

LINUX JOURNAL™

Put Your Servers in
the Cloud with **Amazon
EC2 and Ubuntu**

Build an
Installation Toolkit

Since 1994: The Original Magazine of the Linux Community

MARCH 2010 | ISSUE 191 | www.linuxjournal.com

System Administration

► Security
Information
Management
with **AlienVault**

► Large-Scale
Sysadmin Tips

► Implement a
Nagios-to-SMS
Alerting Service

PLUS:

POINT/COUNTERPOINT:
/opt vs. /usr/local



REVIEWED:
Axigen Mail Server

\$5.99US \$5.99CAN

03>

0 0928103102 4

03>

MPLS for the masses

\$39⁹⁵



Usually MPLS routers cost more than \$1000, but not anymore. MikroTik gives you the ability to use MPLS in any network. No more big box prices for MPLS! A chicken in every pot!

MPLS stands for Multi Protocol Label Switching. It can be used to replace IP routing - packet forwarding decision is no longer based on fields in IP header and routing table, but on labels that are attached to the packet.

MPLS makes it easy to create “virtual links” between nodes on the network, regardless of the protocol of their encapsulated data. It is a highly scalable, protocol agnostic, data-carrying mechanism. MPLS allows one to create end-to-end circuits across any type of transport medium, using any protocol.

Features:

- Label Distribution Protocol for IPv4
- Virtual Private Lan Service
 - * VPLS LDP signaling
 - * VPLS MP-BGP based autodiscovery and signaling
 - * split-horizon bridging
- RSVP TE Tunnels
 - * explicit paths
 - * CSPF path selection
 - * OSPF extensions for TE tunnels
- Virtual Routing and Forwarding
- MP-BGP based MPLS IP VPN
- OSPF and RIP as CE-PE protocols

Benefits:

- higher speed forwarding in network core
- ability to implement transparent L2 and L3 VPNs (VPLS & VRF)
- reduced VPN overhead compared to legacy tunneling solutions
- traffic engineering to implement QoS and optimize network usage
- ability for the ISP to create VPNs without user interaction
- separate tunnels for voice, video, or data

All MikroTik RouterBOARDS support MPLS, including the RB750 which costs \$39.95. The RB750 is a SOHO router with a 400MHz Atheros CPU, five ethernet ports, plastic case and PSU. With MPLS, RB750 is capable of wire speed throughput for 1000byte packets and up, maximum 80000 pps with smaller packets.

NEW!

Easy to configure. Always adjustable.

FLEXIBLE SERVER



1&1® DYNAMIC CLOUD SERVER

A virtual server environment with full root access – adjust the processor core, RAM, and/or hard disk space at any time. Prices will be reflected accordingly.

Basic Configuration:

- **1 AMD Opteron™ 2352 Core Processor**
(up to 4 cores available)
- **1 GB RAM**
(up to 15 GB RAM available)
- **100 GB Hard Disk Space**
(up to 800 GB available)

All Configurations Include:

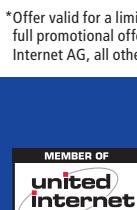
- 2000 GB Traffic
- Full Root Access
- Windows Server 2008 R2 Standard
Available as an add-on, additional fees apply.
- Parallels Plesk Panel 9
- 24/7 Toll-Free Support

**Special Offer:
3 Months
FREE!***

Basic Configuration:

~~\$49.99~~

More special offers are available online.
For details, visit www.1and1.com



Call 1-877-GO-1AND1
Visit us now www.1and1.com

1&1

®

*Offer valid for a limited time only. 3 months free offer applies to basic configuration only. 12 month minimum contract term and set up fee apply. Visit website for full promotional offer details. Program and pricing specifications and availability subject to change without notice. 1&1 and the 1&1 logo are trademarks of 1&1 Internet AG, all other trademarks are the property of their respective owners. © 2010 1&1 Internet, Inc. All rights reserved.

CONTENTS

MARCH 2010
Issue 191



FEATURES

SYSTEM ADMINISTRATION

50
**TAMING
THE BEAST**

An introduction to large-scale system administration.

Jason Allen

54
**ALIENVAULT:
THE FUTURE
OF SECURITY
INFORMATION
MANAGEMENT**

OSSIM: giving security management a brain.

Jeremiah Bowling

60
**USE SSH
TO CROSS A
SUSPECT HOST
SECURELY**

Don't fear the Valley of the Shadow of Death. SSH through it.

der.hans

66
**USING AN
SMS SERVER
TO PROVIDE
A ROBUST
ALERTING
SERVICE FOR
NAGIOS**

Texting: not just for teenagers.

Eric Pearce

ON THE COVER

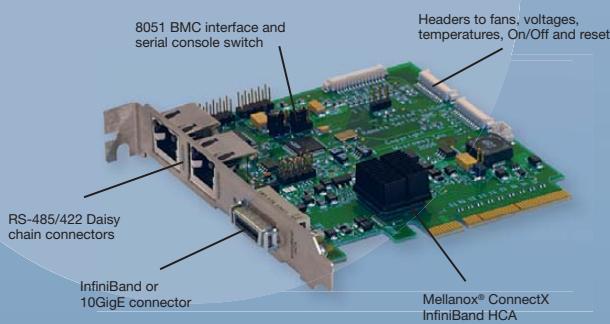
- PUT YOUR SERVERS IN THE CLOUD WITH AMAZON EC2 AND UBUNTU, P. 72
- BUILD AN INSTALLATION TOOLKIT, P. 34
- SECURITY INFORMATION MANAGEMENT WITH ALIENVAULT, P. 54
- LARGE-SCALE SYSADMIN TIPS, P. 50
- IMPLEMENT A NAGIOS-TO-SMS ALERTING SERVICE, P. 66
- POINT/COUNTERPOINT: /OPT VS. /USR/LOCAL, P. 76
- REVIEWED: AXIGEN MAIL SERVER, P. 46

Cover Photo by Jason Allen with permission from Fermilab. This photo was taken at Fermi National Accelerator Laboratory, which is managed by Fermi Research Alliance, LLC under Management and Operating Contract (DE-AC02-07CH11359) with the Department of Energy.

Your Applications Will Run Faster With Next Generation Microway Solutions!

TriCom™ X

- QDR/DDR InfiniBand HCA
- ConnectX™ Technology
- 1µsec Latency
- Switchless Serial Console
- NodeWatch™ Remote Management



Teraflop GPU Computing

For Workstations and HPC Clusters

- NVIDIA® Tesla™ GPU with 240 Cores on One Chip
 - CUDA™ SDK
- NVIDIA® Quadro® Professional Graphics
- AMD® FireStream™ GPU
 - Stream SDK with Brook+



NumberSmasher®

Large Memory Scalable SMP Server



- Scales to 1 TB of Virtual Shared Memory
- Up to 128 CPU Cores
- 8U System Includes 32 Quad Core CPUs
- QDR 1 µsec Backplane



FasTree™ X

- Mellanox® InfiniScale™ IV Technology
- QDR/DDR InfiniBand Switches
- Modular Design
- 4 GB/sec Bandwidth per Port
- QSFP Interconnects
- InfiniScope™ Real Time Diagnostics

Call the HPC Experts at Microway to Design Your Next High-Reliability Linux Cluster or InfiniBand Fabric.

508-746-7341

Sign up for Microway's Newsletter at
www.microway.com

Microway
Technology you can count on™

CONTENTS MARCH 2010

Issue 191

COLUMNS

18 REUVEN M. LERNER'S AT THE FORGE
Testing JavaScript

22 DAVE TAYLOR'S WORK THE SHELL
Still Parsing the Twitter Stream

24 MICK BAUER'S PARANOID PENGUIN
Linux VPNs with OpenVPN, Part II

30 KYLE RANKIN'S HACK AND /
Linux Troubleshooting, Part I: High Load

34 DIRK ELMENDORF'S ECONOMY SIZE GEEK
Installation Toolkit

76 KYLE RANKIN AND BILL CHILDERS' POINT/COUNTERPOINT
/opt vs. /usr/local

80 DOC SEARLS' EOF
A Cloud of One's Own

REVIEW

46 AXIGEN MAIL SERVER
Mitch Frazier

INDEPTH

72 RUNNING UBUNTU 9.10 UNDER AMAZON'S ELASTIC CLOUD
No servers? No problem, with servers in the cloud!

Bill Childers

IN EVERY ISSUE

- 8** CURRENT_ISSUE.TAR.GZ
- 10** LETTERS
- 14** UPFRONT
- 40** NEW PRODUCTS
- 42** NEW PROJECTS
- 65** ADVERTISERS INDEX
- 78** MARKETPLACE

The screenshot shows the Axigen WebAdmin interface with the 'Clustering' section selected. It includes tabs for 'LDAP Connectors', 'User Maps', and 'Routing and Authentication'. A central panel displays 'Existing LDAP connectors:' and a button to '+ Add Connector'. Below this is a 'Global Settings' section with a 'Log Level' slider set to 'Critical messages'. Other settings include 'Log' (set to 'Use internal log'), 'Thread Management' (set to 'Limit no. of threads to 5'), and a note about log file location. A 'Save Configuration' button is at the bottom.

46 AXIGEN MAIL SERVER

The screenshot shows the AlienVault interface with various monitoring dashboards. One dashboard shows a pie chart of network traffic and a bar chart of top 10 hosts. Another dashboard shows a timeline of events and incident status. The left sidebar lists modules like Network, Hosts, Events, Reports, and Configuration.

54 ALIENVAULT



66 NAGIOS-TO-SMS

Next Month: SOFTWARE DEVELOPMENT

Next month, we look into the root cause of all bugs: Software Development. So if you've got bugs in your Web application, don't miss our article on the Selenium testing tool. If you're still in the development stages of your Web app, read our article on jsormdb, an embedded JavaScript database, and see if it can simplify your app. And, if your Web app is on the server side (and you need something slim), read our article on Mongoose, an embeddable Web server in C.

After you've digested all of that, don't miss our review of the Android Linux-based Motorola DROID. Read the review then go get one, but don't call us, we'll call you.

USPS LINUX JOURNAL (ISSN 1075-3583) (USPS 12854) is published monthly by Belltown Media, Inc., 2211 Norfolk, Ste 514, Houston, TX 77098 USA. Periodicals postage paid at Houston, Texas and at additional mailing offices. Cover price is \$5.99 US. Subscription rate is \$29.50/year in the United States, \$39.50 in Canada and Mexico, \$69.50 elsewhere. POSTMASTER: Please send address changes to *Linux Journal*, PO Box 16476, North Hollywood, CA 91615. Subscriptions start with the next issue. Canada Post: Publications Mail Agreement #41549519. Canada Returns to be sent to Bleuchip International, P.O. Box 25542, London, ON N6C 6B2



ABERDEEN
S E R V E R S A N D S T O R A G E

WHAT'S THE DEAL WITH THESE GUYS?

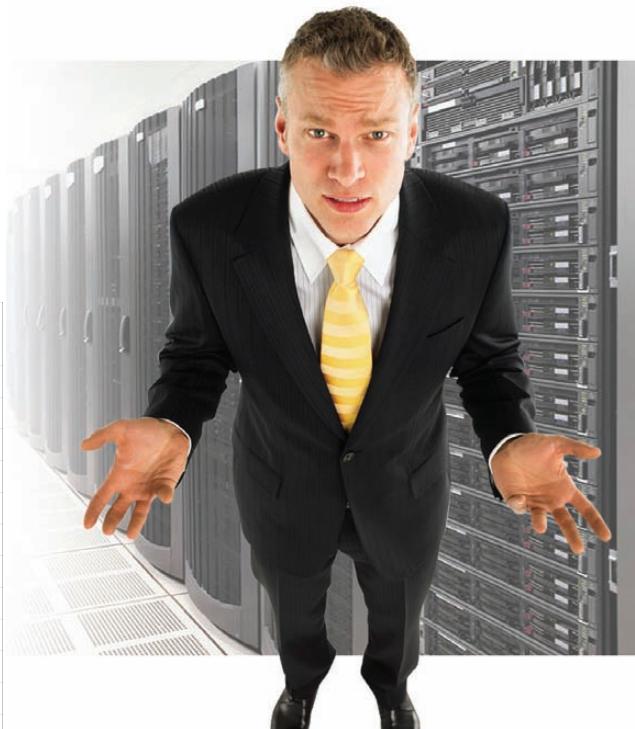
Sometimes you have to ask, "What are they thinking?"

Companies need to increase ROI without being taken to the cleaners by manufacturers selling servers featuring entry-level benefits with enterprise-level pricing.

Aberdeen gets it. Businesses are in desperate need of Network Attached Storage servers that simply deliver the best bang for the buck.

Look at these features and benefits:

	Dell PowerVault	HP StorageWorks	Aberdeen AberNAS
Hot-Swap Disk Drives	✓	✓	✓
Hardware RAID	✓	✓	✓
Dual Port Gigabit Ethernet	✓	✓	✓
Built-in Replication	✓	✓	✓
Microsoft® WSS 2008 Models	✓	✓	✓
iSCSI Target	✗	✓	✓
Linux Storage System Models	✗	✓	✓
System Recovery Disk	✗	✓	✓
DAS Storage Expansion	✗	✓	✓
VMware® Ready Certified	✗	✗	✓
Independent OS Drive	✗	✗	✓
Out of Band RAID Management	✗	✗	✓
Available w/ 2TB Drives	✗	✗	✓
Warranty	3 Years	3 Years	5 Years



Who gives you the best bang for the buck?

	Dell PowerVault NX300	HP StorageWorks X1400	Aberdeen AberNAS 163
Intel® Xeon® Processor	E5504 2GHz	E5504 2GHz	E5504 2GHz
Memory	3GB	2GB	3GB
Drive Interface	SATA	SATA	SATA
Installed Capacity	2TB	2TB	2TB
Rails	Included	Included	Included
Windows Storage Server 2008	\$3,419	\$4,635	\$2,995
Linux Storage System	Not Available	Not Available	\$2,995



Prices for the above specific configurations obtained from the respective websites on Oct. 12, 2009. Intel, Intel Logo, Intel Inside, Intel Inside Logo, Pentium, Xeon, and Xeon Inside are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries. VMware is a registered trademark or trademark of VMware, Inc. in the United States and/or other jurisdictions. For terms and conditions, please see www.aberdeencorp.com/abpolo/abterms.htm. Ij032

888-297-7409
www.aberdeencorp.com/ij032

LINUX JOURNAL™

Since 1994: The Original Magazine of the Linux Community

DIGITAL EDITION NOW AVAILABLE!

Read it first

Get the latest issue before it
hits the newsstand

Keyword searchable

Find a topic or name
in seconds

Paperless archives

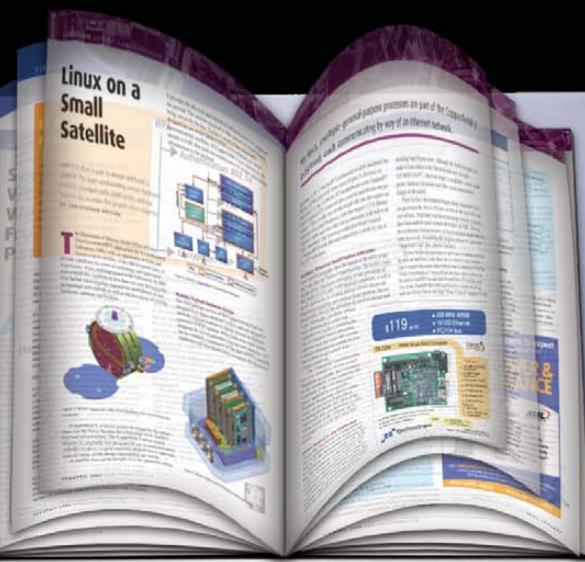
Download to your computer for
convenient offline reading

Same great magazine

Read each issue in
high-quality PDF

Try a Sample Issue!

www.linuxjournal.com/DLISSUE



LINUX JOURNAL

Executive Editor

Jill Franklin

jill@linuxjournal.com

Senior Editor

Doc Searls

doc@linuxjournal.com

Associate Editor

Shawn Powers

shawn@linuxjournal.com

Associate Editor

Mitch Frazier

mitch@linuxjournal.com

Art Director

Garrison Antikajian

garrison@linuxjournal.com

Products Editor

James Gray

newproducts@linuxjournal.com

News Editor

Justin Ryan

news@linuxjournal.com

Editor Emeritus

Don Marti

dmarti@linuxjournal.com

Technical Editor

Michael Baxter

mab@cruzio.com

Senior Columnist

Reuven Lerner

reuven@lerner.co.il

Security Editor

Mick Bauer

mick@visi.com

Hack Editor

Kyle Rankin

lj@greenfly.net

Virtual Editor

Bill Childers

bill.childers@linuxjournal.com

Contributing Editors

David A. Bandel • Ibrahim Haddad • Robert Love • Zack Brown • Dave Phillips • Marco Fioretti
Ludovic Marcotte • Paul Barry • Paul McKenney • Dave Taylor • Dirk Elmendorf

Proofreader

Geri Gale

Publisher

Carlie Fairchild

publisher@linuxjournal.com

General Manager

Rebecca Cassity

rebecca@linuxjournal.com

Sales Manager

Joseph Krack

joseph@linuxjournal.com

Associate Publisher

Mark Irgang

mark@linuxjournal.com

Webmistress

Katherine Druckman

webmistress@linuxjournal.com

Accountant

Candy Beauchamp

acct@linuxjournal.com

Linux Journal is published by, and is a registered trade name of, Belltown Media, Inc.
PO Box 980985, Houston, TX 77098 USA

Editorial Advisory Panel

Brad Abram Balilo • Nick Baronian • Hari Boukis • Steve Case
Kalyana Krishna Chadalavada • Brian Conner • Caleb S. Cullen • Keir Davis
Michael Eager • Nick Falts • Dennis Franklin Frey • Alicia Gibb
Victor Gregorio • Philip Jacob • Jay Kruizenga • David A. Lane
Steve Marquez • Dave McAllister • Carson McDonald • Craig Oda
Jeffrey D. Parent • Charnell Pugsley • Thomas Quinalan • Mike Roberts
Kristin Shoemaker • Chris D. Stark • Patrick Swartz • James Walker

Advertising

E-MAIL: ads@linuxjournal.com

URL: www.linuxjournal.com/advertising

PHONE: +1 713-344-1956 ext. 2

Subscriptions

E-MAIL: subs@linuxjournal.com

URL: www.linuxjournal.com/subscribe

PHONE: +1 818-487-2089

FAX: +1 818-487-4550

TOLL-FREE: 1-888-66-LINUX

MAIL: PO Box 16476, North Hollywood, CA 91615-9911 USA

Please allow 4-6 weeks for processing address changes and orders

PRINTED IN USA

LINUX is a registered trademark of Linus Torvalds.





LinuxFest Northwest

**April 24-25 2010
Bellingham, WA**

Grassroots linux gathering ↗
Exhibits of all flavors ↗
Presentations of all levels ↗
Prizes, after parties ↗
FREE admission & parking ↗
FREE open source software ↗
Bring the whole family! ↗

linuxfestnorthwest.org



SHAWN POWERS

If It Works, Don't Fix It

In fact, even if it's broken, please don't fix it. It's probably not actually broken; most likely, you've forgotten to plug it in, or turn it on. Or, perhaps you've forgotten your password. Or, you put your peanut butter sandwich into the manual feed tray.

