system was also deployed, which has been found to be useful for detecting large-scale scans and viruses. By the end of this stage, the auditors returned and performed both a management review and a technical review. The resulting grade: "effective" (A).

There were three points Evard felt were key factors in their success in deploying appropriate security policies and infrastructures. The first was that the highest level of management "got it," and that they bought into the process and the necessity of having security. The second was that audits work and provide valuable motivation and feedback. The third factor was that everyone helped and became involved.

Talking to the Walls (Again)

Mark Burgess, Oslo University College

Summarized by Siddharth Aggarwal

Mark Burgess discussed the evolution of pervasive computing and the challenges it could pose to system administrators in the years to come.

He introduced the topic by looking at smart houses and smart cities, which will make extensive use of pervasive computing in the future. According to Burgess, pervasive computing brings up new challenges for a system administrator because of the diversity of devices that have to be managed, coupled with the high density of communication. Because of limited consumer demand, the slow introduction of these devices will tend toward a non-standardized, heterogeneous computing environment. This also leads to a lot of security issues.

Burgess grouped the challenges posed by pervasive computing into three categories: diversity, stability, and sociology of interaction. When implementing pervasive computing, a key decision to be made is who should control the system. Who decides the policies and controls the resources? This leads to another question: Should humans and computers cooperate with each other or compete against one another? Should a device adapt to the environment, or should the environment adapt to the device when it comes into a system? Burgess discussed various techniques, such as game theory, for modeling interaction between such systems.

Burgess finished by introducing modern concepts like the pull model of communication between systems having an emergent behavior, human-computer swarms, and pseudo-hierarchical social swarms. The emphasis is on systems having probable control, probable risk, and probable behavior rather than absolute control. He concluded by saying that the world is controlling us as much as we are controlling it. The challenge lies with system administrators to find stable points for equilibrium.

Through the Lens Geekly: How Sysadmins Are Portrayed in Pop Culture

David N. Blank-Edelman, Northeastern University

Summarized by Ari Pollack

David Blank-Edelman presented a highly entertaining talk on the portrayals of sysadmins in US popular culture. In the minds of the public, sysadmins typically get lumped into a broader "computer person" category along with programmers and hackers/crackers, so the examples in this talk included both sysadmin and sysadmin-related characters, mostly from the movies. David noted that portrayals of sysadmins broke down into three polarities: "competent or incompetent," "good or evil," or "hip or really uncool." Examples were shown of each, much to the amusement of the crowd.

After this demonstration, David suggested that these portrayals are closely tied to the public's views on computing and technology in general (e.g., people's views of computers as being totally competent or incompetent get projected onto sysadmins). Given that people accept the stereotypes they see in popular culture when they interact with sysadmins on a daily basis, David ended

with tips on ways to respond to these stereotypes in the workplace.

How To Get Your Papers Accepted at LISA

Tom Limoncelli, Lumeta Corporation; Adam Moskowitz, Menlo Consulting

Summarized by Carrie Gates

Limoncelli and Moskowitz based their talk on their experiences as program committee paper referees. Their first advice to potential authors was to read and *follow* the instructions on the call for papers.

The paper submission process for LISA consists first of submitting an extended abstract (not a full paper) and a paper outline. An "extended abstract" is a short version of the full paper, consisting of about 4–5 pages (not 4–5 paragraphs!). It should not be a teaser but, rather, should provide enough details to allow the committee to make a decision, without providing details of required background knowledge.

Abstracts are then reviewed by the committee members. Each paper is assigned to 4 or 5 readers, who rank the paper on a scale of 1 to 5 in various categories, such as the quality of writing and appropriateness to the conference. The committee meets as a whole and reviews the rankings of the various papers, accepting the papers with obviously high scores, and rejecting papers with obviously low scores. The committee then reviews each of the remaining papers until a final program has been designed.

The three main criteria for getting a paper accepted at LISA are:

1. Is the work worthwhile? (For work that is publishable but not appropriate for LISA, the reviewers will suggest other forums for publication.)
2. Has it been done before?
3. Can the author write well?

