



Rik is the editor of `;login:`.  
[rik@usenix.org](mailto:rik@usenix.org)

**A**rguing with Mark Burgess is easy. When I read his article in this issue, I found myself agreeing with some parts and disagreeing with others. System administration has changed in many ways, but some things will remain the same.

In his opinion piece, Mark makes a point that he has made before: that sysadmins need to become *business relevant*. Although I understand his point, I also doubt that any sysadmin (who wasn't fired) was ever business irrelevant.

## The Dark Ages

I began working as a sysadmin in 1984, managing single systems for a handful of people. I co-wrote (with Becca Thomas) one of the first books on system administration [1], starting at a point when the people who were actually doing the work at UC Berkeley had never even heard of the term “system administration.” This was the same tack taken by Becca Thomas earlier when she wrote the first wildly successful tech book about UNIX [2]. Becca had looked at the logs created by process accounting to discover which UNIX tools people most commonly used at UC Berkeley and wrote about them. I went to UCB to collect data on what sysadmins were actually doing, but as there was no way to distinguish what ordinary users and sysadmins did via process accounting (besides using `su`), I interviewed sysadmins.

Not so surprisingly, some things really haven't changed. Those sysadmins were managing user accounts, managing printer queues, and dealing with disk issues. One issue that has largely gone away were techniques for preventing disk volumes from filling up. But others, such as backup strategies or migrating users to balance load, are still common today.

To support my research, I consulted as a sysadmin for small companies using UNIX. In those days, most of the work involved AT&T's System V, as you could not run UNIX without a license from AT&T. I also worked for a software development company that would get different UNIX-based workstations for the purpose of porting their software. Many of these computers were BSD-based, which, then as now, meant differences in pathnames, directory hierarchies, and even the formats of configuration files. Thanks to included online documentation (man pages), getting systems set up and ready for work was possible, with some head-scratching. The real oddballs were HP's HPUX, IBM's AIX, and Apollo's DomainIX, in increasing order of weirdness.

At this time, networks consisted of serial lines. I became an expert at installing RS-232 connectors and still have a breakout box (used for debugging RS-232). Offices had serial-connected terminals, and the software company used UUCP to copy files between computers, which wasn't as bad as it might seem. Sure, having a communication speed of 9600 to 19200 baud makes it seem like file transfers would take forever: a one megabyte transfer could take 20 minutes! But files were smaller then, too.

## The Business

The point of this history is not how primitive things were. Things were interesting, but they did work. The real point is that none of this would have been done *unless I was performing some task required by the business that had hired me*. None of it was business irrelevant.

Having written that, I will readily admit that things have changed. Not just that communications are blazingly fast, disks amazingly huge, and computers incredibly faster and cheaper. Rather, we now have so many systems to manage. Instead of managing a couple of office systems, or tens of workstations, some people now manage clusters of thousands, even tens of thousands of servers. So doing what I did in the 1980s just doesn't scale; you cannot sit in front of a terminal and use the command line to configure thousands of servers. You need automation.

Although I had moved on to security by the late 1980s, I really began to see the changes when I researched patch management for NASA in 2002. I had heard that the San Diego Supercomputing Center had centralized patch management, so I flew out there to take a look. There were just four people managing nearly 300 UNIX workstations, and seven people managing 400 Windows desktops and a few servers. The UNIX folks were using Cfengine, my first real exposure to Configuration Management (CM), whereas the Windows sysadmins were using Microsoft's proprietary image management system, SMS [3].

This is where I find myself really agreeing with Mark. If SDSC had opted to build their own CM tools, I doubt that things would have worked out so well. The real difficulties were that not all systems were the same. And it's these differences that will always ensure that there are sysadmins in the loop. Some differences include having to run different patch levels, or even different versions of OS software to support applications, while others have to do with differences in things as diverse as hardware used and management style. And by management, I do mean the bosses.

## The Lineup

Mark Burgess is first up, with his article about the future of sysadmin. You will notice that Mark is wondering whether DevOps is the future, not the relatively narrow focus that I decided to pick on. I've argued privately with Mark before (and recently), but a lot of what I've fired off to him has more to do with his choice of words than the ideas behind them. Mark has both thought and written a lot about this topic, and his article deserves careful reading.

Garduno et al. have produced an article based on their award-winning LISA paper about a means of visually monitoring Hadoop clusters. They convincingly argue that having visual consoles that convert the output of Hadoop logs into different presentations makes understanding where problems lie much simpler. For example, they show how you can tell the difference between node (hardware) issues and problems with a particular MapReduce job.

I interviewed Cory Lueninghoener, a strong advocate for CM, but not about CM, but rather about supercomputers. While one

might think that a Hadoop cluster is a type of supercomputer, supercomputers are really quite different. They use specialized hardware, software, and operating systems, and focus on computation rather than data crunching, what Hadoop clusters are designed to do. Cory provides interesting insights into his world, where CM goes beyond being just a management tool, but also a form of audit.

David Lang presents his research on setting up WiFi networks at the SCALE Linux conference. While most of us will not be doing anything like this, David, who is also a licensed radio operator, explains that setting up multiple access points (APs) *and* having them function efficiently relies on understanding how radios work. David has some simple, down-to-earth tips for how best to configure multiple APs.

Kirk McKusick writes about the changes he has made to the FreeBSD 8 fast filesystem (FFS) and fsck. Kirk was attending FAST 2013, and the very first paper was about changes to the Linux ext3 filesystem and fsck that could greatly speed up filesystem checking. Kirk was inspired, and went off that very afternoon and wrote patches to take advantage of some of the ideas presented.