Although those things might sound far-fetched, they honestly do describe some of the things a system administrator faces on a daily basis—at least, those system administrators who deal with tech support regularly. Apart from PEBKAC errors, as sysadmins, we do have duties that require us to accomplish more things than a human being can possibly accomplish—that's where Linux comes in. Not only do Linux servers (in my experience) require less maintenance, but the tools available for Linux administrators are amazing, and the community is wonderful.

Before you solve a computer problem, it's important to find what is actually going wrong. Kyle Rankin gives us the first part of a series on troubleshooting. This month, he helps us figure out abnormally high server loads. Perhaps I shouldn't run Seti-at-Home on Kyle's servers anymore; he's bound to figure out what process is eating up his cycles! As strange as it sounds, occasionally it's simpler to re-install than it is to troubleshoot. Dirk Elmendorf shows us a wide variety of tools to make installation painless and possibly even fun. Speaking of fun, anyone that has been subject to configuring Sendmail over the years certainly will appreciate Mitch Frazier's review of Axigen, an e-mail server with a GUI interface by default.

Nothing, however, makes the life of a system administrator easier than good planning. Jason Allen describes multiple aspects of planning a successful server infrastructure. In many ways, I wish he'd have sent me the article years ago, but even if your server farm is well established, his tips can help turn a nightmare into a dream job, or at least make the nightmare a little less scary.

Once your servers and workstations are set up, security is extremely important. Contrary to many user's opinions, security is not in place to hinder a user's abilities, but rather to protect the user from harm. Yes, sometimes that means protecting users from themselves, but it also means monitoring for strange activity and keeping a consistent interface. Jeremiah Bowling demonstrates one tool that helps sysadmins with that task. AlienVault is a security information management system that provides a

common interface for several aspects of security management. If you manage computers, you manage security. You'll likely want to check it out.

Another downside of being in charge of system administration is that computers generally work 24/7. That means we have to be available at any time, and from anywhere. Eric Pearce understands that need and shows off how he gets Nagios to alert him via SMS messaging. Unfortunately, being alerted is only half the solution. If you're in an unfamiliar network, or even a network you know is unsafe, der.hans' approach to SSH tunneling can get you back to your network safely. Sadly, we don't have an article on how to explain to your date why you need to set your laptop up in the middle of dinner. You're on your own for that one.

Now that cloud computing is all the rage, it's possible your "server room" doesn't even exist anywhere other than in some mystical on-line space. Bill Childers demonstrates using Ubuntu 9.10 in Amazon's Elastic Cloud. Feel free to argue amongst yourselves whether cloud computing is the future of server infrastructure or just an annoying fad. But, if you really want some fun arguing, Bill is the one to listen to. No, not about cloud computing, but rather about /opt versus /usr/local. Kyle and Bill, as usual, have drastically differing views on the topic in the Point/Counterpoint column. I'm not sure who I agree with this month. You can decide on your own.

What if you're not a system administrator? Well, I'd argue that you do need to be at least the sysadmin of your own computer. But, fear not, we have lots of other goodies for you this month. Reuven M. Lerner talks about JavaScript, Dave Taylor continues his scripting lessons using Twitter as his test subject, and Mick Bauer continues his series on Linux VPNs. When you add news, reviews, product announcements, tech tips and other geeky info, this issue will benefit even the least system administrate-y readers. So, shove a peanut butter sandwich in your manual feed tray, and while you're waiting for tech support to come down and fix your printer, enjoy this issue of *Linux Journal*. I know I did. ■

Shawn Powers is the Associate Editor for *Linux Journal*. He's also the Gadget Guy for LinuxJournal.com, and he has an interesting collection of vintage Garfield coffee mugs. Don't let his silly hairdo fool you, he's a pretty ordinary guy and can be reached via e-mail at shawn@linuxjournal.com. Or, swing by the #linuxjournal IRC channel on Freenode.net.

YOUR HIGH PERFORMANCE COMPUTING HAS ARRIVED.

The ServersDirect® Systems with the Intel® Xeon® Processor helps you simplify computing operations, accelerate performance and accomplish more in less time

STARTING AT **\$899****ENTRY LEVEL INTELLIGENT SERVER**

SDR-S1341-T00 is among our most cost-effective 1U Xeon Servers, and it is ideal for large high-performance computing deployments

STARTING AT **\$959****APPLICATION SERVER**

Refresh your servers with new **SDR-S1337-T02** powered by Intel® Xeon® processor 5500 series, based on intelligent performance, automated energy efficiency and flexible virtualization.

**SDR-S1343-T04**
STARTING AT **\$1,099****1U INTEL® XEON® PROCESSORS 5500 SERIES SERVER W/ 4X 3.5" HOT-SWAP SATA DRIVE BAYS**

- Supermicro 1U Rackmount Server with 560W Power Supply
- Supermicro Server Board w/Intel® 5520 Chipset
- Support up to Dual Intel® 5500 series Xeon® Quad/Dual-Core, with QPI up to 6.4 GT/s
- Support up to 96GB DDR3 1333/ 1066/ 800MHz ECC Reg.DIMM
- 4x 3.5" Hot-swap SATA Drive Bays
- Intel® 82576 Dual-Port Gigabit Ethernet Controller

**SDR-S2311-T08**
STARTING AT **\$1,159****2U INTEL® XEON® PROCESSORS 5500 SERIES SERVER W/ 8X 3.5" HOT-SWAP SAS/SATA BAYS**

- Supermicro 2U Rackmount Server with 560W Power Supply
- Supermicro Server Board w/Intel® 5500 Chipset
- Support up to Dual Intel® 5500 series Xeon® Quad/Dual-Core, with QPI up to 6.4 GT/s
- Support up to 24GB DDR3 1333/ 1066/ 800MHz ECC Reg.DIMM
- 8x 3.5" Hot-swap SATA Drive Bays
- Dual Intel® 82574L Gigabit Ethernet Controller

**SDP-IP308-T10**
STARTING AT **\$1,599****PEDESTAL INTEL® XEON® PROCESSORS 5500 SERIES SERVER W/ 10X HOT-SWAP (OPT.) SATA BAYS**

- Intel Pedestal Chassis w/ 750W (1+1) Power Supply
- Supermicro Server Board w/Intel® 5520 Chipsets
- Support up to Dual Intel® 5500 series Xeon® Quad/Dual-Core, with QPI up to 6.4 GT/s
- Support up to 96GB DDR3 1333/ 1066/ 800MHz ECC Reg./unbuffered DIMM
- Option 10x 3.5" Hot-swap SATA Bays
- Intel® 8257EB Dual-port Gigabit Ethernet Controller

**SDR-S4313-T24**
STARTING AT **\$1,899****4U INTEL® XEON® PROCESSORS 5500 SERIES SERVER W/ 24X 3.5" HOT-SWAP SAS/SATA BAYS**

- Supermicro 4U Rackmount 900W (1+1) Red. Power Supply
- Supermicro Server Board w/ Dual Intel® 5520 Chipsets
- Support up to Dual Intel® 5500 series Xeon® Quad/Dual-Core, with QPI up to 6.4 GT/s
- Support up to 144GB DDR3 1333/ 1066/ 800MHz ECC Reg. DIMM
- 24x 3.5" Hot-swap SATA Drive Bay
- Intel® 82576 Dual-port Gigabit Ethernet Controller

**SDR-S3305-T16**
STARTING AT **\$1,979****3U INTEL® XEON® PROCESSORS 5500 SERIES SERVER W/ 16X 3.5" HOT-SWAP SAS/SATA BAYS**

- 3U Rackmount Server with 1+1 900W Red. Power Supply
- Supermicro Server Board w/ Dual Intel® 5520 Chipsets
- Support up to Dual Intel® 5500 series Xeon® Quad/Dual-Core, with QPI up to 6.4 GT/s
- Support up to 96GB DDR3 1333/ 1066/ 800MHz ECC Reg.DIMM
- 16x Hot-swap SAS/SATA Drive Bays
- Intel® Dual 82576 Dual-Port Gigabit Ethernet (4 ports)

**SDR-C9303-T50**
STARTING AT **\$4,339****9U INTEL® XEON® PROCESSORS 5500 NEHALEM SERIES SERVER W/ 50X HOT-SWAP SATA II / SAS BAYS**

- 9U Chassis with 1620W Redundant Power Supply
- Supermicro Server Board w/ Dual Intel® 5520 Chipsets
- Support up to Dual Intel® 5500 series Xeon® Quad/Dual-Core, with QPI up to 6.4 GT/s
- Support up to 144GB DDR3 1333/ 1066/ 800MHz ECC Reg. DIMM
- 50 x 3.5" Internal SATA Drives Trays
- Intel® 82576 Dual-port Gigabit Ethernet Controller

**SERVERS DIRECT CAN HELP YOU CONFIGURE YOUR NEXT HIGH PERFORMANCE SERVER SYSTEM - CALL US TODAY!**

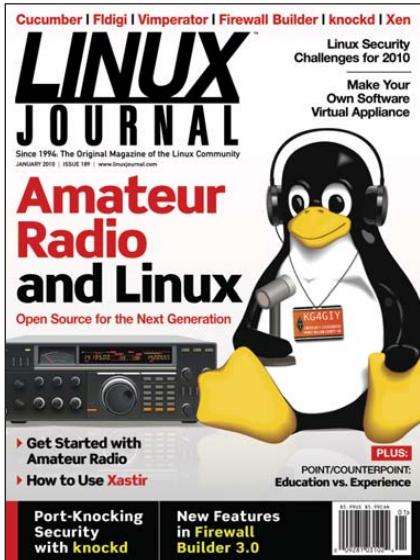
Our flexible on-line products configurator allows you to source a custom solution, or call and our product experts are standing by to help you to assemble systems that require a little extra. Servers Direct - your direct source for scalable, cost effective solutions.

1.877.727.7886 / www.ServersDirect.com

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, Pentium, and Pentium III Xeon are trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

**Powerful.
Intelligent.**

letters



Correction

Regarding a quotation in "They Said It" in the January 2010 issue of *LJ*: "We act as though comfort and luxury were the chief requirements of life, when all that we need to make us happy is something to be enthusiastic about" is not by Einstein, but rather by Charles Kingsley. I'd give you a link, but it's easy enough to Google it yourself.

--
Jason Gade

He's correct, that should have been Charles Kingsley.—Ed.

Marconi?

I noticed on page 4 of the December 2009 issue that next month's issue will focus on Amateur Radio. The accompanying text reads "...Marconi's ushered in the era of radio communications...". I hope a reputable publication such as yours is not going to perpetuate the myth that Marconi invented radio. There's no doubt he was brilliant at self-promotion, but he did not invent radio. Many people contributed to the development of radio technology, and Marconi was one of them. But if you insist on giving recognition to only one person, it should be Nikola Tesla. The US Supreme Court ruled in 1943 against Marconi and in favor of Tesla, deciding that Tesla's patent (645,576) had

priority. Please do some fact-checking before perpetuating a ridiculous myth. Here's a few links to start with:
en.wikipedia.org/wiki/Invention_of_radio,
en.wikipedia.org/wiki/History_of_radio,
en.wikipedia.org/wiki/Nikola_Tesla and
en.wikipedia.org/wiki/Guglielmo_Marconi.

--
Jeff Harp

Bose?

In "When All Else Fails—Amateur Radio, the Original Open-Source Project" in the January 2010 issue, guest editor David A. Lane, KG4GIY, wrongly mentioned that Marconi invented radio. In fact, Sir J. C. Bose, the Indian scientist, was the true inventor of radio. He pioneered the iron filling coherer and lead galena crystal semiconductor receiver. Sir Bose invented the horn antenna and studied properties of refraction, diffraction and dispersion of millimeter and sub-millimeter waves. The above facts were published in the *Proceedings of the IEEE* a few years back. I am an Advanced Grade licensed Indian Radio Amateur and life member of the Amateur Radio Society of India for nearly three decades.

--
Ananda Bose, VU2AMB

David A. Lane replies: Ask 100 people who invented the radio, and of those who bother to answer, you will likely get one answer, and neither Tesla nor Bose will be it. Perhaps the paragraph should have read "...almost since Marconi popularized the thing...". The truth is that history misremembers all the time, and the true geniuses are forgotten by those who come up with the reference implementations. Clearly, both Tesla and Bose contributed to the science that has led us to where we are today, just as much as Marconi and Motorola.

Webinars

I continually receive invitations to various Webinars on many topics and issues. Some don't suit me but many do. Here's the problem: my work schedule often prevents me from tuning in or participating at the time the Webinar is presented. I'd like to find a way to "record" these

Webinars with all the video and audio, so that in the evening when back at home or on the weekend, I can sort through and watch the most pertinent of these events. Is this possible using Linux? Please advise!

--
Scott S. Jones

I feel your pain. There's a couple problems working against us with Webinars. One is that the format is far from standard. The other is that for some reason, many are specifically designed to be done only in real time. (I suspect those are specifically when demonstrations are happening in real time.) I don't think there's anything as Linux users we can do differently, apart from using a screen capture package at the onset of a Webinar. Ideally, they would be archived afterward, so we could watch at our leisure. We learned that ourselves here at Linux Journal. We did a live show (Linux Journal Live) that got many more views after the fact. Our schedules are just too full!—Ed.

Dark Days?

Every month, I read the Letters section, and more often than not, there is an end user who knocks Linux as being for "computer specialists" and not ready for the mainstream. I find this attitude unfortunate. With every release, distributions get better and better, and don't suffer from the constant barrage of malware, viruses and the like we see on Windows. In the December 2009 issue, a reader commented that "It's back to the dark days of MS-DOS all over again." Dark days? I seem to remember back in the mid-1980s, DOS actually worked quite well. What gets me is that Microsoft continues to produce OSes and products that have numerous bugs and security flaws. Users are forced to license virus and malware scanners that are resource hogs and are doing a job that should have been prevented in the first place. Maybe we should all send a bill for our lost time in re-installing, scanning, cleaning and so on to Microsoft—they seem to have no issues collecting license fees. What about the hundreds of hours my staff has lost over the years?

--
George

Heck, I'd settle for just the hours spent trying to find drivers for hardware under Windows! I agree, it is sad people still see Linux as a difficult, cryptic, command-line-only operating system. It's still largely a matter of unfamiliarity. Many folks new to Linux are still overwhelmed by its nuances. The same can be said for people switching from Windows to OS X though, so I don't think it's really a Linux problem. I think we just need to keep pushing Linux in places it makes sense. As usual, my suggestion would be to start in schools!—Ed.

Waiting for Godot

Regarding Mitch Frazier's "Non-Linux FOSS" (December 2009): Explore2fs has been around for more than a decade. It works as designed, but development is slow. The write function has remained on the to-do list for most of the decade. I simply got tired of waiting. I found an excellent driver at www.fs-driver.org.

--
Peter Bratton

Slow or stalled development is often a fact of life with small open-source software projects, which is why it's important to help support projects you find useful. Note that the driver provided at www.fs-driver.org is freeware, but it is not open source.—Ed.

Peace Prize?

From the Portland Linux/UNIX Group e-mail list, by Keith Loftstrom:

Since the Nobel Peace Prize is often given to politicians, some disagree with the choices. But it is often given to non-politicians who create international efforts to change the world for the better.

Look at the massive international efforts represented by SC09, and realize that much of it started from the work of a 21yo Finnish college student named after 1962 Nobel Peace Prize winner Linus Pauling. It would be fitting to honor that international effort by giving a Peace Prize to Linus Torvalds, perhaps in 2011 on the 20th anniversary of the August 1991 Linux announcement, or in 2012 on the 50th anniversary of Pauling's award.

Linux is one of the largest cooperative international efforts ever undertaken. It inspired Ubuntu, One Laptop Per Child, and many other global projects. Linux conquered the supercomputer space, the server space, the embedded computer space—by peaceful means! Linux helped sequence the human genome, helps protect the world computer infrastructure from viral attack, and is now the pathway for millions to learn computer programming and participate in new international efforts.

The 2007 Nobel Peace Prize recipient (a politician some disagree with, please disagree in a different thread, thanks) is giving the keynote to SC09 as I write this—meaning that we are all three handshakes away from the people that decide on future Peace Prizes. Perhaps it is time to launch some messages through our connections and see what makes it to the committee meetings in Oslo.

According to the list on Wikipedia, the five people to convince are Thorbjorn Jagland (chair), Kaci Kullmann Five (deputy chair), Sissel Ronbeck, Inger-Marie Ytterhorn, and Agot Valle. We can start by sending them Norsk language Ubuntu disks.

While I imagine Linus Torvalds would be embarrassed by the attention, it would sure make his parents happy. And it would mean one less Peace Prize for a politician.

Sounds like a great idea. What do you guys think?

--
Michael Rasmussen

A Peace Prize? For Linus? I dunno, I've read some of his posts to the kernel mailing list. Hehehe, all joking aside, I think the community Linus represents certainly deserves recognition. Since the prize goes to an individual, it does take some of the focus away from some other amazing contributors. That said, I could think of many worse recipients. He's got my vote!—Ed.

Security Tip

Mick, hope you enjoyed DEFCON, and excellent article in October 2009 issue of *Linux Journal*. It's moot now, but I thought I'd mention that when I travel, I edit my /etc/hosts file with entries for important DNS names (my bank, my Webmail and so on) to reduce the chance someone is spoofing them on an untrusted LAN. I comment out the entries when I get back home. I don't know if this really adds to my security, but I pretend it does. Thanks for the great work.

--
Paul

Mick Bauer replies: As a matter of fact, your /etc/hosts idea is an excellent one. DNS spoofing is an important part of many man-in-the-middle attacks, so being able to skip DNS lookups altogether when using untrusted networks definitely would be useful.

It also may be possible to run a local DNS-caching daemon like nsqd (which is commonly included in many distros by default), tuned in such a way that if you visited a given important site before you travel, your system will use its local cached lookup results instead of doing new DNS lookups while you're on the road. Thanks for passing on your idea and your kind words!

Decaf, Amazon EC2 on Android

As a longtime *Linux Journal* reader (I started when I was still studying more than ten years ago), I would like to draw your attention to decaf. decaf is an Android application for managing and monitoring your Amazon EC2 infrastructure. We were finalists in the Android Developer Challenge 2, resulting in a sixth place in the Misc. Category. (We were a bit disappointed but very proud to have come that far in the competition.)

We developed decaf primarily for ourselves, but we are trying to grow a community to make decaf development sustainable. I see that you covered Amazon EC2 multiple times, therefore, I think decaf might be of interest to your community.

You can read about decaf at decaf.9apps.net. I hope you find this interesting. If you have any questions, please ask.

--
Jurg van Vliet

[LETTERS]

Cool! Thanks Jurg. I just bought a Droid, so I'll have to check it out.—Ed.