What makes a good paper? First, the potential author should note that the purpose of the refereed-papers track at LISA is to advance the state of the art in system administration. Otherwise good papers might be rejected if they do not meet this criterion. Alternatively, an author can be asked to give an invited talk instead (ITs tend to be on hot topics or by cool people). The author should recognize that the audience is highly technical and write for this audience. If there is any confusion about the level at which a paper should be written, review the papers that have been published at previous LISA conferences (available on the USENIX Web site).

In terms of style, the author should introduce the topic immediately, and then proceed to explain the terms or process or arguments. This allows the reader to know immediately what the paper is about, rather than needing to read several paragraphs before finding the actual topic. Also, the author should explain why the work is original, showing how his or her work is different from (or, hopefully, better than!) work that others have done in the same area. (All authors should list their references in their extended abstracts – this is a pet peeve of some of the program committee.)

In summary, a good paper is clearly written, concise, relevant to LISA, and advances the current knowledge in the area of system administration. It clearly shows the data, methodology, and results, and it discusses related work, showing how the current approach is different from or better than previous approaches.

Security Lessons from "Best in Class" Organizations

Gene Kim, Tripwire, Inc.

Summarized by Carrie Gates

Gene Kim gave a presentation on some research he has been doing on the security practices of "best-in-class" organizations, such as Verisign and the New York Stock Exchange. His goal is to determine the characteristics of a best-in-class organization, and how these can be achieved in other organizations.

Best-in-class operations and security organizations can be recognized by four criteria. First, they have the highest server to system administrator ratio, often with 100+ servers per administrator. The second characteristic is that they have the lowest mean time to repair, as well as the highest mean time between failures. The final characteristic is they demonstrate the earliest integration of security into operations (when compared with other organizations).

Many of the problems encountered by organizations today are created by people. For example, the IT department often does not know about changes that have been made by the security department. This results in an adversarial relationship between security and operations instead of a close working relationship. To further complicate matters, many downsized companies have developers instead of administrators maintaining production servers. Finally, documentation is often not performed, resulting in only a couple of people in the entire organization who know how things really work.

This situation affects how work is performed, resulting in constant firefighting rather than proactive server management. This further results in situations where no two servers are the same, complicating the system administration practice.

By comparison, best-in-class organizations have controls embedded in security and operations to manage change. These organizations have identified what they consider to be the key issues (e.g., outages with a long remediation time, inconsistent system footprints in 1000+ servers running critical business processes), and have developed approaches to controlling these issues (e.g., integrity scans every 10 minutes for business continuity, regular audits to determine

whether system footprints across servers are identical).

The main observations are that best-in-class organizations have developed practices that make it easy to understand, know, and recover to good states in the system. Additionally, they have developed proper processes and procedures for managing change, rather than taking an ad hoc, firefighting approach to the process.

What Washington Still Doesn't Get

Declan McCullagh, CNET News.com

Summarized by William Reading

Why do we need Washington? They provide national defense and handle foreign affairs and interstate commerce, among other things.

However, Washington also wants to regulate where it is actually difficult or impossible to do so without a number of very negative implications.

Although it was struck down, the Communications Decency Act was one of Congress's first attempts at online censorship. It banned "indecent" or "patently offensive" words. As former Sen. James Exon (D-Neb) said, "This is the time to put some restrictions or guidelines on it."

Washington politicians, Bill Clinton among them, also suggested having a sort of "V-Chip" for Internet access.

Al Gore, who still claims that he "took initiative in creating the Internet," supported an equivalent to the "Clipper chip" for computer networks.

Some politicians do not even realize that some legislation is simply impossible, having indicated that they do not support bills such as "602P," which was a hoax that claimed the U.S. Postal Service would begin to charge for email.

The "Office of Cybersecurity" does not seem to gauge threats very well, with cybersecurity advisor to the White House Richard Clarke resigning over the Sapphire worm.