Springer et al. explain the tool they have created to analyze network traffic, Chimera. We typically use IDS software, like Snort or Bro, for monitoring network traffic. The trouble with these tools is that they are either too simple (Snort) or hard to program (Bro) to analyze patterns of traffic that extend across millions of packets. Chimera helps with this through the use of an SQL-like programming language that outputs optimized code that works with Bro. The authors hope that their tool will make it easier for non-specialists to extract useful information from network logs.

David Blank-Edelman educates us about Perl documentation. I know, sounds boring. But it's not. David explains the cool things you can do using perldoc, as well as adding perldoc annotations to the Perl code you create.

Dave Beazley has decided not to take another deep dive into Python. Dave has been working on the *Python Cookbook* [4] and shares some of his own surprises based on his work revising the book. As always, I sit with the Python prompt waiting, trying out the tricks that Dave exposes.

Dave Josephsen finishes his three part series on Nagios XI, a commercial Nagios product that answers most of the criticisms made by some people against Nagios. Dave begins with his usual rant, this time against "religious battles," before pointing out just how XI solves the objections many people have to using Nagios.

Robert Ferrell muses about cause-and-effect, as in, why aren't more people aware of the harm that ignoring this causes? Although texting while driving can produce spectacular results (and I don't mean increases in Twitter followers), Robert also

takes a hard look at how many people respond to having someone point out security vulnerabilities to them. We are, after all, a society that discretely points out unzipped zippers, so why men get their panties in a bunch when unzipped Internet applications get pointed out is beyond both Robert and me.

Elizabeth Zwicky has written wonderful reviews of five books for this issue. She begins with one of her favorite topics, learning statistics, then covers several books on programming, and a new Donald Norman book. Mark Lamourine covers a Maker book about the Raspberry Pi. The more I learn about the Pi, the more interesting it sounds. I've also written a review of a book about learning vim, the replacement for vi.

We finish up this issue with summaries from LISA 2012. USENIX has started posting summaries to the *login*: portion of the Web site as soon as they are ready, instead of waiting until the next print issue of *login*: comes out. This can mean that summaries will appear weeks before they would have back when we waited for the issues to appear in the mail, or show up on the Web site [5].

The pace of software development has sped up enormously, with the big data companies, like Google and Facebook, pushing out new versions of their software every couple of weeks. These updates would not be possible without the existence of something that gets called DevOps, and it is totally unlike the world of sysadmin that I researched back in the mid-'80s.

But not every company or organization is like Google or Facebook. Not everyone manages to be an amazing programmer or infrastructure engineer (to borrow a term from Mark). For many businesses, a slower, steadier pace works well enough. And that's a good thing, as the supply of geniuses is pretty limited. While it is important to follow best practices, it is just as important to realize the limitations of the actual workforce, and not expect that everyone will be a heroic programmer or sysadmin.

*Editor's Note:* With the April 2013 issue, we have changed to a two column layout for articles and columns. Two columns has been the standard layout for conference reports for many years now. This new layout gets rid of most of the white space present in the wide margins, and leaves us with room for more articles and columns without increasing page count, that is, using more paper.

The April issue is also the one I have the hardest time filling every year. Print magazines have long lead times, and the time for writing April articles falls during December and early January. Few people are willing to write over the Christmas holidays, and once January begins, they are busy catching up after a week (or more) spent dealing with the holidays.

I also underestimated the effect of the layout changes, as my goal is to produce a great magazine everytime, while not going beyond a set page length. I plan on taking advantage of the additional space in future issues.—*Rik*

### References

- [1] R. Thomas, R. Farrow, *Unix Administration Guide for System V* (Prentice-Hall, 1989): <http://www.amazon.com/Unix-Administration-Guide-System-V/dp/0139428895>.
- [2] R. Thomas, *A User Guide to the Unix System*, 2nd edition (McGraw-Hill Osborne Media, 1984).
- [3] SMS, now, System Center Configuration Manager: [http://en.wikipedia.org/wiki/System\\_Center\\_Configuration\\_Manager](http://en.wikipedia.org/wiki/System_Center_Configuration_Manager).
- [4] A. Martelli, A.M. Ravenscroft, D. Ascher, *The Python Cookbook* (O'Reilly Media, 2005): <http://www.amazon.com/Python-Cookbook-Alex-Martelli/dp/0596007973>.
- [5] *login*.: <https://www.usenix.org/publications/login>.

# SAVE THE DATE!



## LISA'13

27th Large Installation  
System Administration Conference

Sponsored by USENIX  
in cooperation with LOPSA

NOVEMBER 3-8, 2013 | WASHINGTON, D.C.

[www.usenix.org/lisa13](http://www.usenix.org/lisa13)

**Come to LISA '13 for training and face time with experts in the sysadmin community.**

### LISA '13 will feature:

#### 6 days of training on topics including:

- Configuration management
- Cloud Computing
- Distributed Systems
- DevOps
- Security
- Virtualization
- And More!

#### Plus a 3-day technical program:

- Invited Talks
- Guru Is In sessions
- Paper presentations
- Vendor Exhibition
- Practice and Experience reports
- Workshops
- Posters and Lightning Talks

**New for 2013! The LISA Labs "hack space" will be available for mini-presentations, experimentation, tutoring, and mentoring.**

**Check out the Call for Participation! Submissions are due Tuesday, April 30, 2013.**

[www.usenix.org/lisa13/cfp](http://www.usenix.org/lisa13/cfp)

**Stay Connected...**



[twitter.com/usenix](https://twitter.com/usenix)



[www.usenix.org/youtube](http://www.usenix.org/youtube)



[www.usenix.org/gplus](http://www.usenix.org/gplus)



[www.usenix.org/facebook](https://www.usenix.org/facebook)



[www.usenix.org/linkedin](http://www.usenix.org/linkedin)



[www.usenix.org/blog](http://www.usenix.org/blog)