Re: Ruby Articles

In response to the letter in the January 2010 issue regarding Ruby articles: I'd suggest looking into Clojure. It's a fairly new, Lisp-based language (it originally appeared in 2007 according to Wikipedia) that I first heard about from a professional Ruby programmer who now swears by it. He's written quite a few Clojure articles on his blog at briancarper.net/tag/clojure. Most of it is over my head as I've been out of the programming game for several years, but it has live code examples and might be an easier starting point than some dry manual or FAQ on clojure.org. Fun fact: that blog itself was written by Brian in Clojure.

Brian's a pretty easy guy to talk to and most likely would have some good recommendations on where to go to learn more about it or dig up material for an article.

As a side note, I'd personally be interested in seeing a roundup of all of the different languages that other *LJ* readers are no doubt sending e-mail messages about similar to this one even as we speak.

--
Marcus Huculak

Our next issue will have an interview with the creator of Clojure. And, perhaps we can get our Webmistress to put up a language poll on LinuxJournal.com. It would be great information to have!—Ed.

Linux Mini vs. Mac Mini

I'm surprised I don't see Linux alternatives to a Mac Mini. The alternative hardware needs to be small and quiet (like Mac Mini). Hardware suppliers like Logic Supply and Polywell have a dizzying selection of hardware,

but the cost is more than \$1,000. Why isn't there a selection of "Linux Mini" alternatives equal or better than a Mac Mini at competitive prices?

For reference, a \$600 Mac Mini (www.apple.com/macmini) features 2.26GHz Intel Core 2 Duo, 2GB DDR3 SDRAM, 160GB hard drive, gigabit Ethernet, 8x double-layer SuperDrive, NVIDIA GeForce 9400M graphics with dual video ports, USB and Firewire ports, Mac OS X Snow Leopard and 14 W power at idle.

--
greg bollendonk

I think one of the problems is that the demand is so low. For hardware companies, I think creating a Linux-based alternative to the Mac Mini would be possible, but they'd most likely sell more if they just made Windows terminals out of them.

One way to build a device like you're describing would be to soup up a thin client. There are a bunch of places that sell Linux thin clients that easily could have a hard drive added to them. Polywell, for example, has several thin-client options that are full-blown computers (at least one less than \$200) just waiting for a Linux install.—Ed.

PHOTO OF THE MONTH

Have a photo you'd like to share with *LJ* readers? Send your submission to publisher@linuxjournal.com. If we run yours in the magazine, we'll send you a free T-shirt.



Here is a picture from our wedding on November 21, 2009 in Sarasota, Florida. Look at the tie! How cool is my wife?
Submitted by Kevin P. Biggs.

LINUX JOURNAL

At Your Service

MAGAZINE

PRINT SUBSCRIPTIONS: Renewing your subscription, changing your address, paying your invoice, viewing your account details or other subscription inquiries can instantly be done on-line, www.linuxjournal.com/subs. Alternatively, within the U.S. and Canada, you may call us toll-free 1-888-66-LINUX (54689), or internationally +1-818-487-2089. E-mail us at subs@linuxjournal.com or reach us via postal mail, Linux Journal, PO Box 16476, North Hollywood, CA 91615-9911 USA. Please remember to include your complete name and address when contacting us.

DIGITAL SUBSCRIPTIONS: Digital subscriptions of *Linux Journal* are now available and delivered as PDFs anywhere in the world for one low cost. Visit www.linuxjournal.com/digital for more information or use the contact information above for any digital magazine customer service inquiries.

LETTERS TO THE EDITOR: We welcome your letters and encourage you to submit them at www.linuxjournal.com/contact or mail them to Linux Journal, PO Box 980985, Houston, TX 77098 USA. Letters may be edited for space and clarity.

WRITING FOR US: We always are looking for contributed articles, tutorials and real-world stories for the magazine. An author's guide, a list of topics and due dates can be found on-line, www.linuxjournal.com/author.

ADVERTISING: *Linux Journal* is a great resource for readers and advertisers alike. Request a media kit, view our current editorial calendar and advertising due dates, or learn more about other advertising and marketing opportunities by visiting us on-line, www.linuxjournal.com/advertising. Contact us directly for further information, ads@linuxjournal.com or +1 713-344-1956 ext. 2.

ON-LINE

WEB SITE: Read exclusive on-line-only content on *Linux Journal*'s Web site, www.linuxjournal.com. Also, select articles from the print magazine are available on-line. Magazine subscribers, digital or print, receive full access to issue archives; please contact Customer Service for further information, subs@linuxjournal.com.

FREE e-NEWSLETTERS: Each week, *Linux Journal* editors will tell you what's hot in the world of Linux. Receive late-breaking news, technical tips and tricks, and links to in-depth stories featured on www.linuxjournal.com. Subscribe for free today, www.linuxjournal.com/eneWSletters.



Consistency Breeds Predictability

Security Blanket ensures enterprise-wide security policy configuration.

How much time do you spend hardening all of your Linux and Solaris servers (assuming you are hardening all of them)? Are you confident that they are all locked down to a consistent, secure state?

Security Blanket automates the lock down process so that you can regain control of your time and have confidence that your servers are secure. Security Blanket will assess your server environment against industry standard security guidelines (such as DISA STIGs, CIS, and SANS Top 20 Security Risks) and automatically configure the OS to comply with security policy.

Saving you time, reducing security risk, and consistently complying with policy is what Security Blanket is all about!

Try it for free at www.TrustedCS.com/SecurityBlanket or call us at 1-866-230-1317 for more information.



diff -u

WHAT'S NEW IN KERNEL DEVELOPMENT

Sam Ravnborg has handed off **KBuild** maintainership to **Anibal Monsalve** and **Michal Marek**. They hadn't planned it this way—both of them just volunteered to take over, and Michal suggested a co-maintainership. A pile of big-time kernel folks thanked Sam for doing all the work he did on it. He certainly is handing off a very robust and reliable element of the kernel. We'll see what direction Anibal and Michal take it now.

In spite of **Linus Torvalds'** proclamation that there was room only for a single process scheduler in the kernel, that doesn't stop people from wanting to roll their own and use it. **Pankaj Parakh** is one of these, and **Peter Williams** has been working on reviving the **CPU plugin scheduler project** on his own (to be renamed CPU_PISCH). In fact, the two of them may now be working on that together. They're also each naturally designing their own schedulers to plug in to the kernel, once they get CPU_PISCH ready. There really are a lot of schedulers out there, partly because it's a really cool, challenging programming project, and partly because it's so fraught with deep magic that it would be difficult for everyone to agree on the best ways for any scheduler to behave.

LogFS, after much struggle, is now headed for inclusion in the main kernel. The main roadblock was that its on-disk format was in flux, and including it in the kernel during that kind of change would

create support nightmares inside the kernel, because users of each fluctuation of the disk format still would need to access their data, long after LogFS had settled on a final format for itself. There were other issues as well, but that was the main one. **Jörn Engel** recently submitted the updated LogFS to the Linux-Next tree, which typically means something will be headed in a fairly standardized way up into the official tree. Not that it's impossible for something to stall out between Linux-Next and the official kernel, but it's not the usual case.

A new development policy is in the works, allowing **subsystem maintainers** to migrate drivers they don't like into the **staging tree**. The staging tree is just a relatively new directory in the official kernel, where drivers that are not ready for true inclusion can hang out and be available to users. It's a way to get lots of testing for things that are still up and coming. Typically, the path is from outside the kernel, to the staging tree, to a proper place in the kernel sources. The proposal is to reverse that direction at the discretion of the subsystem maintainers. There are plenty of pros and cons, and the debate is fairly heated. Undoubtedly, the policy's true algorithm will settle down over time, but something along those lines does seem to have Linus Torvalds' support and the support of a lot of other folks. It's definitely something that's going to be

worked out in the field, rather than made perfect beforehand. We can look forward to angry complaints from driver maintainers, and probably some users, until the kinks are worked out.

John Hawley took advantage of a momentary lull on **master.kernel.org**, due to it being nighttime in Japan during the kernel summit this past year, and upgraded the operating system on that server. He reported that the upgrade went fairly well, with a reboot and a six-hour configuration effort. At the time he reported on it, there were only a few remaining glitches left to iron out.

LTtng (Linux Trace Toolkit Next Generation) version 0.164 will be released under some additional licenses. It'll still have the GPL, but now some of the code will also be released under the LGPL, and some will be released under the BSD license. **Mathieu Desnoyers** made the announcement and also said he was still waiting for **IBM** to give its permission to include its contributions in the relicensing effort.

Michael Cree and **Matt Turner** have joined forces in common frustration at the large number (more than a dozen) of unimplemented system calls on the **Alpha architecture**. They plan to work together to implement them, once they can figure out some of the tricky technical issues standing in the way.

—ZACK BROWN

LINUX JOURNAL INSIDER

If you're the type of *Linux Journal* reader who waits by your mailbox every month, setting up a tent to sleep in and taking days off work as you anxiously await the new issue, quite frankly, we want you to seek medical attention. If you're just a *Linux Journal* fan who would like to hear about the issue as it rolls off the presses, but before it is actually in your hands, we've got a special treat for you.

This year, Kyle Rankin and I are putting together a

monthly podcast called, "*Linux Journal Insider*", where we give you the ins and outs of the upcoming issue. We discuss the issue focus, read letters to the editor and usually manage to throw in a few bad puns and cheesy jokes along the way. Swing by the Web site (www.linuxjournal.com) and look for the RSS feed. It's fun for us to talk about the issue we've been working on and hopefully fun for you to hear about it!

—SHAWN POWERS

If your company or project has news you'd like to see on *LinuxJournal.com*, send your announcement to news@linuxjournal.com. Although we can't run everything, it all gets a look.

NON-LINUX FOSS

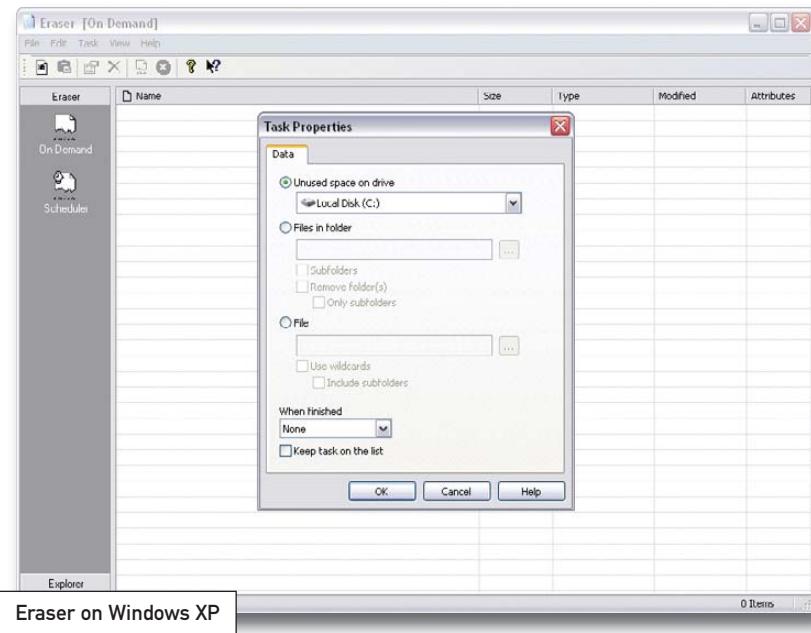
If you're the paranoid type and you're still stuck using Windows, you need to get Eraser. Eraser is an open-source security tool for Windows that makes sure deleted files are erased and overwritten completely before they are deleted.

Many of us assume that when we delete a file, it's gone, but that's rarely the case. Most deletions merely mark the file as being deleted and recycle that area of the disk, nothing is written to it until the space is needed by a new file or by the growth of an existing file.

Even after the area is overwritten, there are some among us (and you know who you are) who believe by a careful analysis of the magnetic fields on the disk surface, one could reconstruct deleted data.

Even if you don't buy into the black helicopter scenario, there's no doubt that, at least for a time, your deleted files may still be accessible. That's where Eraser comes in. It overwrites the file with other data before it deletes the file. And, that's not all. Eraser not only overwrites the disk area used by the file, it also actually gets out a knife and scrapes off that part of the disk surface that contained the file (just kidding about that last part).

Eraser runs on all modern versions of Windows from Windows 95 on. It includes its own user interface as well as Windows Explorer extensions. The current version is 5.8.7 and is available from eraser.heidi.ie.—**MITCH FRAZIER**



Become a Ninja on LinuxJournal.com

Okay, so it is probably *slightly* more complicated than that, but a few visits to LinuxJournal.com certainly will help put you on the path to ninja status. I can only assume that you are reading the awesome collection of goodness that is this month's *Linux Journal* with the intention of becoming a Linux ninja or adding to your collection of ninja weapons. I am here to help you on that quest. You see, although there is ample ammunition here in your hands, a few essential bits of arsenal inventory are lurking on-line at LinuxJournal.com. In particular, the entire HOW-TOs section is full of such gems. One of my favorites from the vault is Shawn Powers' video tech tip on command-line substitution: "Forgetting Sudo (we've all done it)" at www.linuxjournal.com/video/forgetting-sudo-weve-all-done-it.

You never know when you'll happen upon a pirate in a dark alley and be very glad you spent some time on LinuxJournal.com. Oh, it'll happen. Trust me.

—**KATHERINE DRUCKMAN**

LJ Index March 2010

1. Number of times "Ubuntu" was used in LinuxJournal.com posts during 2009: **329**
2. Number of times "openSUSE" or "SUSE" was used: **68**
3. Number of times "Debian" was used: **116**
4. Number of times "Red Hat" was used: **58**
5. Number of times "Fedora" was used: **49**
6. Number of times "CentOS" was used: **2**
7. Number of times "Gentoo" was used: **13**
8. Number of times "Ubuntu" was used in *Linux Journal* print articles during 2009: **343**
9. Number of times "Debian" was used: **88**
10. Number of times "openSUSE" or "SUSE" was used: **47**
11. Number of times "Red Hat" was used: **49**
12. Number of times "Fedora" was used: **51**
13. Number of times "CentOS" was used: **17**
14. Number of times "Gentoo" was used: **8**
15. Number of *Linux Journal* issues with a Programming or Development focus (1994–2009): **14**
16. Number of *Linux Journal* issues with a System Administration focus: **11**
17. Number of *Linux Journal* issues with a Security focus: **10**
18. Number of *Linux Journal* issues with an Embedded focus: **7**
19. Number of *Linux Journal* issues with an Ultimate Linux Box focus: **5**
20. Number of *Linux Journal* issues with a Desktop focus: **4**

Sources: 1–20: mysql/sed/grep/wc

IRC, Still the Best Support Around

If you haven't gotten our subtle hints during the past year or so, IRC certainly is not dead. It really is the best way to get knowledgeable support from the folks who know best. There are a few caveats, however, that may not be obvious to people new to this old-school chat protocol.

Get a Good Client

If you just want to stop into the #linuxjournal channel for some quick banality, a Web-based client like the one at linuxjournal.com/irc is fine. You can drop in, request a !coffee from JustinBot, and chitchat with fellow geeks. If you're looking for something a bit more useful for the long haul, a native client makes more sense. Many people (myself included) like X-Chat. There are plenty of other options, like the command-line-only Irssi, but X-Chat offers a nice balance between features and usability.

If you look back at Kyle Rankin's Hack and / articles from the past year or so,

you'll find easy ways to integrate your entire lifestyle into IRC. Kyle does everything from chatting to twittering inside his terminal window, and he shows us all how to do the same.

The opposite approach, which is actually what I do, is to add IRC as another instant-messaging protocol on my IM client. Although Kopete and Empathy may be slick-looking for instant messaging, none come close to Pidgin's elegance with IRC. Check out my video tech tip on how to set up IRC inside Pidgin if that makes more sense to the way you work during the day: www.linuxjournal.com/video/irc-chats-pidgin.

IRC Etiquette

Every channel you visit will have a different "personality" to it. The #linuxjournal channel on Freenode, for example, is really a goofy, easy-going channel full of geeks having fun. If you come visit us and say "Garble bangle, loopy loo", no one will find you odd.

In fact, you'll fit in quite nicely. On other channels, specifically channels where developers hang out related to a specific application, the atmosphere might be a bit more stuffy. My suggestion: hang out in a room for a while before you post questions. There may be links in the channel pointing to FAQs or information about how to conduct yourself without making anyone angry.

Be Patient

IRC is the sort of thing most geeks leave running but don't monitor constantly. If you pose a question, but don't get a response for a while, just wait. If you have a question for a specific person, typing his or her name in the channel often will alert the person (I have Pidgin set up to do that, and many folks do the same with their IRC clients). And finally, don't forget, it's a community. If you see a question you can answer, do it!

—SHAWN POWERS

Sync Your Life

For those of us lucky enough to use Linux on all of our computers, Canonical's Ubuntu One is a great way to keep files in sync between computers. Unfortunately, most of us are stuck using other operating systems throughout the day. We all have our own ways of managing such things, but I thought a glimpse into my "world of sync" might help others synchronize their lives.

Files

At home, I have a centralized file server, and at work, I have the same thing. But, sometimes I want to access documents regardless of my location—like from a coffee shop during lunch. For my word processing and spreadsheet files, along with a handful of other commonly used documents (*Linux Journal* digital PDFs come to mind), I use Dropbox. It is a cross-platform, free program that allows you to sync many computers in real time. The free version is limited to a gig or

two, but for basic documents, it's perfect (www.dropbox.com).

Bookmarks

I use Firefox on every operating system, but even if you are forced to use Internet Explorer, Safari or Google's Chrome browser, Xmarks syncs your bookmarks quite nicely between different browsers on different platforms. The service is free and works very well. I can't imagine life without Xmarks (www.xmarks.com).

Contacts and Calendars

Love it or hate it, Google has infiltrated every operating system rather effectively. I use a plethora of applications to keep my different devices (laptops, desktops, phones, PDAs) in sync with contacts and calendars, but they all are based on Google. My favorite feature is that in a pinch, I can access everything from a Web browser. A quick search for "google sync" brings up many options, most free, that

should get you a consistent contact and calendar base across any platform.

Voicemail

This is starting to feel like a Google ad, so I'll stop with this one. Google Voice is the way I consolidate all my phone numbers. I like having a single number that I can give freely and then filter incoming calls however I want. Again a free solution, Google Voice offers features I'd likely pay for, although I'm certainly not complaining at the price.

So, there you have it. I currently have two cell phones, a Skype Wi-Fi phone, Magic Jack, home landline, work landline, three Linux laptops, one Windows laptop, one Apple laptop, three desktops at home, three desktops at work and enough media-playing devices in my house to open a movie theater. If I didn't sync some of my services, I'd go more insane than I already am!

—SHAWN POWERS

Stupid tar Tricks

One of the most common programs on Linux systems for packaging files is the venerable tar. tar is short for tape archive, and originally, it would archive your files to a tape device. Now, you're more likely to use a file to make your archive. To use a tarfile, use the command-line option `-f <filename>`. To create a new tarfile, use the command-line option `-c`. To extract files from a tarfile, use the option `-x`. You also can compress the resulting tarfile via two methods. To use bzip2, use the `-j` option, or for gzip, use the `-z` option.