Rep. Howard Berman (D-Cal) proposed that "a copyright owner shall not be liable in any criminal or civil action for disabling, interfering with, blocking, diverting, or otherwise impairing the unauthorized distribution, display, performance, or reproduction of his or her copyrighted work on a publicly accessible peer-to-peer file trading network" (*http://thomas.loc.gov/cgi-bin/bdquery/z?d107:h.r.05211*).

Others advocate destroying computers: "If we can find some way to do this without destroying their machines, we'd be interested in hearing about that," Sen. Orrin Hatch (R-Utah) said. "If that's the only way, then I'm all for destroying their machines. If you have a few hundred thousand of those, I think people would realize [the seriousness of their actions]. There's no excuse for anyone violating copyright laws," Hatch said.

Stick, Rudder, and Keyboard: How Flying My Airplane Makes Me a Better Sysadmin

Ross Oliver, Tech Mavens, Inc.

Summarized by Robert W. Gill

Ross Oliver has been a sysadmin for 15 years and a pilot for 13. He has logged over 500 flying hours and is almost instrument rated. His invited talk focused on the lessons sysadmins can take from aviation. The talk was relaxed, fun, and chockfull of useful ideas to make the lives of sysadmins easier.

Despite new laws like HIPAA, IT is still very unregulated. Ross presented nine areas in which he thinks IT and sysadmins can learn from aviation. Briefly summarized, his points were:

1. Make use of checklists. Use them as memory aids and as tools to avoid missteps. Checklists allow you to standardize tasks for multiple actors and can be used as a training tool.

2. Prepare for abnormal procedures. Anticipate what things can go wrong and prepare how to deal with them before there is a problem. Drilling is important to ensure that the steps you've worked out are correct and to provide confidence when you need to use the procedures under fire.

3. Perform "pre-flight" planning. Planning ahead reduces in-flight workload and puts all variables on the table. You will save time and effort by making decisions in advance, adhering to a checklist format, and allowing for peer review.

4. Know how things work. A checklist will not cover everything, and instruments can lie. By understanding the underlying technology, sysadmins can better cope with situations that fall outside normal operations.

5. Learn to assess risk. Understand your own biases so that they don't distort your viewpoint.

6. Identify chains of errors, in which several different factors combine to cause an accident. Aviation has, for the most part, routed out most single-cause failures; instead, crashes often result from a series of missteps. Such tragedies often occur after signs of a low-level problem have been ignored.

7. Deal with crew resource management. Command and control structures are, at times, too rigid for the environment. Sysadmins are often soloists, accustomed to working at their own pace. Each group needs to find the right amount of structure (checklists, peer review, etc.).

8. Work toward continuous improvement. Strive to find little things you can do to make things better. Learn from other industries (such as aviation with its 100 years of experience).

9. Beware automation. Automation is best applied to frequently utilized and well-understood functions, but is worst suited to exception handling, since it is

difficult to account for all the possible exceptions.

As technology becomes more involved in public safety, the risks become greater. Ross's talk offered excellent examples of how these steps have helped the aviation industry improve its safety record and how they can be applied to the work of sysadmins.

Security Without Firewalls

Abe Singer, San Diego Supercomputer Center

Summarized by Ari Pollack

Abe Singer presented a look at why firewalls are so popular these days, why they should be used, and why they don't need to be used. A common misconception among technical and non-technical people alike is that you're not secure unless you have a firewall. Firewall vendors want to make you think installation will solve all your problems; in reality, firewalls fail all the time, and they do require a great deal of effort to be configured properly. Misconfigured firewalls can inhibit real productivity and do nothing to enhance security. Additionally, there are no data or statistics about the effectiveness of firewalls.

The SDSC currently takes many security precautions to ensure that their systems will be secure against an attack, even without a firewall. Some of these precautions, such as using restricted sudo or patching early and often, may be commonplace in many organizations, but they provide an added level of security nonetheless and have little to no impact on day-to-day usability. Inexperienced users may do things by accident, and in many cases they do not care about security; they just want to do their work, and will try to get around defenses that make it harder for them to perform their job.