Instead of using a tarfile, you can output your tarfile to stdout or input your tarfile from stdin by using a hyphen (`-`). With these options, you can tar up a directory and all of its subdirectories by using:

```
tar cf archive.tar dir
```

Then, extract it in another directory with:

```
tar xf archive.tar
```

When creating a tarfile, you can assign a volume name with the option `-V <name>`. You can move an entire directory structure with tar by executing:

```
tar cf - dir1 | (cd dir2; tar xf -)
```

You can go even farther and move an entire directory structure over the network by executing:

```
tar cf - dir1 | ssh remote_host "( cd /path/to/dir2; tar xf - )"
```

GNU tar includes an option that lets you skip the `cd` part, `-C /path/to/dest`. You also can interact with tarfiles over the network by including a host part to the tarfile name. For example:

```
tar cvf username@remotehost:/path/to/dest/archive.tar dir1
```

This is done by using rsh as the communication mechanism. If you want to use something else, like ssh, use the command-line option `--rsh-command CMD`. Sometimes, you also may need to give the path to the rmt executable on the remote host. On some hosts, it won't be in the default location `/usr/sbin/rmt`. So, all together, this would look like:

```
tar -c -v --rsh-command ssh --rmt-command /sbin/rmt
-f username@host:/path/to/dest/archive.tar dir1
```

Although tar originally used to write its archive to a tape drive, it can be used to write to any device. For example, if you want to get a dump of your current filesystem to a secondary hard drive, use:

```
tar -cvzf /dev/hdd /
```

Of course, you need to run the above command as root. If you are writing your tarfile to a device that is too small, you can tell tar to do a multivolume archive with the `-M` option. For

those of you who are old enough to remember floppy disks, you can back up your home directory to a series of floppy disks by executing:

```
tar -cvMf /dev/fd0 $HOME
```

If you are doing backups, you may want to preserve the file permissions. You can do this with the `-p` option. If you have symlinked files on your filesystem, you can dereference the symlinks with the `-h` option. This tells tar actually to dump the file that the symlink points to, not just the symlink.

Along the same lines, if you have several filesystems mounted, you can tell tar to stick to only one filesystem with the option `-1`. Hopefully, this gives you lots of ideas for ways to archive your files.

—JOEY BERNARD

They Said It

All scenarios likely to result from Oracle's acquisition of the [MySQL] copyrights, whatever Oracle's business intentions may be, are tolerable from the point of view of securing the freedom of the codebase.

—Eben Moglen, a Columbia University Law School professor, director of the Software Freedom Law Center

The real problem is not whether machines think but whether men do.

—B. F. Skinner

There are three roads to ruin: women, gambling and technicians. The most pleasant is with women, the quickest is with gambling, but the surest is with technicians.

—George Pompidou

Technology is dominated by two types of people: those who understand what they do not manage, and those who manage what they do not understand.

—Archibald Putt

Fine print: All prices are final, there are no bogus fees and unfees. Period. Only SIP devices that have already been created can be connected to sip.callwithus.com to make calls. Please ensure you only use devices approved by you (please do not try and connect using two tin cans and a piece of string, as we do not yet support this, but we may support this in the future—the work is in progress and preliminary results are positive). Callwithus.com monthly subscription charge of \$0 must be paid in advance and does not include tax of \$0, which also must be paid in advance. You will be billed an activation fee of \$0 plus tax and this must be paid in advance. Calls made incur tax at the rate of 0% each month and must be paid in advance. On cancellation of the service you will be charged a one-time disconnection charge of \$0. Additional features such as caller ID with name on incoming calls will be billed at the additional rate of \$0 per call. All **YOUR** rights reserved.

—The “Fine Print” from callwithus.com



REUVEN M. LERNER

Testing JavaScript

A look at Screw.Unit, a framework for JavaScript testing.

In mid-November 2009, at a meeting of my research group in Chicago, I proudly unveiled the most recent beta of the software I'm writing for my PhD dissertation, a Web application (written in Ruby on Rails) that promotes collaboration among students and scientists. I was pretty confident the testing would not reveal too many technical problems, in part because I had used Cucumber and rcov to ensure a high degree of test coverage. True, my application uses some AJAX, which means there are certain things Cucumber cannot test. But, given how localized such functions are, and the fact that I used and tested them myself on a day-to-day basis, how much did it matter?

The good news is that for the most part, the beta test went quite well. There were a few problems to fix, and I started to come up with a plan to get to them. What bothered me most was not that bugs existed, but rather that the bugs were all related to JavaScript and AJAX. In other words, the high level of test coverage that I had achieved was good, but it was not sufficient, because it looked only at my Ruby code and not at the equally important JavaScript code in my application.

This was not the first time I had encountered issues with JavaScript testing. A project I worked on through much of 2009 used a great deal of JavaScript, and we tried to test it in a number of ways, none of which were particularly satisfactory.

So, I was pleasantly surprised to discover I'm not the only Web developer who has been trying to improve test coverage for Web applications that include a great deal of JavaScript. Indeed, currently a number of frameworks and libraries are available for JavaScript testing—some of which are specific to a particular JavaScript framework, some of which are plugins for Ruby on Rails (or other Web application frameworks) and still others that are fairly flexible and agnostic.

This month, I look at Screw.Unit, a framework for JavaScript testing I have begun to use in my own work. Even if you don't use Screw.Unit specifically, modern Web developers constantly must consider ways to write testable code, not only in their server-side language of choice, but also in JavaScript. JavaScript plays a central role in modern Web applications, and failing to test it thoroughly can lead to unforeseen problems, as I saw myself.

Downloading and Installing Screw.Unit

Screw.Unit originally was written by Nick Kallen (of Pivotal Labs) and distributed as open source on GitHub. A number of forked versions exist, and you might need to poke around to find one that is sufficiently mainstream and modern for your needs. I have been using Kallen's original version and rely on it for this article's examples. GitHub provides a number of methods for downloading software, but the easiest is just to "clone" the existing Git repository, with:

```
git clone git://github.com/nkallen/screw-unit.git
```

Inside the screw-unit directory, you will find a number of JavaScript libraries and CSS files, all of which are there to assist you when running JavaScript tests.

The basic idea of Screw.Unit is that you introduce a set of related tests with `describe()` and then each individual test with `it()`. The second parameter to `it()` is a function that invokes one or more assertions, using the defined `expect()` function.

Thus, let's assume you have a function defined that multiplies its parameter by 3:

```
function triple(i) {
    return i * 3;
}
```

You can test it in Screw.Unit with the following:

```
describe("Triple should triple", function() {
    it("returns 6 for 2", function() {
        expect(triple(2)).to(equal, 6);
    });
});
```

Notice the three separate levels of functions that are involved here:

- `describe` introduces a block of common specifications.
- `it` describes and introduces a single specification.
- `expect` executes one test for that specification.

In order to run these tests, you need to wrap the entire `describe` block inside an anonymous function,

Listing 1. triple.html

```
<html>
<head>
<script src="lib/jquery-1.2.6.js"></script>
<script src="lib/jquery.fn.js"></script>
<script src="lib/jquery.print.js"></script>
<script src="lib/screw.builder.js"></script>
<script src="lib/screw.matchers.js"></script>
<script src="lib/screw.events.js"></script>
<script src="lib/screw.behaviors.js"></script>
<link rel="stylesheet" href="lib/screw.css">

<!-- Here is the function we define, to test --&gt;
&lt;script type="text/javascript"&gt;
    function triple(i) {
        return i * 3;
    }
&lt;/script&gt;

<!-- Here is the test itself --&gt;
&lt;script type="text/javascript"&gt;
    Screw.Unit(function() {
        describe("Triple should triple", function() {
            it("returns 6 for 2", function() {
                expect(triple(2)).to(equal, 6);
            });
        });
    });
&lt;/script&gt;
&lt;/head&gt;
&lt;body&gt;
&lt;/body&gt;
&lt;/html&gt;</pre>
```

passed as the first parameter to Screw.Unit():

```
Screw.Unit(function() {
    describe("Triple should triple", function() {
        it("returns 6 for 2", function() {
            expect(triple(2)).to(equal, 6);
        });
    });
});
```

Finally, you need to pull in a bunch of JavaScript libraries that not only define Screw.Unit, but also the objects and functions on which it relies. The final version is shown in Listing 1, triple.html. Notice that while you are testing JavaScript, Screw.Unit assumes you are doing so within an HTML file. That allows you not only to load an (unfortunately long) list of JavaScript libraries, but also the CSS file that is used within Screw.Unit to display test results.

The test is passed when Screw.Unit() is executed.

If it works well, the body of the HTML document is modified accordingly, using CSS classes (defined in screw.css) that give the traditional green (for passing) or red (for failing) report on the tests you performed.

I'm going to add two more tests, one that I know will pass, which uses the `not_equal` test modifier. The other test will fail, so you can examine what it looks like when one does. If all goes well, you should see two green bars and one reddish-brown bar, the last of which indicates failure. The test itself looks like this:

```
<script type="text/javascript">
Screw.Unit(function() {

    describe("Triple should triple", function() {
        it("returns 6 for 2", function() {
            expect(triple(2)).to(equal, 6);
        });

        it("does not return 100 for 2", function() {
            expect(triple(2)).to_not(equal, 100);
        });

        it("does return 100 for 2 -- fail!", function() {
            expect(triple(2)).to(equal, 100);
        });
    });
}</script>
```

As you can see, you can include as many `it` statements inside a `describe` block as you need. Over time, you will see your spec grow to contain more and more descriptions, `it` statements and `expect` statements.

Checking the DOM

Testing JavaScript functions is certainly a good thing to do. However, one of the primary uses of JavaScript is to modify the DOM—the document object model that provides a handle onto the contents of an HTML page. Using the DOM, you can view or modify the page, both in its tags and in its content.

Thus, DOM manipulations are a primary target for JavaScript tests. You want to be able to say that when a particular piece of HTML is clicked on, another piece of HTML should appear.

Now, some of the documentation for Screw.Unit will tell you that you can (and should) use a `click()` method to simulate clicking on an element of your page. I not only found the `click()` method to be unreliable, but also was persuaded by a posting on the Screw.Unit mailing list to put my text-hiding code in a separate function, which can then be

Listing 2. clickview.html

```

<html>
<head>
<script src="lib/jquery-1.2.6.js"></script>
<script src="lib/jquery.fn.js"></script>
<script src="lib/jquery.print.js"></script>
<script src="lib/screw.builder.js"></script>
<script src="lib/screw.matchers.js"></script>
<script src="lib/screw.events.js"></script>
<script src="lib/screw.behaviors.js"></script>
<link rel="stylesheet" href="lib/screw.css">


<script type="text/javascript">
    function hide_paragraph() {
        $("#hideme").hide();
    }

    $(document).ready(function() {
        $('#hideme').click(function() {
            hide_paragraph();
        });
    });
</script>

<!-- Here is the test itself --&gt;
&lt;script type="text/javascript"&gt;
    Screw.Unit(function() {

        describe("Paragraph", function() {

            it("should be hidden when clicked", function() {
                hide_paragraph();
                expect($('#hideme').is(':hidden')).to(equal, true);

            });
        });
    });
&lt;/script&gt;

&lt;/head&gt;
&lt;body&gt;
&lt;p id="hideme"&gt;Click to hide&lt;/p&gt;
&lt;/body&gt;
&lt;/html&gt;
</pre>

```

called from within the click() handler for the paragraph and also from the test within an it block. This not only worked, but also encouraged a style that is more readable and easily workable, in my opinion.

The full file, clickview.html, is in Listing 2. The idea is that the document contains a paragraph:

```
<p id="hideme">Click to hide</p>
```

You then attach a click() event handler to the paragraph, invoking a function when the paragraph is clicked on:

```

function hide_paragraph() {
    $("#hideme").hide();
}

$(document).ready(function() {
    $('#hideme').click(function() {
        hide_paragraph();
    });
});

```

Finally, you set up a Screw.Unit() test block, as follows:

```

Screw.Unit(function() {

    describe("Paragraph", function() {

        it("should be hidden when clicked", function() {
            hide_paragraph();
            expect($('#hideme').is(':hidden')).to(equal, true);

        });
    });
});

```

When you load the page, Screw.Unit first invokes the function hide_paragraph(), which has the same effect that clicking on it would have. Then it checks to make sure, using a pseudo-class (:hidden) to identify hidden text. If no text with the ID "hideme" is currently hidden, jQuery returns an empty list, and the assertion fails.

The fact that everything in Screw.Unit, as in jQuery, is done using CSS selectors makes it easy and fast to work with. It would seem that there are people doing TDD (test-driven development) and BDD (behavior-driven development) using Screw.Unit; although I don't count myself among those, I do see myself using this sort of testing in the future, if only to avoid egg on my face among my software users. Besides, testing JavaScript in this way, at least to my mind, gives me a feeling of working on serious software, rather than one or more basic hacks.

I should note that the style in which I presented Screw.Unit in this column is a concise, but not idiomatic way that most users will put it into practice. Screw.Unit users typically separate tests into multiple files, allowing them not only to define custom test matchers, but also to have an entire library of tests, rather than just one file. Once you get started with Screw.Unit, I'm sure you will find a style that suits your needs,

without running too much against the grain of the system's expected use.

Conclusion

Screw.Unit is an easy-to-understand, easy-to-use framework for testing your JavaScript code. It is not the only test system of its kind, but the fact that its syntax is reminiscent of RSpec makes it easier for people like me, who like and use RSpec, to start using it quickly. RSpec advocates also will want me to point out that Screw.Unit offers JavaScript developers the same sort of BDD that characterizes RSpec and Cucumber, focusing on the results that the user sees, rather than the internal workings.

If you have never tested your JavaScript before, there's no time like the present to begin! If nothing else, you want to be sure that clicking on various parts of your HTML page does not lead to errors.■

Reuven M. Lerner, a longtime Web/database developer and consultant, is a PhD candidate in learning sciences at Northwestern University, studying on-line learning communities. He recently returned (with his wife and three children) to their home in Modi'in, Israel, after four years in the Chicago area.

Resources

The home page for the main version of Screw.Unit is at GitHub at github.com/nkallen/screw-unit. The documentation on that page is somewhat sparse, but it offers several examples of how to create and use Screw.Unit tests.

A small tutorial for Screw.Unit, also hosted at GitHub, is at github.com/bsigglekow/screw-unit-tutorial. This tutorial uses the Sinatra framework for Web applications, so you need to have a copy of Ruby and the Sinatra gem for Ruby in order to get started.

There is an e-mail list for Screw.Unit users, and you can subscribe at groups.google.com/group/screw-unit.

A blog post, based on the e-mail message that showed me the importance of not using the click() method and describing in greater detail how to write better tests, is available at blog.runcoderun.com/post/177871245.

jQuery, the JavaScript library used in Screw.Unit and one you might want to explore for your own in-browser applications, is at jquery.org.



visit us at www.siliconmechanics.com
or call us toll free at 866-352-1173

As a Senior Account Executive for Silicon Mechanics, Michael collaborates with customers to expertly match hardware with processing needs. Lately he has been inviting a good many of those customers to have a close look at the Hyperform HPCg R2504, powered by NVIDIA Tesla. This workstation has earned its place among our most popular products for very good reasons.

We start with two Quad-Core Intel® Xeon® Processor 5500 Series CPUs, for fast, reliable, energy-efficient processing. Then we add up to four NVIDIA Tesla C1060 GPUs, to dramatically accelerate parallel processing for applications like ray tracing and finite element analysis. With dual-IOH design, the system provides non-blocking connectivity between the GPUs and CPUs to maximize system performance. Include up to 96GB of registered DDR3 memory and you have the power of a cluster in a workstation form factor at a price you don't want to miss.

When you partner with Silicon Mechanics, you get more than collaborative service and affordable performance—you get an expert like Michael.

For configuration and pricing on the
Hyperform HPCg R2504 visit
www.siliconmechanics.com/R2504



**Powerful.
Intelligent.**



Expert included.

Silicon Mechanics and the Silicon Mechanics logo are registered trademarks of Silicon Mechanics, Inc. Intel, the Intel logo, Xeon, and Xeon Inside, are trademarks or registered trademarks of Intel Corporation in the US and other countries.



DAVE TAYLOR

Still Parsing the Twitter Stream

How do you keep track of which tweets you've already answered?

Last month, you'll hopefully remember that we took the big step in our Twitter stream parsing program of actually having it parse the incoming messages and strip out quotes and other HTML noise. I also republished the send-tweet script too, which we'll use this month.

The biggest challenge we face with the tweet-parser is knowing what messages we've already answered and which are new since the last time the program was run. The solution? To go back and tweak the original script a bit. It turns out that each and every tweet has a unique ID value, as you can see here:

```
<id>2541771</id>
```

You'll recall that early in the script we have this grep command:

```
grep -E '<screen_name>|<text>' | \
```

Simple enough. We'll tweak it to include |<id> and grab that value too. Except, of course, it's not that simple. It turns out that two <id> strings show up in the XML data from Twitter: one that's the ID of the account sending the message, and another

The biggest challenge we face with the tweet-parser is knowing what messages we've already answered and which are new since the last time the program was run.

that's the ID of the message itself—both conveniently labeled the same. Ugh!

Timestamps and Tricky XML

I can kvetch and wish Twitter would fix its XML to have USERID or similar, but what's the point? They have the same thing with the overloaded <created_at> tag too, so we're going to have to bite the bullet and accept that we are now grabbing four data fields from the XML feed, only three of which we care about.

Once we know that we're going to have four lines of output, cyclically, we simply can decide

which of those are actually important and tweak them in the awk statement:

```
$curl -u "davetaylor:$pw" $inurl | \
grep -E '<screen_name>|<text>|<id>' | \
sed 's/@DaveTaylor //;s/ <text>//;s/</text>/' | \
sed 's/ *<screen_name>//;s/</</screen_name>/' | \
sed 's/ *<id>//;s/</>/' | \
awk '{ if (NR % 4 == 0) {
        printf ("name=%s; ", $0)
    } else if (NR % 4 == 1) {
        printf("id=%s; ",$0)
    } else if (NR % 4 == 2) {
        print "msg="" $0 """
    }
}' > $temp
```

That's a pretty complicated sequence, so let's look at the awk conditional statement a little closer. We have four input records (lines) that we're stepping through. The value of NR is the number of records processed so far. So if NR mod 4 equals 0, it's the first of the four records (lines). The first record is the name value.

Did you see that two lines have printf, and the third uses a simpler print statement? Since we want each set of variables on a separate line, we use the print statement, because it automatically appends a newline to the output. Of course, the same effect could be achieved by putting the newline as a format string passed to printf. Example output follows:

```
name=thattalldude; id=6507045947; msg="Rates?"
name=KateC; id=6507034680; msg="hours"
name=pbarbanes; id=6507033698; msg="thanks"
name=jodie_nodes; id=6507022063; msg=" $$?"
name=KateC; id=6507019757; msg="price"
name=tarahn; id=6507008559; msg="impact"
name=GaryH2UK; id=6507004771; msg="directions"
```

We're going to hand these again, line by line, to the eval statement to set the three variables: name, id and msg. Then, it's a simple parsing problem, comparing msg to the known queries we have. Basically, it's what we did last month, except this time, every single tweet also has a unique ID value associated with it.

A typical test might now look like this:

```
if [ "$msg" == "hours" ] ; then
    echo "@$name asked what our hours are in tweet $id"
fi
```

Nice! It's simple, straightforward and well worth the preprocessing hoops we've jumped through.

Working with IDs Included

Indeed, I run that against my Twitter stream (after asking people to send me sample queries), and here's what I see:

```
@TheNose100 asked what our hours are in tweet 6507436100
@crepeauf asked what our hours are in tweet 6507187325
@jdscott asked what our hours are in tweet 6507087136
@KateC asked what our hours are in tweet 6507034680
@inspiremetoday asked what our hours are in tweet 6506966654
```

I bet you can see how to proceed from here. We write static responses, calculate values as needed and use send-tweet to respond to the user:

```
$tweet "@$name our hours are Mon-Fri 9-5, Sat 10-4."
```

For fun, I'll let people send the query "time" and get the current output of the date command too, just to demonstrate how that might work. Here's the code block:

```
if [ "$msg" == "time" ] ; then
    echo "@$id asked for the time"
    $tweet "@$name the local time on our server is $(date)"
fi
```

Great. Got it all, except for where we started out. How do you track which tweets you've already answered?

But What Have We Already Seen?

The answer isn't that hard. The stream is newest to oldest, and the message ID values are assigned sequentially by the server, so all we need to do is cache the most recent message ID we've seen after we have answered all queries. Then, on subsequent invocations, compare each query ID to the most recent we've answered. If they're greater, we need to answer them. If not, we've already done so. Like this:

```
if [ "$id" == "$previouslatestid" -o $answered -eq 1 ] ; then
    echo "already answered query \"$msg\" from $name: skipped"
    answered=1
else
    ...
fi
```

The previouslatestid is what's cached. We'll also capture the most recent ID of the current wave of queries like this:

```
if [ -z "$latestid" ] ; then
    latestid=$id      # store most recent ID
fi
```

Of course, there are a few more steps. We need to grab the cached value at the beginning of the script:

```
if [ -f "$lastidcache" ] ; then
    previouslatestid=$(cat "$lastidcache")
else
    previouslatestid="0"
fi
```

And, we need to save it at the end:

```
echo $latestid > "$lastidcache"
```

That's it. I've run out of space, but the full script is available at <ftp://linxjournal.com/pub/lj/listings/issue191/10695.tgz>. Next month, we'll polish it a bit and see what fun we can have with a tweetbot! ■

Dave Taylor has been hacking shell scripts for a really long time. He's the author of the popular *Wicked Cool Shell Scripts* and can be found on Twitter as @DaveTaylor and more generally at www.DaveTaylorOnline.com.

EMBEDDED SERVER

- Fanless x86 500MHz/1GHz CPU
- 512MB/1GB DDR2 RAM On Board
- 4GB Compact Flash Disk
- 10/100 Base-T Ethernet
- Reliable (No CPU Fan or Disk Drive)
- Two RS-232 Ports
- Four USB 2.0 Ports
- On Board Audio
- Dimensions: 4.9 x 4.7 x 1.7" (125 x 120 x 44mm)
- Standard SIB (Server-In-a-Box) Starting at \$305 Quantity 1.
- Locked Compact Flash Access
- Analog SVGA 3D Video
- Optional Wireless LAN
- EMAC Linux 2.6 Kernel
- XP Embedded & WinCE 6.0

www.emacinc.com/servers/standard_sib.htm

EMAC, inc.
EQUIPMENT MONITOR AND CONTROL

Phone: (618) 529-4525 • Fax: (618) 457-0110 • www.emacinc.com



MICK BAUER

Linux VPNs with OpenVPN, Part II

Build a simple, secure VPN connection now!

Last month, I began a new series on how to build a Linux-based Virtual Private Network (VPN) solution using OpenVPN. I described what VPNs are, what they're used for, and I listed some popular ways of building VPNs with Linux. That column ended with some pointers for obtaining and installing OpenVPN.

This month, I continue with detailed instructions on how to build a quick-and-dirty single-user VPN connection that allows you to connect securely from some untrusted remote site, like a coffee shop wireless hotspot, back to your home network.

Quick Review

If you missed last month's column, here's a two-paragraph primer on VPNs. First, they're generally used for two things: connecting different networks together over the Internet and connecting mobile/remote users to some corporate or home network from over the Internet. In the first case, a VPN connection is usually "nailed"—that is, it stays up regardless of whether individual users actually are sending traffic over it. In the latter case, end users each create their own tunnels, bringing them up only as needed.

Several protocols are in common use for VPNs. The two most important of which are probably IPsec and SSL. IPsec is nearly always used to create an "encrypted route" between two networks or between one system and a network. In contrast, SSL, whether in the context of SSL-VPN (which uses a Web browser as client software) or in other SSL-based VPNs (like OpenVPN), can be used either to tunnel specific applications or entire network streams.

IPsec and SSL-VPN are out of the scope of this series of articles, which mainly concern OpenVPN. However, I will cover at least two different remote-access usage scenarios: single-user and multiuser. A later installment in this series may include site-to-site VPNs, which actually are simpler than remote-access solutions and which use a lot of

the same building blocks. If I don't cover site-to-site VPNs, or if you need to build one sooner than I get around to it here, you'll have little trouble figuring it out yourself even after just this month's column!

The Scenario

Let's get busy with a simple scenario: setting up a single tunnel to reach your home network from the local coffee shop (Figure 1).

In this simple example, a laptop is connected to a wireless hotspot in a coffee shop (Coffee Shop WLAN), which in turn is connected to the Internet. The laptop has an OpenVPN established with a server on the home network; all traffic between the laptop and the home network is sent through the encrypted OpenVPN tunnel.

What, you may wonder, is the difference between the hardware and software comprising the OpenVPN "server" versus that of the "client"? As it happens, the command `openvpn` can serve as either a server daemon or client daemon, depending on how you configure and run it. What hardware you run it on is totally up to you, although obviously if you're going to terminate more than a few tunnels on one server, you'll want an appropriately powerful hardware platform.

In fact, if you need to support a *lot* of concurrent tunnels, you may want to equip your server with one of the crypto-accelerator hardware cards ("engines") supported by OpenSSL (on which OpenVPN depends for its cryptographic functions). To see which are supported by your local OpenSSL installation, issue the command

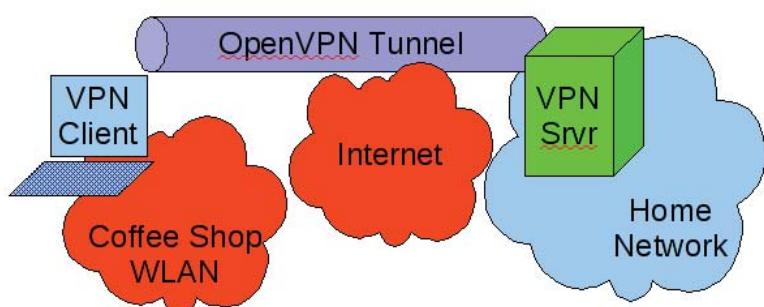


Figure 1. Remote-Access Scenario

`openvpn --show-engines`. See the documentation at www.openssl.org for more information on its support for crypto engines.

For this simple example scenario, let's assume both client and server systems are generic laptop or desktop PCs running current versions of some flavor of Linux with their respective distributions' standard OpenVPN and OpenSSL packages. The example OpenVPN configurations I'm about to walk through, however, should work with little if any tweaking on any of OpenVPN's supported platforms, including Windows, Mac OS X and so forth.

Although this scenario implies a single user connecting back to the home server, the configurations I'm about to describe can just as easily support many users by changing only one server-side setting (`max-clients`) and by generating additional client certificates. Have I mentioned certificates yet? You'll need to create a Certificate Authority (CA) key, server certificate and at least one client certificate. But have no fear, OpenVPN includes scripts that make it quick and easy to create a homegrown Public Key Infrastructure.

Configuring the Server

Let's get to work configuring the server. Here, I explain how to create a configuration file that puts OpenVPN into "server" mode, authenticates a single client by checking its RSA certificate for a valid CA signature, transparently generates dynamic session keys, establishes the tunnel, and then pushes settings back to the client that give the client a route back to the home network.

And, let's even force the client to use the tunnel (and therefore the home network) as its default route back to the outside world, which is a potent protection against DNS spoofing and other attacks you otherwise might be vulnerable to when using an untrusted network.

Configuring OpenVPN consists of creating a configuration file for each OpenVPN listener you want to run and creating any additional files (certificates and so forth) referenced by that file. Prior to OpenVPN 2.0, you needed one listener per tunnel. If ten people needed to connect to your OpenVPN server concurrently, they'd each connect to a different UDP or TCP port on the server.

OpenVPN as of version 2.0, however, is multithreaded, and running in "server" mode, multiple clients can connect to the same TCP or UDP port using the same tunnel profile (that is, you can't have some users authenticate via TLS certificates and other users authenticate via shared secret on the same port). You still need to designate different ports for different tunnel configurations.

Even though the example scenario involves only one client, which would be amply served by a "peer-to-peer" OpenVPN tunnel, it really isn't any more complicated to use a "server mode" tunnel instead (that, again, you can use to serve multiple clients after changing only one line). As far as I can tell, using server mode for a single user doesn't seem to have any noticeable performance cost either. In my testing, even the relatively computationally intensive RSA public

What about Static Keys?

Conspicuously absent from my OpenVPN examples are static keys (also known as pre-shared secret keys), which provide a method for authenticating OpenVPN tunnels that is, arguably, simpler to use than the RSA certificates described herein. Why?

An OpenVPN shared secret takes the form of a small file containing cryptographically generated random data that is highly, highly infeasible for an external attacker to guess via some sort of dictionary or brute-force attack. However, unlike WPA or IPsec shared secrets, an OpenVPN shared secret is used as a de facto session encryption key for every instance of every tunnel that uses it; it is *not* used to generate other, temporary, session keys that change over time.

For this reason, if attackers were to collect encrypted OpenVPN packets from, say, four different OpenVPN sessions between the same two endpoints and were to later somehow obtain a copy of that tunnel's shared secret file, they would be able to decrypt

all packets from *all four* captured sessions.

In contrast, if you instead authenticate your OpenVPN tunnel with RSA certificates, OpenVPN uses the certificates dynamically to re-key the tunnel periodically—not just every time the tunnel is established, but even during the course of a single tunnel session. Furthermore, even if attackers somehow obtain *both* RSA certificates and keys used to key that tunnel, they can't easily decrypt *any* prior captured OpenVPN session (which would involve reconstructing the *entire* key negotiation process for every session key used in a given session), although they easily can initiate new sessions themselves.

So in summary, although it is a modest hassle to set up a CA and generate RSA certificates, in my opinion, using RSA certificates provides an increase in security that is much more significant than the simplicity of using shared secrets.

Listing 1. Server's server.ovpn File

```

port 1194
proto udp
dev tun

ca 2.0/keys/ca.crt
cert 2.0/keys/server.crt
key 2.0/keys/server.key # This file should be kept secret

dh 2.0/keys/dh1024.pem

tls-auth 2.0/keys/ta.key 0

server 10.31.33.0 255.255.255.0
ifconfig-pool-persist ipp.txt

push "redirect-gateway def1 bypass-dhcp"

keepalive 10 120

cipher BF-CBC      # Blowfish (default)

comp-lzo
max-clients 2

user nobody
group nogroup
persist-key
persist-tun

status openvpn-status.log

verb 3
mute 20

```

key routines involved in establishing my tunnels completed very rapidly.

Listing 1 shows a tunnel configuration file, `server.ovpn`, for our home network's OpenVPN server.

Let's walk through Listing 1's settings. `port` obviously designates this listener's port number. In this case, it's OpenVPN's default of 1194. `proto` specifies that this tunnel will use fast, connectionless UDP packets rather than slower but more reliable TCP packets (the other allowable value being `tcp`). Note that OpenVPN uses information in its UDP data payloads to maintain tunnel state. Even though UDP is by definition a "stateless" protocol, the OpenVPN process on either end of an OpenVPN UDP tunnel can detect dropped packets and request the other side to retransmit them.

`dev` sets the listener to use the Linux kernel's `/dev/tun` (tun) special device rather than `/dev/tap`

(which would be specified by `tap`). Whereas the `tap` device is used to encapsulate entire Ethernet frames, the `tun` device encapsulates only IPv4 or IPv6 packets. In other words, the `tap` device tunnels all network traffic regardless of protocol (IPX/SPX, Appletalk, Netbios, IP). For this example, let's stick to the `tun` device; this will be an IP-only tunnel.

Next, there is the RSA certificate information: `ca`, `cert` and `key`, which specify the respective paths of a CA certificate, the server's certificate and the server's private key. The CA certificate is used to validate client certificates. If the certificate presented by a client contains a valid signature corresponding to the CA certificate, tunnel authentication succeeds. The server key is used during this authentication transaction and also, subsequently, during key negotiation transactions.

Note that certificate files are public information and as such don't need highly restrictive file permissions, but key files must be kept secret and should be root-readable only. Never transmit any key file over any untrusted channel! Note also that all paths in this configuration file are relative to the configuration file itself. If the file resides in `/etc/openvpn`, then the `ca` path `2.0/keys/ca.cert` actually expands to `/etc/openvpn/2.0/keys/ca.cert`.

`dh` specifies a file containing seed data for the Diffie-Hellman session-key negotiation protocol. This data isn't particularly secret. `tls-auth`, however, specifies the path to a secret key file used by both server and client daemons to add an extra level of validation to all tunnel packets (technically, "authentication", as in "message authentication" rather than "user authentication"). Although not necessary for the tunnel to work, I like `tls-auth` because it helps prevent replay attacks.

Before I go any further explaining Listing 1, let's generate the files I just described. The first three, `ca`, `cert` and `key`, require a PKI, but like I mentioned, OpenVPN includes scripts to simplify PKI tasks. On my Ubuntu systems, these scripts are located in `/usr/share/doc/openvpn/examples/easy-rsa/2.0`. Step one in creating a PKI is to copy these files to `/etc/openvpn`, like so:

```

bash-$ cd /usr/share/doc/openvpn/examples/easy-rsa
bash-$ su
bash-# cp -r 2.0 /etc/openvpn

```

Notice that contrary to preferred Ubuntu/Debian practice, I "su-ed" to root. This is needed to create a PKI, a necessarily privileged set of tasks.

Step two is to customize the file `vars`, which specifies CA variables. First, change your working directory to the copy of `easy-rsa` you just created, and open the file `vars` in vi:

```
bash-# cd /etc/openvpn/2.0  
bash-# vi vars
```

Here are the lines I changed in my own vars file:

```
export KEY_COUNTRY="US"  
export KEY_PROVINCE="MN"  
export KEY_CITY="Saint Paul"  
export KEY_ORG="Wiremonkeys"  
export KEY_EMAIL="mick@wiremonkeys.org"
```

Next, initialize your new PKI environment:

```
bash-# source ./vars  
bash-# ./clean-all  
bash-# ./build-dh
```

And now, finally, you can create some certificates. First, of course, you need the CA certificate and key itself, which will be necessary to sign subsequent keys:

```
bash-# ./pkictool --initca
```

The output of that command consists of the files keys/ca.crt and keys/ca.key. By the way, if you want pkictool's output files to be written somewhere besides the local directory keys, you can specify a different directory in the file vars via the variable KEY_DIR.

Next, generate your OpenVPN server certificate:

```
bash-# ./pkictool --server server
```

This results in two files: keys/server.crt and keys/server.key. There's nothing magical about the last parameter in the above command, which is simply the name of the server certificate; to name it chuck (resulting in keys/chuck.crt and keys/chuck.key), you'd use ./pkictool --server chuck.

Last comes the client certificate. Unlike the server certificate, whose key may need to be used by some unattended daemon process, we expect client certificates to be used by human beings. Therefore, let's create a client certificate with a password-protected (encrypted) key file, like so:



Powerful: Rhino

Rhino M6500/E6500

- Dell Precision M6500 w/ Core i7 Quad (8 core)
- Dell Latitude E6500 w/ 2.2-3.0 GHz Core 2 Duo
- Up to 17" WUXGA LCD w/ X@1920x1200
- NVidia Quadro FX 3800M
- 80-500 GB hard drive
- Up to 16 GB RAM (1333 MHz)
- DVD±RW or Blu-ray
- 802.11a/g/n
- Starts at \$1240

- High performance NVidia 3-D on a WUXGA RGB/LED
- High performance Core i7 Quad CPUs, 16 GB RAM
- Ultimate configurability — choose your laptop's features
- One year Linux tech support — phone and email
- Three year manufacturer's on-site warranty
- Choice of pre-installed Linux distribution:



Tablet: Raven

Raven X200 Tablet

- ThinkPad X200 tablet by Lenovo
- 12.1" WXGA w/ X@1280x800
- 1.2-1.86 GHz Core 2 Duo
- Up to 8 GB RAM
- 80-500 GB hard drive / 256 GB SSD
- Pen/stylus input to screen
- Dynamic screen rotation
- Starts at \$2080



Rugged: Tarantula

Tarantula CF-30

- Panasonic Toughbook CF-30
- Fully rugged MIL-SPEC-810F tested: drops, dust, moisture & more
- 13.3" XGA TouchScreen
- 1.6 GHz Core 2 Duo
- Up to 8 GB RAM
- 80-500 GB hard drive
- Call for quote

EmperorLinux
...where Linux & laptops converge

www.EmperorLinux.com

1-888-651-6686

Model specifications and availability may vary.



```
bash-# ./pkictool --pass minion
```

You'll be prompted twice for the key file's passphrase, which will be used to encrypt the file keys/minion.key (keys/minion.crt also will be created by not password-protected). If minion.key falls into the wrong hands, it won't be usable unless the thief also knows its password. However, this also means that every time you use this certificate, you'll be prompted for the key file's password, which I think is a reasonable expectation for VPN clients.

Now that you've got a working PKI set up, all you'll need to do to generate additional client certificates is repeat that last command, but with different certificate names, for example
`./pkictool --pass minion102.`

Warning: be careful about how you transmit client certificates and keys to end users! Unencrypted e-mail is a poor choice for this task. You should instead use scp, sftp or some other secure file-transfer protocol, or even transport

So in summary, although it is a modest hassle to set up a CA and generate RSA certificates, in my opinion, using RSA certificates provides an increase in security that is much more significant than the simplicity of using shared secrets.

them manually with a USB drive. Once the client certificate and key have been copied where they need to go (for example, /etc/openvpn/keys on the client system), make sure the key file is root-readable only! Erase any temporary copies of this file you may have made in the process of transporting it—for example, on a USB drive.

The OpenVPN server does not need local copies of client certificate or key files, though it may make sense to leave the "original" copies of these in the server's /etc/openvpn/2.0/keys directory (in my examples) in the event of users losing theirs due, for example, to a hard drive crash.

In the interest of full disclosure, I should note that contrary to my examples, it is a PKI best practice *not* to run a PKI (CA) on any system that actually *uses* the PKI's certificates. Technically, I should be telling you to use a dedicated, non-networked system for this purpose! Personally, I think if *all* you use this particular PKI for your OpenVPN RSA certificates, if your OpenVPN server is configured securely overall, and you keep all

key files root-readable only, you probably don't need to go that far.