There is a place for firewalls, but they may not be worth the effort for all networks. In some cases, 95 to 100% of the security effort at an organization is spent on firewalls. In reality, this should be closer to 5%. Firewalls can be useful for hosts that can't be secured on their own, such as printers or embedded devices, and they can give an extra layer of protection, but firewalls should not be used as the only line of defense.

Workshop Series

AFS

Esther Filderman, The OpenAFS Project; Garry Zacheiss, MIT; and Derrick Brashear, CMU

The AFS workshop covered many topics: Open AFS roadmap, Kerberos integration, IBM's Stonehenge project, APIs, and other AFS workshops.

Derrick Brashear presented the OpenAFS roadmap:

- 1.3 coming soon.
- MacOS 10.3 support now (on OpenAFS 1.2.10a).
- large file support "coming soon" (actually available, but only limited testing has been done).
- FreeBSD and OpenBSD ports are coming along nicely.
- Linux 2.6 kernel is problematic with respect to the interface used by PAGs (Process Authentication Groups). IBM Germany and SUSE have been working together some on this as well.

The second theme was managing Kerberos: MIT vs. Heimdal vs. OpenAFS (or Arla or Transarc AFS). The consensus is that most common configuration questions and issues have solutions, and that those interested should consult the AFS Wiki, as well as the OpenAFS mailing list archives.

Next, we heard what IBM has been doing with the Stonehenge project. In a nutshell, the Stonehenge project is about putting together a turnkey storage management system that uses AFS as its networked filesystem layer. IBM has been developing the management interfaces and has released a Java API so that others can build management tools for AFS as well.

Alf Wachsmann and Venkata Achanta discussed the Perl API they have been working on under the direction of Norbert Gruner (which utilizes XS). The API now containts vos and vldb interfaces, so volume management programs can be written as well. For those interested in a different Perl API, Phil Moore has released his to CPAN. The primary difference in the two APIs is that Phil's forks off shell calls to the underlying commands, while Norbert's uses XS and saves the overhead of the fork/exec. Phil's API is more complete, however.

Wolfgang Friebel gave an update on the German AFS workshop that took place October 7–10, 2003. Alf Wachsmann and Randy Melen announced an AFS Best Practices workshop, to be hosted by Stanford Linear Accelerator Center on March 24–26, 2004.

Sysadmin Education

Curt Freeland, University of Notre Dame; and John Sechrest, Peak Internet Services

This workshop featured discussions of core topics for system administration education programs, a roundtable presentation of participants' courses, and discussions of future work in the area.

Participants assembled a list of core topics in system administration and discussed how this hypothetical list compared to the actual syllabi various programs offer. A consensus is that a single system administration course is not enough, and that programs need to be more comprehensive. Various issues and strategies for encouraging schools and departments to offer system administration courses were discussed.

Curt Freeland and John Sechrest have assembled a list of universities that offer courses and programs in system administration. While there are many such courses and programs, of particular note are two new programs in Europe:

Netherlands Master in System and Network Engineering (*http://www.os3.nl/*).

Master's degree in Network and System Administration at Oslo University College (*http://www.iu.hio.no/data/msc.html*)

These two are of special note as they lead to Master's degrees and are not simply standalone courses.

Work on the theoretical foundations of system administration is advancing as can be seen in this year's SAGE Achievement awards. Participants discussed some ways to help students join with faculty to do further research in system administration.

Advanced Topics

Moderators: Adam Moskowitz, Menlo Computing; Rob Kolstad, SAGE

Transcribed and summarized by Josh Simon and Rob Kolstad

This meeting included experimental use of IRC as a backchannel for interpersonal communications to keep the interruptions down. It led to a few interestingly surreal conversations and mixed evaluations.

The meeting led off with introductions and then the opening question: What's the most difficult challenge you have right now? Or, What do you wish you had to address challenges? Replies included: Overcoming cultural and political resistance to centralized system administration. Sales is a problem. Some have succeeded (with templates and the like). One participant said: "I can sell it. Only takes a 1–2 hour presentation to sell management . . . which is 50% of the problem. Technical dudes MUST buy in!" Standard builds were advocated.

Linux was said to be a hard sell but used anyway due to its affordability – lots of machines coming in under the radar.

Someone noted that heterogeneous cluster participants seem willing give up some autonomy for functionality and its darker side: "If you drive people to outsource their S.A., you're screwed again."

More draconian measures included the "network citizenship" notion: "We just unplug machines that aren't in conformance with our standards." Another participant disables ports when viruses are discovered.

But "Technical dictatorships don't often work well enough. Standardization is good; innovation is good. There must be collaboration and accommodation. [Sharing] the goals helps."

The next discussion is shown in fairly deep detail in order to convey a sense of the workshop's ebb and flow. It has been dramatically condensed even in this lengthy summary.

Cash flow was one participant's #1 problem. "We don't have good structures for doing things like collaborative administration, charge-backs, funny money (between departments). Industry-wise: Administrative toolsets that we have don't support sysadmins well enough (we end up using sneakernet, telephone, etc.). How do we create for the service industry something like financial instruments in the financial community? We'll want data-feeds between/among our toolsets. Consider carrying around a little micro-charging header on services being rendered (e.g., a virus elimination). Millions of small businesses need this!"

Discussion ensued: "Granularizing these tasks hurts innovation. We are pure overhead."

"We're on the tail end of the stick and get our budgets cut first when things aren't great."

"Of course, being a profit center doesn't help that much – everyone else is just as messed up as we are."

". . . and this leads to bad local optimizations."

"People think they want detailed summaries of IT costs, but then they balk and refuse to buy certain services/products. Bad global impact."

"You must be in a very large company for market forces to work effectively among divisions – otherwise you don't have the proper efficiencies of scale."

"It's good to know costs. Sometimes, though, this perverts the problem solution technique by pushing costs around. Monetary values on various services sometimes thwart corporate missions."

"People buy bandwidth, CPU, disk and want to own it 'forever'. They want to pay *once*. They prefer to think of having a computer, not the use of 100,000 CPU cycles to do an operation."

"From whose point of view does one look at costs [and value]? Customer, VP, Manager, CIO, CEO: different points of view!"

"Don't artificially granulate the cost. I like the all-you-can-eat approach. Tiered plans are fine, but try to avoid artificial situations with costs over which you have no control. Try to cost things so that both sides of the arrangement arrive at mutual efficiency. I wonder if monthly billing is going to increase our customers' perceptions of us."

"We should teach them what we're doing! 'I did a tuneup for you.'"

"Auto repair; you pay book rate, independent of how long it takes. We need to insert (deliberately) a noisy level of suffering-causing failures so people understand what 'good' is."

[General group muttering: It's unethical.]

[Consider a] "popup [that] says 'Network failed . . . we repaired it for you in background'"

"Valuation of services is the main problem. Outsourcing has hidden costs (e.g., cost of data access). 'Flexibility' is never valuated. Agility counts!"

"Must valuate the 'cost' or 'value' of NOT doing something."

"Management at our institution wanted disaster recovery after a disaster, despite our requests for years prior."

Why can't we describe ourselves/our job?

"J. Deming says, 'You can't fix [manage] what you can't measure.'"

"We just got into metrics. We use RUM ('resource utilization metric'): 10% of time doing tickets, 10% training, etc. Management prefers this to '17 minutes to add a user.'"

"I am opposed to bad metrics and bad charges – these are worse than having nothing at all. Metrics disincentize. [For example,] you promote or terminate people based on the number of tickets closed (thus punishing those who can solve difficult problems)."

One group member told this story: "I tried to morph into an MBA; failed. I'm really a consultant. I repeatedly encountered a request for 'a better way to do system administration'. Yet, lots of organizations denied there was a problem. I finally theorized: I think it's *our* fault. The knowledge we bubble up to our management is 'good news' intended to make us look good. 'We're doing fine; everything is under control.' Instead, we need to send more details than the CIO/COO wants to hear. We need face-time with management structure to make them learn enough to understand the real problems in their own infrastructure. IT buttresses all people. We need to make that clear!"