Okay, we've got a working PKI and some certificates. That may have been a lengthy explanation, but in my opinion, the process isn't too difficult or unwieldy. It probably will take you less time to *do it* than it just took you to *read about it*.

You've got two more files to generate before continuing working down server.ovpn. To generate your Diffie-Hellman seed file (still working as root within the directory /etc/openvpn/2.0), use this command:

```
bash-# openssl dhparam -out keys/dh1024.pem 1024
```

And, last of all the supplemental files, generate that TLS-authentication file, like so:

```
bash-# openvpn --genkey --secret 2.0/keys/ta.key
```

Conclusion

At this point, I've got good news and bad news. The good news is, you've made it through the most complicated part of OpenVPN configuration: creating a PKI and generating certificates and related files. The bad news is, you've also reached the end of this month's column!

If you can't wait until next time to *use* these certificates, to get OpenVPN running, you probably can figure out how to do so yourself. See the openvpn(8) man page and the sample configuration files server.conf.gz and client.conf under /usr/share/doc/openvpn/examples/sample-config-files, upon which my examples are based. Good luck!■

Mick Bauer (darth.elmo@wiremonkeys.org) is Network Security Architect for one of the US's largest banks. He is the author of the O'Reilly book *Linux Server Security*, 2nd edition (formerly called *Building Secure Servers With Linux*), an occasional presenter at information security conferences and composer of the "Network Engineering Polka".

Resources

Official OpenVPN Home Page:

www.openvpn.net

Ubuntu Community OpenVPN Page:

<https://help.ubuntu.com/community/OpenVPN>

"Linux VPN Technologies" by Mick Bauer, *LJ*, January 2005: www.linuxjournal.com/article/7881

Charlie Hosner's "SSL VPNs and OpenVPN: A lot of lies and a shred of truth": www.linux.com/archive/feature/48330



7th USENIX Symposium on Networked Systems Design and Implementation

April 28–30, 2010 • San Jose, CA

Sponsored by USENIX in cooperation with ACM SIGCOMM and ACM SIGOPS

NSDI '10 will focus on the design principles of large-scale networked and distributed systems in a 3-day technical program including topics such as:

- Cloud services
- Datacenter and wireless networks
- And more!
- Web browsers and servers
- Malware

Join researchers from across the networking and systems community—including computer networking, distributed systems, and operating systems—in fostering cross-disciplinary approaches and addressing shared research challenges.

USENIX has new ways for you to save. Check out the discounts available!

www.usenix.org/nsdi10/discounts

A dark green hexagonal badge with the text "Early Bird Discount" in white, tilted at an angle.

Register by Monday, April 5, and save!

www.usenix.org/nsdi10/lja

Don't Miss the Co-located Workshops!

All workshops will take place on April 27, 2010.

- 3rd USENIX Workshop on Large-Scale Exploits and Emergent Threats (LEET '10)
- 2010 Internet Network Management Workshop/Workshop on Research on Enterprise Networking (INM/WREN '10)
- 9th International Workshop on Peer-to-Peer Systems (IPTPS '10)

www.usenix.org/nsdi10/workshops

USENIX



www.usenix.org/facebook



www.twitter.com/usenix

www.usenix.org



KYLE RANKIN

Linux Troubleshooting, Part I: High Load

What do you do when you get an alert that your system load is high? Tracking down the cause of high load just takes some time, some experience and a few Linux tools.

This column is the first in a series of columns dedicated to one of my favorite subjects: troubleshooting. I'm a systems administrator during the day, and although I enjoy many aspects of my job, it's hard to beat the adrenaline rush of tracking down a complex server problem when downtime is being measured in dollars. Although it's true that there are about as many different reasons for downtime as there are Linux text editors, and just as many approaches to troubleshooting, over the years, I've found I perform the same sorts of steps to isolate a problem. Because my column is generally aimed more at tips and tricks and less on philosophy and design, I'm not going to talk much about overall approaches to problem solving. Instead, in this series I describe some general classes of problems you might find on a Linux system, and then I discuss how to use common tools, most of which probably are already on your system, to isolate and resolve each class of problem.

For this first column, I start with one of the most common problems you will run into on a Linux system. No, it's not getting printing to work. I'm talking about a sluggish server that might have high load. Before I explain how to diagnose and fix high load though, let's take a step back and discuss what load means on a Linux machine and how to know when it's high.

Uptime and Load

When administrators mention high load, generally they are talking about the *load average*. When I diagnose why a server is slow, the first command I run when I log in to the system is uptime:

```
$ uptime
18:30:35 up 365 days, 5:29, 2 users, load average: 1.37, 10.15, 8.10
```

As you can see, it's my server's uptime birthday today. You also can see that my load average is 1.37, 10.15, 8.10. These numbers represent my average system load over the last 1, 5 and 15 minutes, respectively. Technically speaking, the load average represents the average number of processes

that have to wait for CPU time during the last 1, 5 or 15 minutes. For instance, if I have a current load of 0, the system is completely idle. If I have a load of 1, the CPU is busy enough that one process is having to wait for CPU time. If I do have a load of 1 and then spawn another process that normally would tie up a CPU, my load should go to 2. With a load average, the system will give you a good idea of how consistently busy it has been over the past 1, 5 and 10 minutes.

Another important thing to keep in mind when you look at a load average is that it isn't normalized according to the number of CPUs on your system. Generally speaking, a consistent load of 1 means one CPU on the system is tied up. In simplified terms, this means that a single-CPU system with a load of 1 is roughly as busy as a four-CPU system with a load of 4. So in my above example, let's assume that I have a single-CPU system. If I were to log in and see the above load average, I'd probably assume that the server had pretty high load (8.10) during the last 15 minutes that spiked around 5 minutes ago (10.15), but recently, at least during the last 1 minute, the load has dropped significantly. If I saw this, I might even assume that the real cause of the load has subsided. On the other hand, if the load averages were 20.68, 5.01, 1.03, I would conclude that the high load had likely started in the last 5 minutes and was getting worse.

How High Is High?

After you understand what load average means, the next logical question is "What load average is good and what is bad?" The answer to that is "It depends." You see, a lot of different things can cause load to be high, each of which affects performance differently. One server might have a load of 50 and still be pretty responsive, while another server might have a load of 10 and take forever to log in to. I've had servers with load averages in the hundreds that were certainly slow, but didn't crash, and I had one server that consistently had a load of 50 that was still pretty

responsive and stayed up for years.

What really matters when you troubleshoot a system with high load is *why* the load is high. When you start to diagnose high load, you find that most load seems to fall into three categories: CPU-bound load, load caused by out of memory issues and I/O-bound load. I explain each of these categories in detail below and how to use tools like top and iostat to isolate the root cause.

top

If the first tool I use when I log in to a sluggish system is uptime, the second tool I use is top. The great thing about top is that it's available for all major Linux systems, and it provides a lot of useful information in a single screen. top is a quite complex tool with many options that could warrant its own article. For this column, I stick to how to interpret its output to diagnose high load.

To use top, simply type top on the command line. By default, top will run in interactive mode and update its output every few seconds. Listing 1 shows sample top output from a terminal.

As you can see, there's a lot of information in only a few lines. The first line mirrors the information you would get from the uptime command and will update every few seconds with the latest load averages. In this case, you can see my system is busy, but not what I would call heavily loaded. All the same, this output breaks down well into our different load categories. When I troubleshoot a sluggish system, I generally will rule out CPU-bound load, then RAM issues, then finally I/O issues in that order, so let's start with CPU-bound load.

CPU-Bound Load

CPU-bound load is load caused when you have too many CPU-intensive processes running at once. Because each process needs CPU resources, they all must wait their turn. To check whether load is CPU-bound, check the CPU line in the top output:

Cpu(s): 11.4%us, 29.6%sy, 0.0%ni, 58.3%id, .7%wa, 0.0%hi, 0.0%si, 0.0%st

Each of these percentages are a percentage of the CPU time tied up doing a particular task. Again, you could spend an entire column on all of the output from top, so here's a few of these values and how to read them:

- us: user CPU time. More often than not, when you have CPU-bound load, it's due to a process run by a user on the system, such as Apache, MySQL or maybe a shell script. If this percentage is high, a user process such as those is a likely cause of the load.

Listing 1. Sample top Output

```
top - 14:08:25 up 38 days, 8:02, 1 user, load average: 1.70, 1.77, 1.68
Tasks: 107 total, 3 running, 104 sleeping, 0 stopped, 0 zombie
Cpu(s): 11.4%us, 29.6%sy, 0.0%ni, 58.3%id, .7%wa, 0.0%hi, 0.0%si, 0.0%st
Mem: 1024176k total, 997408k used, 26768k free, 85520k buffers
Swap: 1004052k total, 4360k used, 999692k free, 286040k cached
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
9463	mysql	16	0	686m	111m	3328	S	53	5.5	569:17.64	mysqld
18749	nagios	16	0	140m	134m	1868	S	12	6.6	1345:01	nagios2db_status
24636	nagios	17	0	34660	10m	712	S	8	0.5	1195:15	nagios
22442	nagios	24	0	6048	2024	1452	S	8	0.1	0:00.04	check_time.pl

Listing 2. Current Processes Example

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
9463	mysql	16	0	686m	111m	3328	S	53	5.5	569:17.64	mysqld
18749	nagios	16	0	140m	134m	1868	S	12	6.6	1345:01	nagios2db_status
24636	nagios	17	0	34660	10m	712	S	8	0.5	1195:15	nagios
22442	nagios	24	0	6048	2024	1452	S	8	0.1	0:00.04	check_time.pl

- sy: system CPU time. The system CPU time is the percentage of the CPU tied up by kernel and other system processes. CPU-bound load should manifest either as a high percentage of user or high system CPU time.
- id: CPU idle time. This is the percentage of the time that the CPU spends idle. The higher the number here the better! In fact, if you see really high CPU idle time, it's a good indication that any high load is not CPU-bound.
- wa: I/O wait. The I/O wait value tells the percentage of time the CPU is spending waiting on I/O (typically disk I/O). If you have high load and this value is high, it's likely the load is not CPU-bound but is due to either RAM issues or high disk I/O.

Track Down CPU-Bound Load

If you do see a high percentage in the user or system columns, there's a good chance your load is CPU-bound. To track down the root cause, skip down a few lines to where top displays a list of current processes running on the system. By default, top will sort these based on the percentage of CPU used with the processes using the most on top (Listing 2).

The %CPU column tells you just how much CPU each process is taking up. In this case, you can see that MySQL is taking up 53% of my CPU.

Listing 3. Processes Sorted by RAM

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
18749	nagios	16	0	140m	134m	1868	S	12	6.6	1345:01	nagios2db_status
9463	mysql	16	0	686m	111m	3328	S	53	5.5	569:17.64	mysqld
24636	nagios	17	0	34660	10m	712	S	8	0.5	1195:15	nagios
22442	nagios	24	0	6048	2024	1452	S	8	0.1	0:00.04	check_time.pl

As you look at this output during CPU-bound load, you probably will see one of two things: either you will have a single process tying up 99% of your CPU, or you will see a number of smaller processes all fighting for a percentage of CPU time. In either case, it's relatively simple to see the processes that are causing the problem. There's one final note I want to add on CPU-bound load: I've seen systems get incredibly high load simply because a multithreaded program spawned a huge number of threads on a system without many CPUs. If you spawn 20 threads on a single-CPU system, you might see a high load average, even though there are no particular processes that seem to tie up CPU time.

Out of RAM Issues

The next cause for high load is a system that has run out of available RAM and has started to go into swap. Because swap space is usually on a hard drive that is much slower than RAM, when you use up available RAM and go into swap, each process slows down dramatically as the disk gets used. Usually this causes a downward spiral as processes that have been swapped run slower, take longer to respond and cause more processes to stack up until the system either runs out of RAM or slows down to an absolute crawl. What's tricky about swap issues is that because they hit the disk so hard, it's easy to misdiagnose them as I/O-bound load. After all, if your disk is being used as RAM, any processes that actually want to access files on the disk are going to have to wait in line. So, if I see high I/O wait in the CPU row in top, I check RAM next and rule it out before I troubleshoot any other I/O issues.

When I want to diagnose out of memory issues, the first place I look is the next couple of lines in the top output:

```
Mem: 1024176k total, 997408k used, 26768k free, 85520k buffers
Swap: 1004052k total, 4360k used, 999692k free, 286040k cached
```

These lines tell you the total amount of RAM and swap along with how much is used and free; however, look carefully, as these numbers can be misleading. I've seen many new and even

experienced administrators who would look at the above output and conclude the system was almost out of RAM because there was only 26768k free. Although that does show how much RAM is currently unused, it doesn't tell the full story.

The Linux File Cache

When you access a file and the Linux kernel loads it into RAM, the kernel doesn't necessarily unload the file when you no longer need it. If there is enough free RAM available, the kernel tries to cache as many files as it can into RAM. That way, if you access the file a second time, the kernel can retrieve it from RAM instead of the disk and give much better performance. As a system stays running, you will find the free RAM actually will appear to get rather small. If a process needs more RAM though, the kernel simply uses some of its file cache. In fact, I see a lot of the over-clocking crowd who want to improve performance and create a ramdisk to store their files. What they don't realize is that more often than not, if they just let the kernel do the work for them, they'd probably see much better results and make more efficient use of their RAM.

To get a more accurate amount of free RAM, you need to combine the values from the free column with the cached column. In my example, I would have 26768k + 286040k, or over 300MB of free RAM. In this case, I could safely assume my system was not experiencing an out of RAM issue. Of course, even a system that has very little free RAM may not have gone into swap. That's why you also must check the Swap: line and see if a high proportion of your swap is being used.

Track Down High RAM Usage

If you do find you are low on free RAM, go back to the same process output from top, only this time, look in the %MEM column. By default, top will sort by the %CPU column, so simply type M and it will re-sort to show you which processes are using the highest percentage of RAM. In the output in Listing 3, I sorted the same processes by RAM, and you can see that the nagios2db_status process is using the most at 6.6%.

I/O-Bound Load

I/O-bound load can be tricky to track down sometimes. As I mentioned earlier, if your system is swapping, it can make the load appear to be I/O-bound. Once you rule out swapping though, if you do have a high I/O wait, the next step is to attempt to track down which disk and partition is getting the bulk of the I/O traffic. To do this, you need a tool like iostat.

The iostat tool, like top, is a complicated and

full-featured tool that could fill up its own article. Unlike top, although it should be available for your distribution, it may not be installed on your system by default, so you need to track down which package provides it. Under Red Hat and Debian-based systems, you can get it in the sysstat package. Once it's installed, simply run iostat with no arguments to get a good overall view of your disk I/O statistics:

```
Linux 2.6.24-19-server (hostname) 01/31/2009

avg-cpu: %user   %nice %system %iowait  %steal   %idle
          5.73    0.07   2.03    0.53    0.00   91.64

Device: tps Blk_read/s Blk_wrtn/s Blk_read Blk_wrtn
sda     9.82    417.96     27.53 30227262 1990625
sda1    6.55    219.10      7.12 15845129 515216
sda2    0.04      0.74      3.31  53506 239328
sda3    3.24    198.12     17.09 14328323 1236081
```

Like with top, iostat gives you the CPU percentage output. Below that, it provides a breakdown of each drive and partition on your system and statistics for each:

- tps: transactions per second.
- Blk_read/s: blocks read per second.
- Blk_wrtn/s: blocks written per second.
- Blk_read: total blocks read.
- Blk_wrtn: total blocks written.

By looking at these different values and comparing them to each other, ideally you will be able to find out first, which partition (or partitions) is getting the bulk of the I/O traffic, and second, whether the majority of that traffic is reads (Blk_read/s) or writes (Blk_wrtn/s). As I said, tracking down the cause of I/O issues can be tricky, but hopefully, those values will help you isolate what processes might be causing the load.

For instance, if you have an I/O-bound load and you suspect that your remote backup job might be the culprit, compare the read and write statistics. Because you know that a remote backup job is primarily going to read from your disk, if you see that the majority of the disk I/O is writes, you reasonably can assume it's not from the backup job. If, on the other hand, you do see a heavy amount of read I/O on a particular partition, you might run the lsof command and grep for that backup process and see whether it does in fact have some open file handles on that partition.

Listing 4. Example iotop Tool Output