General discussion about actual use and sizes of LDAP scaling.

Challenge: Document management system. Anyone have any good solutions?

Xerox Docushare was mentioned repeatedly. Webdav, twiki, Zope, and DCWorkflow were mentioned.

Challenge: How do we keep things fun (esp. for those with spouses and children)?

Comments included: "movie bucks," general agreement, free coffee, the notion that the job *isn't* fun, the notion that it shouldn't be too much fun, engendering pride, development vs. firefighting, uninterrupted time for projects, SWAT team assignments vs. development projects, project demos, fellowship, recognition, separating work from socialization/other-parts-of-life, involving others in purchases (e.g., peripherals), "development teams" to attack projects in a sprint, and two-way radios.

A short discussion of spam covered its volume and demoralization potential. Email size limits were discussed. Sometimes email is the only way to share large files; this means that a new mechanism is required.

How does one evaluate value? How much is RH10 really worth?

General discussion of integration costs/issues, pricing, etc., continued. One person noted: "There's nothing RH'ish about this question. Senior level sysadmin means understanding vendors pull the rug out at any time and we must proactively deal with this. I don't put certain products in core services. Building too many dependencies on something you're locked into can be bad. Recently Verisign changed their licensing terms to charge us a *lot* since we're an unusual site. Plan for this! We use open source when we can, open standards. We need to be agile."

Complexity was raised as an issue: "We see increasing complexity: volume managers, grids, etc. etc. How do we keep these different level of sysadmins current on these when we have 1,000 machines, many of which change a lot?"

Comments included: the expense of diversity, the impossibility of having "all senior sysadmins," a disagreement about that, and a list of different solutions for NAS, SAN, and other storage management.

One of the group had "a management issue. I have a good fire-fighting sysadmin, short tasks, etc. This person wants to do 'more meaty' things but can't." How to solve this?

Suggestions included: career counseling and a set of discussions about that, "spinning his own job to him," encouraging him to grow, his inability to recognize his own failure, training, the adrenaline and endorphins of firefighting, and a thought that maybe he should be a firefighter/savior kinda of person.

What about lifecycle management for files? We get thousands of new files per day and we need to manage where they reside, where the copies are, etc. Anyone know any software to do this?"

Suggestions included: Permabit and Alien Brain (though that is mostly in the audio space).

One person had an interesting issue: "Availability is declining. I see three management psychoses. First one: Every time availability declines, they increase 'process' to fix the problem. Currently, we have a 90-minute daily change management meeting. They're squeezing. #2: Ownership. They're so afraid about someone dropping a problem, they create process to thwart moving the solution to the best person for the job. Admins work on a per-machine basis, not on subsystems. Ownership is sticky – must stay on phone with people for hours to fix things. #3: We're 'xxx.com', and we do things differently and no one can teach us anything."

Discussion included: hire a consultant (though the problem owner said that that would be impossible), a general throwing-up-of-hands that this problem was unsolvable, the notion that 'fear to fix problems' is also part of the uptime problem in addition to 'procedurizing,' 'philosophy of processes,' be careful of

sensitivity to alarms, demonstration of how process hurts the metric, and admonitions to play by the (presumably defective) rules until it's clear they're bad.

Finally the group made predictions for 11/16/2004. A few of them were particularly interesting:

- Unemployment levels will still be above 5% for the national average [100%]
- Context-aware services (those that are location-dependent) will begin to be deployed (there'll be some in major cities) [14/30]
- Sun's market share will continue to decline [100%]
- Spam will force a sea change (discontinuity) in either government, or business, or both, such as major legislation or some major company doing something really dramatic, or something [27/30]
- There'll be a significant backlash against the RIAA in particular and digital rights management in general, probably from a university or collection of universities, with the potential to completely change the landscape [22/30]
- The SCO thing will still be going on and still nobody will give a $#!+ [28/30]
- You will still not be able to use native IPv6 end to end across the Internet in any useful way [28/30]
- No technical solution will stem the tide of spam on the Internet backbone [100%]
- There still won't be a widespread music CD copy-protection system [100%]
- Most consumer PCs sold will not have a floppy drive and will have writable DVD drive [25/30]
- A Windows-based virus/worm/whatever will cause widespread data loss [26/30]
- SCO will lose the lawsuit [100%]

THE LISA GAME SHOW

Summarized by Josh Simon

This year's quiz show was more exciting than in years past for a few reasons. On Monday, Rob Kolstad's laptop – the ancient piece of crap with a broken screen – was stolen out of a locked room, which was supposedly guarded by security as well. He didn't have the most current version of the code or questions backed up to his home network. (Lesson: Back up your laptop frequently!) So Rob was more invisible than usual this conference, rewriting the game show software, writing new questions, choosing audio songs involving smoke and fire (because of the nearby wildfires and the ashfall the first half of the week), and trying not to go completely insane. In addition to the hardware and software issues, we'd changed the format slightly. We now had four rounds of four contestants (involving 16 people) instead of the three rounds of three (nine people). Consensus after the fact was that it kept Rob from spending time with the contestants and in the banter that's very popular.

Things in the show itself were going okay, modulo a "wrong answer" buzzer effect every time we exited a question to go back to the board, regardless of the correctness (or not) of the answer, until for no apparent reason the software crashed just before the midpoint of game one. Luckily, we'd been keeping a manual transaction log at the judging table so we had it to recover from. The show resumed (after Rob did a code fix in real time with the main monitors off and Dan Klein did an improvisational comedy routine to keep folks entertained) only to have the buzzer system fail spectacularly in the middle of another game. So Dan and Josh went to the backup system of contestants raising their hands. We had a couple of instances where the contestants didn't wait to be acknowleged and so the wrong person answered, but it didn't seem to affect the final scoring much.

The first- and second-place finishers in each round won one of the Linux adapter kits for their Sony PlayStations; the third- and fourth-place contestants in each round won a variety of books from several publishers.

The final round (with the winners from the first four rounds) ended in a tie for first and second place, so we played a tie-breaker catgeory. That caused us to end in a tie for second and third place, so we played another tie-breaker category. When all was said and done, we declared Ken Hornstein the winner, and he walked away with his Linux adapter kit, a satellite photo of the smoke plumes from the San Diego fire (with the Town & Country more or less centered on the map), and an signed (by Dan Klein) photo of the Sunday sun, with the visible sunspots. Final-round winners also received valuable cash prizes in the form of pictures of dead presidents ($25 each for third and fourth place, $50 for second place, and $100 for the grand winner).

SAVE THE DATE!

13th USENIX Security Symposium

August 9–13, 2004 ◆ San Diego, California

The USENIX Security Symposium brings together researchers, practitioners, system administrators, system programmers, and others interested in the latest advances in security of computer systems.

"This is the most important conference I go to."

–Steve Bellovin, *AT&T Fellow, AT&T Labs Research; co-author of* Firewalls and Internet Security: Repelling the Wily Hacker *(Addison-Wesley Professional, 2003)*

http://www.usenix.org/sec04/

NEW FORMAT!

SAVE THE DATE!

2004 USENIX Annual Technical Conference

June 27–July 2, 2004 ◆ Boston, Massachusetts

The new-format Annual Tech '04 will feature:

- 2.5 days of General Sessions—original and innovative papers about modern computing systems
- 2.5 days of FREENIX—a showcase for the latest developments in and interesting applications of free and open source software
- 5 days of content from Special Interest Group Sessions, including UseLinux, Security, and more
- 6 days of training with up to 30 tutorial offerings
- Famous-name Plenary Sessions every day
- Special social events every evening
- Plus BoFs and Guru Is In Sessions

http://www.usenix.org/usenix04/