```
Total DISK READ: 189.52 K/s | Total DISK WRITE: 0.00 B/s
TID PRIO USER    DISK READ  DISK WRITE  SWAPIN    IO>      COMMAND
8169 be/4 root    189.52 K/s    0.00 B/s  0.00 %  0.00 % rsync --server --se
4243 be/4 kyle    0.00 B/s    3.79 K/s  0.00 %  0.00 % cli /usr/lib/gnome-
4244 be/4 kyle    0.00 B/s    3.79 K/s  0.00 %  0.00 % cli /usr/lib/gnome-
1 be/4 root    0.00 B/s    0.00 B/s  0.00 %  0.00 % init
```

As you can see, tracking down I/O issues with iostat is not straightforward. Even with no arguments, it can take some time and experience to make sense of the output. That said, iostat does have a number of arguments you can use to get more information about different types of I/O, including modes to find details about NFS shares. Check out the man page for iostat if you want to know more.

Up until recently, tools like iostat were about the limit systems administrators had in their toolboxes for tracking down I/O issues, but due to recent developments in the kernel, it has become easier to find the causes of I/O on a per-process level. If you have a relatively new system, check out the iotop tool. Like with iostat, it may not be installed by default, but as the name implies, it essentially acts like top, only for disk I/O. In Listing 4, you can see that an rsync process on this machine is using the most I/O (in this case, read I/O).

Once You Track Down the Culprit

How you deal with these load-causing processes is up to you and depends on a lot of factors. In some cases, you might have a script that has gone out of control and is something you can easily kill. In other situations, such as in the case of a database process, it might not be safe simply to kill the process, because it could leave corrupted data behind. Plus, it could just be that your service is running out of capacity, and the real solution is either to add more resources to your current server or add more servers to share the load. It might even be load from a one-time job that is running on the machine and shouldn't impact load in the future, so you just can let the process complete. Because so many different things can cause processes to tie up server resources, it's hard to list them all here, but hopefully, being able to identify the causes of your high load will put you on the right track the next time you get an alert that a machine is slow.■

Kyle Rankin is a Systems Architect in the San Francisco Bay Area and the author of a number of books, including *The Official Ubuntu Server Book*, *Knoppix Hacks* and *Ubuntu Hacks*. He is currently the president of the North Bay Linux Users' Group.



DIRK ELMENDORF

Installation Toolkit

Having the right tools for the job.

I don't know if it's because I have a column in *Linux Journal* or because of the release of Karmic Koala, but either way, I seem to be installing Linux a lot lately. If there is one lesson I have learned from time spent on other hobbies, it's that things are always easier if you have the right tools handy. In this case, that means taking a step back from the install process to see what tools might help.

ISOs, ISOs Everywhere

ISO 9660 is the standard for storing images of CD-ROM filesystems. Although I vaguely remember trying to install Slackware from floppy disks, and once in a while, I'll see a DVD-ROM ISO for a distro, most stick to the 650MB image. As a result, I have lots of them. These days, having the right ISO is useful for a fresh desktop install (once you finish burning it), or it can be used directly if you are creating a virtual machine. This is pretty much the entry level of installation tools. My only piece of advice is make sure that when you burn an ISO, burn the contents and not the file. If, when you mount the disc and see `ubuntu-9.10-desktop-amd64.iso` on the disc, you missed a step.

New-School—Small, Portable and Green?

Another option for installation media is the thumbdrive. Prices have dropped, capacities have skyrocketed, motherboard support has expanded, and tools have improved. All that adds up to making this a really great option.

Ubuntu ships with a tool called `usb-creator`. It's a very straightforward tool for converting your thumbdrive into a bootable utility. However, I prefer UNetbootin (unetbootin.sourceforge.net). This handy tool does the same thing, but it adds a helpful hand by offering to auto-download a variety of Linux distributions.

Both tools make it incredibly easy to make your thumbdrive bootable. One thing to keep in mind: in most cases, you need only 650MB, but when I wrote this, it was cheaper on Amazon to buy 2GB thumbdrives than 1GB thumbdrives. Manufacturers constantly are chasing the biggest capacities, which means the sweet spot in pricing often is just behind this—much like hard drives (have you priced an 80GB hard drive lately?). I ended up buying a three-pack of 2GB thumbdrives just for this purpose. They are installed with the current x86 version of Ubuntu, SystemRescueCD and Clonezilla. I am contemplating adding on the x64 version of Ubuntu (as I seem to be

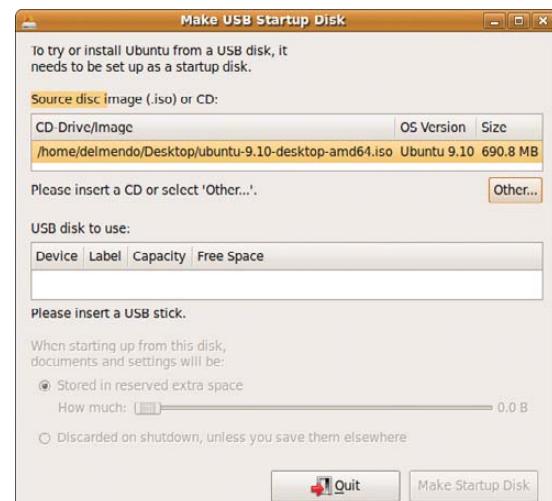
Figure 1. `usb-creator`

Figure 2. UNetbootin

choosing that more often) and Darik's Boot and Nuke (comes in handy as you decommission equipment). The nice thing about the thumbdrive form factor is that I can just keep them on a key chain in my laptop bag. I don't have to worry about scratching them, and when updates come out, I can re-install over them.

Old-School Industrial

CDs and thumbdrives work great, but if you are going to be doing a lot of installing, there is another tool to add to your arsenal—PXE booting. PXE (pronounced "pixie") stands for Preboot eXecution Environment. I've used it a lot at hosting companies I've worked at, but I never have gotten around to setting it up on my home network.

PXE allows you to boot a computer off your

network. In this case, I am going to set it up so I can boot the installation environment and then switch back to booting locally. More work is involved if you want to make thin clients (meaning, if you want a computer to boot off the network and not have any local storage).

In order for this to work, you need a server on your network to host the PXE, DHCP and other services required. The target computer has to be connected to the same network, and the BIOS of that computer must support PXE (or as I learned later, you have a gPXE ISO and a CD drive). The good news is that most modern motherboards support PXE (usually labeled as boot off of LAN). You may be able to tell the computer to offer you a boot menu on startup. This allows you to boot off the network one time without forcing you to modify your BIOS.

I sat down to start the process. I have a file server (keg) that will handle all the PXE services. PXE also expects DHCP. Many of the guides I found on-line assume the PXE server also will handle DHCP. In my case, all the networking is handled by my main DD-WRT router (co2). That means I will have to modify it as well to make things work.

Configuring co2

The DHCP server needs to offer up a PXE boot image for this to work. In my case, that meant I needed to update my DD-WRT configuration. I decided to change it first. I went to the Services tab on my router and added `dhcp-boot=pxelinux.0,keg,192.168.210.254` to the box for Additional DNSMasq options. That is the PXE image, the name of my file server and my file server's IP. If you are running a DHCP server as well, you need to add something similar, depending on what DHCP software you are running.

Configuring Keg

Now the rest of the configuration happens on my file server. I installed a tftp server to serve up the PXE boot image:

```
sudo apt-get install tftpd-hpa
```

This creates a directory (`/var/lib/tftpboot`) and a config file (`/etc/default/tftpd-hpa`). Edit the config file by changing `RUN_DAEMON="no"` to `RUN_DAEMON="yes"`. Then:

```
sudo /etc/init.d/tftpd-hpa restart
```

The next step is to get an installer:

```
sudo cd /var/lib/tftpboot
sudo wget -r -nH --cut-dirs=8 -np ftp://archive.ubuntu.com/
  ubuntu/dists/karmic/main/installer-i386/current/images/netboot/
```

Now you can boot up your machine and have it

load the standard Karmic installer. Part of the fun and flexibility of having a PXE server is that you can choose what to install on the fly. I'm going to start with adding on the x64 installer for Karmic. First, I needed to clean up the directory structure a bit. I removed everything (except the `pxelinux.0` and `pxelinux.cfg`, which both turned out to be symlinks to files I deleted):

```
sudo mkdir -p ubuntu/karmic/i386
sudo mkdir ubuntu/karmic/amd64
sudo wget -r -nH --cut-dirs=8 -np ftp://archive.ubuntu.com/
  ubuntu/dists/karmic/main/installer-i386/current/images/netboot/
    -P /var/lib/tftpboot/ubuntu/karmic/i386
sudo wget -r -nH --cut-dirs=8 -np ftp://archive.ubuntu.com/
  ubuntu/dists/karmic/main/installer-amd64/current/images/netboot/
    -P /var/lib/tftpboot/ubuntu/karmic/amd64
sudo cp ubuntu/karmic/i386/ubuntu-installer/i386/pxelinux.0
  /var/lib/tftpboot/
sudo mkdir /var/lib/tftpboot/pxelinux.cfg
```

Now, set up a menu to choose between the two installers. Create a `/var/lib/tftpboot/boot.txt` file with:

- Install Options -

```
karmic_i386_install
karmic_i386_expert
karmic_amd64_install
karmic_amd64_expert
```

Create a `/var/lib/tftpboot/pxelinux.cfg/default` file with:

DISPLAY boot.txt

```
LABEL karmic_i386_install
  kernel ubuntu/karmic/i386/ubuntu-installer/i386/linux
  append vga=normal initrd=ubuntu/karmic/i386/
    -ubuntu-installer/i386/initrd.gz --
LABEL karmic_i386_expert
  kernel ubuntu/karmic/i386/ubuntu-installer/i386/linux
  append priority=low vga=normal initrd=ubuntu/karmic/
    -i386/ubuntu-installer/i386/initrd.gz --
LABEL karmic_amd64_install
  kernel ubuntu/karmic/amd64/ubuntu-installer/amd64/linux
  append vga=normal initrd=ubuntu/karmic/amd64/
    -ubuntu-installer/amd64/initrd.gz --
LABEL karmic_amd64_expert
  kernel ubuntu/karmic/amd64/ubuntu-installer/amd64/linux
  append priority=low vga=normal initrd=ubuntu/karmic/
    -amd64/ubuntu-installer/amd64/initrd.gz --
```

From here on, you just need to download a distribution you want to support and add it to the menu. If you are the graphical type, you even can modify the system to allow menus and pretty icons.

Wireless PXE?

PXE is great, because it allows you to boot up and choose the installer you need. The problem is that it assumes you are on the physical network. In the case of my wireless computers (Netbooks, laptops and that one computer that sits out in my brew house), I couldn't use PXE. The main workaround is to get a wireless bridge so the device can boot over a physical port on the bridge, and let the bridge worry about the Wi-Fi. The question of how to do that has floated around so long, I am not holding out hope a solution will show up any time soon.

Alternative for the Lazy

Let's say you want some of the features of PXE, but you don't want to bother installing a server. It turns out you have two different options for doing pretty much the same thing over the Internet. You can use **boot.kernel.org** or **www.netboot.me**. Both are using gPXE, which allows you to do the same things I did, but you don't need to have any infrastructure (although you may want to have a lot of bandwidth, depending on how much installing you do).

Caching Is Good

This brings me to another recommendation—if you are going to be doing a lot of installing or if you have a large collection of Ubuntu boxes, make sure you install an apt cache. There are several solutions for this. One is just to mirror an Ubuntu server to a central server on your network. That seems like a lot of disk space and bandwidth, but it could be worthwhile if you are going to be installing a lot of machines that all use a lot of packages.

In my case, I went with a slightly less resource-intensive solution—an apt caching system. There are a number from which to choose. I originally was leaning toward approx (**git.debian.org/?p=pkg-ocaml-maint/packages/approx.git**), because it is supposed to be stable and very easy to use. The problem is that it will allow only one client to update at a time. That's not the end of the world, but I would prefer not to have the limitation.

As a result, I switched to apt-cacher-ng. It has an added benefit of a Web page that shows you the status and how much you have cached, which is very useful for troubleshooting. I had to download the deb from its site (**www.unix-ag.uni-kl.de/~bloch/acng**). I installed it on keg since it handles these kind of network services:

```
cd /tmp
wget
http://ftp.debian.org/debian/pool/main/a/apt-cacher-ng/
apt-cacher-ng_0.4.2-1_i386.deb
sudo dpkg -i apt-cacher-ng_0.4.2-1_i386.deb
```

Unfortunately, keg is running jaunty (I still have more Linux to update). So I actually had to install apt-cacher-ng 0.4 because of a libc conflict.

In the past, when I have installed this kind of software, I always ended up updating the /etc/apt/sources.list file to point to this server. As a result, I often forgot to point new sources at the cache. It turns out that apt provides an easy way to override the location from which you are downloading your debs:

```
sudo echo 'Acquire::http { Proxy "http://192.168.210.254:3142"; };' >
/etc/apt/apt.conf.d/001apt-catcher
```

In this case, I put keg's IP in the configuration, mostly so it would be easy to copy that to other computers on my network. apt-cacher-ng also provides a Web interface to see statistics and do other management. It is available on my file server at <http://192.168.210.254:3142/acng-report.html>. You will need to replace the IP in the above configurations with the IP of your apt-cacher-ng server.

Lots and Lots of Vanilla Installs

Now that I have all of this set up, I can install Linux to my heart's content. There is only one problem. Every time I install it, I know I am going to have to spend some time configuring the box to my liking. It's not impossible to get everything set up by hand, but it is incredibly annoying. So the next thing I wanted was a way to customize my install.

I have three different use cases: a standard package I would like installed automatically (MySQL), a non-standard package I would like installed automatically (Skype) and a piece of software that is not a package that I would like installed automatically (RubyMine).

Two different tools can handle automating the install process—preseeding and kickstart. Preseeding is a technology that comes to Ubuntu from Debian, and it allows you to automate the process of answering questions the installer will ask. Kickstart is a technology from Red Hat, developed to automate installing a Linux system. I have used preseeding for small tasks in the past, and I have used kickstart to automate the installation of a large number of servers. I never really thought about mixing the two technologies together, but it turns out, you can.

You can install a graphical client for generating kickstart configurations by installing the following tool:

```
sudo apt-get install system-config-kickstart
```

Then, you can run the application by going to Applications→System Tools→Kickstart. This allows you to use kickstart's features, but they have added support for calling preseed—meaning you get the

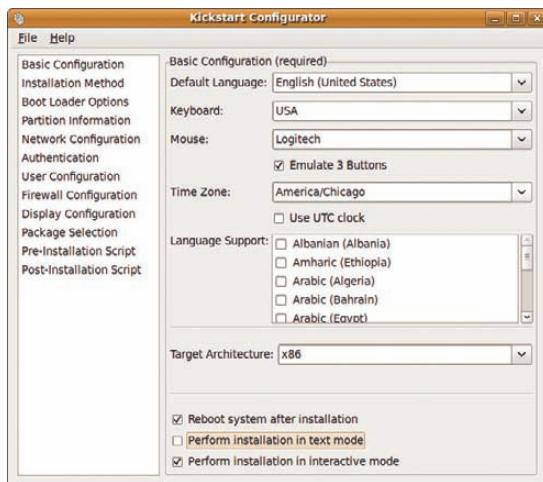
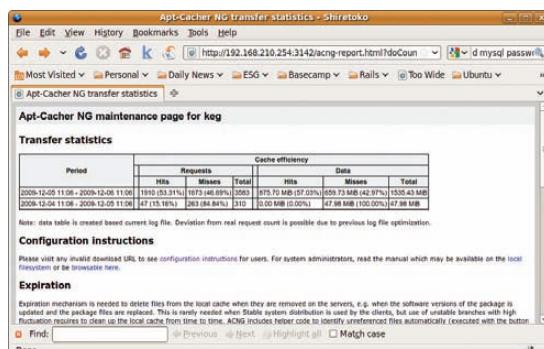


Figure 3. Kickstart Configurator

best of both worlds.

This allowed me to go through and preselect things (like defaulting to America/Chicago time zone). An important option is selecting the right source for the location from which to download the packages. If you provide the name of your apt-cacher-ng (in my case 192.168.210.254:3142) and tell it the URL is /us.archive.ubuntu.com/ubuntu, the installation will use your cache automatically, which is the reason you set it up, right?



Configuration instructions

Please visit any invalid download URL to see configuration instructions for users. For system administrators, read the manual which may be available on the local filesystem or be browsable here.

Expiration

Expiration mechanism is needed to delete files from the local cache when they are removed on the servers, e.g. when the software versions of the package is updated and the package files are replaced. This is rarely needed when Stable distribution is used by the clients, but use of unstable branches with high fluctuation requires to clean up the local cache from time to time. ACMG includes helper code to identify referenced files automatically (executed with the button).

Figure 4. Apt-Cache NG Reports

The screenshot of the Apt-Cache NG reports (Figure 4) shows the cache results after two installs. What the picture does not show is that during the first install, I left to do something else (because of the download speed), but all installs after that went fast enough that I could watch the install process.

There appears to be a long-standing bug in the tool when it comes to package selection. It does not actually show you a list of packages from which to choose, and you cannot type in packages. (A list of valid groups can be retrieved by executing `tasksel --list-tasks`). In the end, I used the GUI to generate a basic `ks.cfg` file. Then, I edited the rest by hand. This allowed me better control over the process.

Fanless HD Media Players with Ubuntu Linux

Ruggedized NVIDIA® ION™ System

No fans, no moving parts. Just quiet, reliable operation.
NVIDIA® GeForce® 9400M graphics for 1080p playback.



Low-Profile Intel® Atom™ Embedded System

Small footprint platform featuring solid state USB storage. Broadcom Hardware Decoder enables 1080p support.

Value only an Industry Leader can provide.

Selecting a complete, dedicated platform from Logic Supply is simple: Pre-configured systems perfect for both business & desktop use, Linux development services for greater system customization, and a wealth of online resources all within a few clicks.

[Learn More >](http://www.logicsupply.com/linux) www.logicsupply.com/linux

I used the preseed command to supply my own password to the MySQL server being installed. You can get a list of options available for preseeding by installing a package (debconf-utils), and then use the following command to see what is set:

```
sudo debconf-get-selections|grep -i mysql
```

Each one of these settings can be provided in the preseed section of the ks.cfg.

For the other two cases, I just handled it in the post section, which allowed me to download the files from the network and move them to the right place. Also note, I am getting them directly from the provider. In an actual production environment, I would keep a local copy.

Here is the relevant section of my ks.cfg:

```
#Package install information

preseed --owner mysql-server-5.1
  ↵mysql-server/root_password password qwerty
preseed --owner mysql-server-5.1
  ↵mysql-server/root_password_again password qwerty

%packages
@ ubuntu-desktop
mysql-server-5.1
mysql-common
mysql-client-5.1
debconf
debconf-utils
wget
#Skype dependencies
libqt4-network
libqt4-dbus
libqtcore4
libqtdgui4
libxss1
libxv1
libaudio2
libmng1
libqt4-xm1

%post --nochroot
mv /target/etc/rc.local /target/etc/rc.local.orig
echo '#!/bin/bash
cd /root;/root/first_boot
exit 0' > /target/etc/rc.local
cat > /target/root/first_boot << EOF
#!/bin/bash
cd /root
#Fetching skype
wget http://www.skype.com/go/getskype-linux-beta-ubuntu-32
dpkg -i skype-ubuntu-intrepid_2.1.0.47-1_i386.deb
#Fetching Rubymine
```

```
wget http://download.jetbrains.com/idea/rubymine-2.0.tar.gz
mkdir /opt/rubymine
tar -xz --strip 1 -C /opt/rubymine/ -f rubymine-2.0.tar.gz
rm /etc/rc.local
mv /etc/rc.local.orig /etc/rc.local
EOF
chmod a+x /target/etc/rc.local
chmod a+x /target/root/first_boot
```

Putting the Plan into Action

Now that you have a kickstart configuration file, you need to get the system to use it. You need to host the file with a Web server. In my case, keg already has Apache running, so I can just stick it up there.

Then, I just add a new option to my boot.txt:

karmic_i386_install_seeded

And, I added a new stanza to my pxelinux.cfg/default:

```
LABEL karmic_i386_install_seeded
  kernel ubuntu/karmic/i386/ubuntu-installer/i386/linux
  append vga=normal initrd=ubuntu/karmic/i386/
    ↵ubuntu-installer/i386/initrd.gz
    ↵ks=http://192.168.210.254/ks.cfg --
```

Now I can choose to boot a standard installer or one that has been preseeded with other software. My first install was completed using a KVM virtual machine. To get it to PXE boot, I needed to set it up to bridge onto my internal network (so it could get a DHCP address and talk to the PXE server), and I had to download a gPXE ISO (rom-o-matic.net/gpxe/gpxe-0.9.9/contrib/rom-o-matic). That allowed me to PXE boot the machine even though the KVM BIOS did not support PXE.

Final Thoughts

The kickstart/preseed stuff was more complicated than I would have liked. It took a lot of rebooting and re-installing to get the syntax correct so that all three cases worked. I found tons of introductory documentation, but as soon as I needed something more detailed, I was on my own. For example, I did not have the owner option set on my preseed, and I originally wanted to do the downloading and installing of Skype in the post but had to resort to using a first boot script. It made me understand why people end up writing their own "post-install" scripts instead of using these larger tools. That being said, now that I have all this together, installation should be a breeze. Maybe I finally will get around to bringing all my machines up to a common and current version of Ubuntu.■

Dirk Elmendorf is cofounder of Rackspace, some-time home-brewer, longtime Linux advocate and even longer-time programmer.

**REGISTER
to attend
DISCOUNT
Feb. 12, 2010**

sxsw.com



SXSW INTERACTIVE FESTIVAL: CONNECT, DISCOVER, INSPIRE

Attracting digital creatives and new media entrepreneurs, the 16th annual South by Southwest (SXSW) Interactive Festival gives you both practical how-to information as well as unparalleled career inspiration. Attend this legendary gathering of the tribes to renew your link to the cutting edge.

SCHEDULED 2010 SPEAKERS:

danah boyd to deliver Opening Remarks
Saturday, March 13, 2010

Spotify's **Daniel Ek** to Keynote SXSW Interactive on Tuesday, March 16, 2010

SCHEDULED 2010 PANELS

INCLUDE: All About the Browser, Baby! • Can Wikipedia Survive Popular Success and Community Decline? • Design for the Dark Side • Engaging The Queer Community • From Trolls to Stars: The Commenter Ecosystem • Gary Vaynerchuk Presentation • History of the Button • I Don't Trust You One Stinking Bit • Jacks of All Trades or Masters of One? • Made It So (Interface Makers in Movies) • Offering Your Content in 100 Languages • Paul Boag Presentation • Real-Time Everything: the Era of Communication Ubiquity • Selling Subculture Without Selling Out • Trials and Tribulations of the Pirate Bay • Unsexy & Profitable: Making \$\$ Without • Visual Note-Taking 101 • Web Framework Battle Royale • You Developed the Content—Now Build The Hardware

For a complete list of currently confirmed sessions for the 2010 event go to:
sxsw.com/interactive/talks/panels

REGISTER TO ATTEND SXSW INTERACTIVE 2010

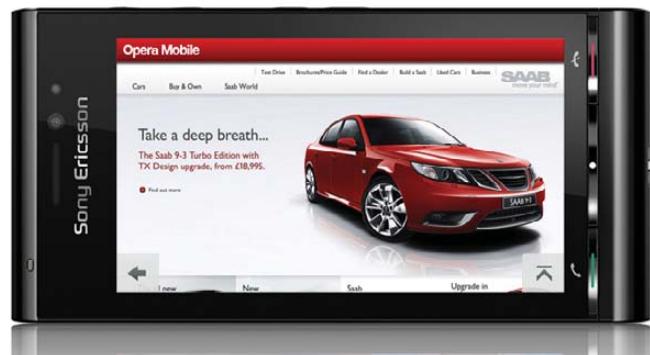
Register before **February 12, 2010** to receive the next early bird rate and get your choice of the best hotels available: sxsw.com/attend

Attend SXSW Film and SXSW Interactive at a bargain rate by purchasing a Gold Badge.

Opera Mobile

While the Opera Web browser has yet to conquer our readers' PCs, the browser maker appears to have had more success on mobile devices. Case in point is Opera Mobile, now in version 10, the cross-platform UI framework for Android, BREW, Windows Mobile and Symbian/S60 smartphones. Opera says that Mobile 10's raison d'être is to open up the Opera browser experience to more people, on more devices, allowing "operators and OEMs to implement the same user experience quickly and cost effectively across their entire range of handsets". Other features include a rich Web 2.0 experience optimized for mobile phones, Opera Turbo data compression technology and the Opera Widgets standalone mini-Web apps.

www.opera.com/business/solutions/mobile



Mandriva InstantOn

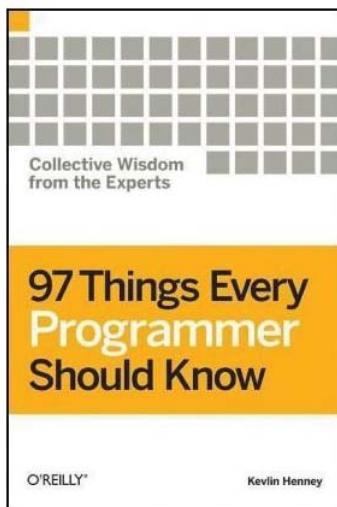
Following on the success of DeviceVM's Splashtop application, Mandriva has introduced InstantOn, a Linux-based application that brings up a usable interface on virtually any PC in a matter of seconds. Designed to complement a base operating system (Linux or Windows), InstantOn offers a choice of applications for near instant display—that is, less than ten seconds and even less than that for hard drives with Flash memory. Applications include Firefox, Rhythmbox, Pidgin, Skype and Thunderbird. An OEM version will offer a customizable interface and 20,000 applications from which to choose.

www2.mandriva.com

chicBuds chicboom Keychain Speaker

Although we failed miserably on getting you this info by Christmas, let us hook you up for Valentine's Day gift-giving (and receiving!). The chicboom Keychain Speaker is designed for the stylish woman who wants a big, mobile sound in a small package. The amplified speaker, which one can attach to any device with a standard 3.5mm stereo jack (MP3s, iPods, laptops and so on), needs only 2 Watts and runs a full four hours on a single charge.

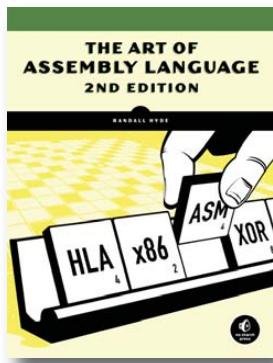
chicbuds.com



Kevlin Henney's 97 Things Every Programmer Should Know (O'Reilly)

Editor Kevlin Henney has distilled essential wisdom from the programming craft into one concise O'Reilly volume, titled *97 Things Every Programmer Should Know: Collective Wisdom from the Experts*. The book contains 97 "short and extremely useful programming tips" from some of the most experienced and respected practitioners in the industry, including Uncle Bob Martin, Scott Meyers, Dan North, Linda Rising, Udi Dahan, Neal Ford and many others. These veterans encourage programmers to push their craft forward by learning new languages, looking at problems in new ways, following specific practices, taking responsibility for their work and becoming as good as possible at the entire art and science of programming. The focus is on practical principles that apply to projects of all types. One can read the book end to end or browse it to find topics of particular interest.

oreilly.com



Randall Hyde's *The Art of Assembly Language, 2nd Edition* (No Starch)

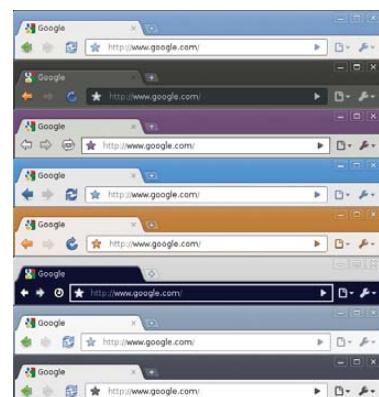
Now in its second edition, Randall Hyde's *The Art of Assembly Language* from No Starch Press has been updated thoroughly to reflect recent changes to the High Level Assembler (HLA) language, the book's primary teaching tool. The comprehensive, 800-page guide teaches programmers how to understand assembly language and how to use it to write powerful, efficient code. It further demonstrates how to leverage one's knowledge of high-level programming languages to make it easier to grasp basic assembly concepts quickly. All code from the book is portable to the Linux, Mac OS X, FreeBSD and Windows operating systems.

nostarch.com

Google Chrome for Linux and Mac OS

If you read our interview several months ago with the Google Chrome team (*Linux Journal*, October 2009), you know that the Linux and Mac editions of Google Chrome were on their way. Well, now they've arrived, are in beta (quite stable by Google's standards) and even offer a range of more than 300 extensions (more advanced on Linux and Windows than Mac). Chrome is built for speed and simplicity and is intended to leapfrog other browsers that were built before the age of rich and complex Web apps. The key innovation on all platforms is the V8 JavaScript engine. Critical elements of the Linux variant are tight integration with native GTK themes and updates that are managed via the standard system package manager. It is reported that the Linux version came along faster than expected due to the yammering of Google's engineers, most of whom run—of course—Linux.

www.google.com/chrome



Black Duck Software's Enterprise Code Search Initiative



Continuing on its mission to improve open-source-based software development, Black Duck Software recently released several new elements in its Enterprise Code Search Initiative. The code search initiative involves three phases: expansion of open-source code available at Koders.com, release of Black Duck Code Sight Enterprise Edition and Free Edition for enterprise code search and an open integration framework initiative for community expansion of integrations with source code management systems. Koders.com is Black Duck's free code search Web site that has been expanded to access more than 2.5 billion lines of open-source code. Black Duck Code Sight is a tool offering enterprise-level code search capability that can index and make software searchable across multiple source code repositories for local or geographically distributed development teams. Finally, the open integration framework offers built-in integration for IBM Rational ClearCase, Subversion, Git, Microsoft Team Foundation Server and other code management systems.

blackducksoftware.com/code-sight

Gluster Storage Platform

The new Gluster Storage Platform combines the GlusterFS filesystem with a new user interface and operating system layer for massively increased performance and improved ease of use. Gluster says that its product allows one to "deploy petabyte-scale storage on industry-standard hardware in just 15 minutes with centralized management and automated maintenance". It works by clustering together storage building blocks, aggregating disk and memory resources and managing data in a single unified namespace. Advantages include low storage cost, high scalability, no bottlenecks (thanks to the lack of a metadata server) and virtual storage for virtual servers.

gluster.com



Please send information about releases of Linux-related products to newproducts@linuxjournal.com or New Products c/o *Linux Journal*, P.O. Box 980985, Houston, TX 77098. Submissions are edited for length and content.

Fresh from the Labs

CountBeats—BPM Finder

www.mellowood.ca/countbeats/index.html

This month, I'm covering projects I've wanted to showcase but have held off on due to space constraints. CountBeats is a cracker of a little application—it's simple, yet it covers such a need for so many musicians, so I'm proud to give it top billing here. To quote the README file:

This is a simple little program designed to help you determine the speed of a piece of music on the radio or on a CD.

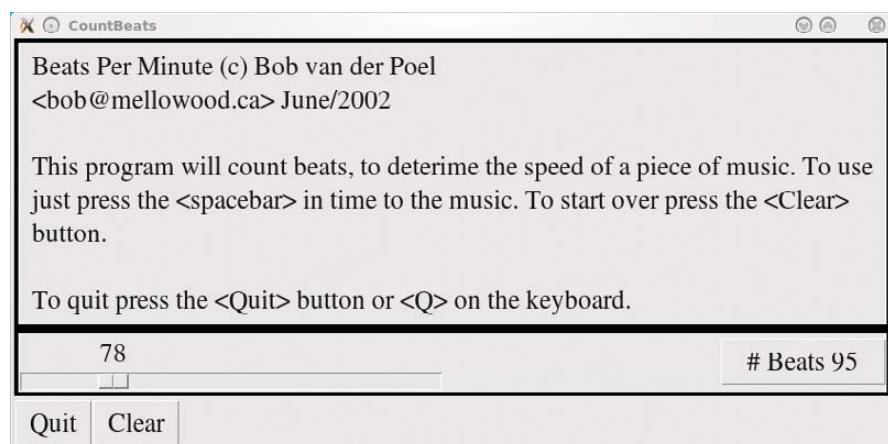
To use it, just invoke the program from the command line. You'll be presented with a screen describing the program, a time-bar, and a few buttons. Most should be self-explanatory. Just start tapping the spacebar in time to the music...the tempo bar will update and show how many beats per second are being played.

Installation First, there is a library you probably won't have installed: Tkinter. This was under the package name python-tk on my system, but it's worth looking in your local package manager for yours. The only other requirement seems to be a working version of Python 2.x onward.

Otherwise, installing this thing is a walk in the park. Simply download the 1K (!) tarball from the Web site, extract it, and open a terminal in the new folder. From here, run CountBeats simply by entering:

```
$ ./countbeats
```

Usage The README file did a pretty good job of explaining how it all works, but I can expand on it a little further. As soon as you have a song for which you want to find the BPM rate, start the program and press the spacebar in time with the music. The more times you do this, the more accurate the rating will be, so stick with it for a good minute or two. If you want to clear away previous readings and start again (perhaps when another



CountBeats is an ingenious little program for determining the tempo of a song by simply tapping away on the keyboard in time to the music.

song comes on), click Clear at the bottom-left corner.

Given the cumbersome nature of a spacebar, it's generally best if you track a song's quarter notes, as eighths or sixteenths may track a bit inaccurately. If you know your way around Python syntax (which I don't), you may want to change the key to something like Ctrl for faster music.

That small niggle aside, this is an invaluable tool for musicians. Many times I've been working on a project and forgotten to note what tempo a song was in, making tracking a nightmare at times. DJ-remixing types probably will use this most of all, as they can use it to gauge the speed of whatever songs they're piecing together and work out which samples will be compatible with each other. All in all, this is a brilliant tool that is simple to use and install, and it probably takes the prize for smallest file size of any project I've covered!

wxGuitar—Tuning Reference for Acoustic Guitars

freshmeat.net/projects/gguitar-01

If you like minimalism and are chasing a tuning reference for your acoustic guitar, this may well be the project for you. I especially meant that minimalism part—the only info I could find on the project's Freshmeat entry was the following: "wxGuitar is a useful application that will easily help the novice

guitarist to faster (and better) tune the guitar." And, that's pretty much it—that's all the information I can find anywhere on the Net. But, maybe I can illuminate things a little here.



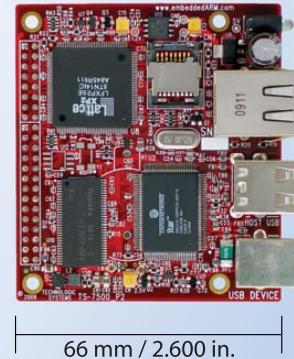
wxGuitar is a very simple reference tool for tuning an acoustic guitar by ear. But an H string? An amusing typo!

Installation I couldn't find an actual home page for wxGuitar, so you have to make do with the files provided on the (brief) Freshmeat page. Source is provided, along with Debian and Gentoo packages. As far as libraries go, the INSTALL file says you'll need wxWidgets >= 2.8.10 (I had to install

TS-7500 Embedded Computer

Faster. Smaller. Cheaper.

Qu. 100 \$84



libwxbase2.8-dev), along with als-utils, including aplay. If you're running with the source, once you have the needed libraries, grab the latest tarball, extract it, and open a terminal in the new folder. Enter the following commands to compile wxGuitar:

```
$ ./configure  
$ make
```

If your distro supports sudo:

```
$ sudo make install
```

If not:

```
$ su  
# make install
```

Once wxGuitar is installed, you may have it in your system's menu, or you can run it with the command:

```
$ wxGuitar
```

Usage When you're inside the main screen, I think you'll find it rather straightforward. wxGuitar is very basic. Turn your speakers on and press any of the buttons on the left to play the corresponding note. The notes start from the highest note E string and go down to the lowest note E string. The second button is curiously marked H, but unless there's some kind of unique Eastern European scale I don't know about, logic dictates that should be a B string.

Press the button on the left for each string's note, and play the corresponding

string on your own guitar as you fine-tune it. If you look on the bottom-left corner, there's a repeat option that will be set to every three seconds by default, which can be made longer or shorter if you so desire.

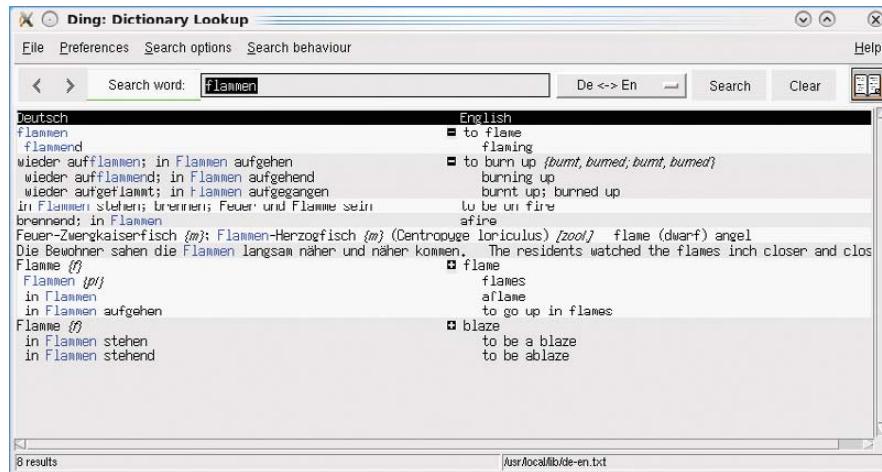
That's pretty much all there is to it. wxGuitar may not be complex (and if you're looking for a sophisticated tuning recognition program, you should look elsewhere), but if you want a minimalist program to tune by ear, this is probably for you.

Ding—German-to-English Reference

www-user.tu-chemnitz.de/~fri/ding

Are you a Rammstein fan trying to decipher those wacky lyrics or a Porsche fan trying to figure out exactly what *Doppelkupplungsgetriebe* means? Are you chasing a German-English translator that's simple to use and painless to install? This is probably the best place to start, especially if you have to type such accents as umlauts and the like (see what I mean further on). According to Ding's Freshmeat entry:

Ding is a dictionary lookup program for the X Window System on Linux/UNIX. It comes with a German-English dictionary with about 253,000 entries. It is based on Tk version >= 8.3 and uses the agrep or egrep tools for searching. In addition, Ding also can search in English dictionaries using dict(1) and check spelling using ispell(1). It has many



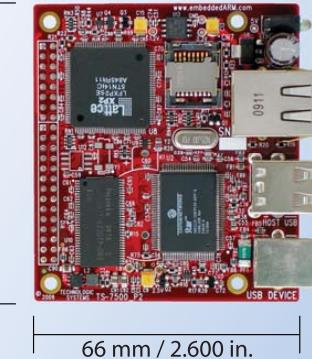
Ding is a simple yet powerful dictionary tool for translating between German and English. Note the handy collapsing menus for derivations on root words.

TS-7500

Embedded Computer

Faster. Smaller. Cheaper.

Qu. 100 \$84



Powered by a
250 MHz ARM9 CPU

- » Low power, fanless, < 2 watts
- » 64MB DDR-RAM
- » 4MB NOR Flash
- » Micro-SD Card slot - SDHC
- » USB 2.0 480Mbit/s host (2) slave (1)
- » 10/100 Ethernet
- » Boots Linux in less than 3 seconds
- » Customizable FPGA - 5K LUT
- » Power-over-Ethernet ready
- » Optional battery backed RTC
- » Watchdog Timer
- » 8 TTL UART
- » 33 DIO, SPI, I²C

» Dev Kit provides out-of-box development + extra features

- » Over 20 years in business
- » Never discontinued a product
- » Engineers on Tech Support
- » Open Source Vision
- » Custom configurations and designs w/ excellent pricing and turn-around time
- » Most products ship next day

 **Technologic**
SYSTEMS

We use our stuff.

visit our TS-7800 powered website at

www.embeddedARM.com

(480) 837-5200

Updates

Gnaural

gnaural.sourceforge.net

In the spirit of this month's "catching up" column, I'm taking a look at recent developments in some of the coolest projects I've been fortunate enough to cover here at *LJ* in the past. My favorite project of all time is Gnaural. For those not in the know, Gnaural is an application to generate Binaural Frequencies that can speed up or slow down brainwaves for relaxation or alertness, using a basic PC and a simple pair of headphones. A recent addition to the CVS code by a user under the mysterious nickname of noname36 has added the extra functionality of using Isochronic Tones instead of Binaural Frequencies, which brings yet more application to an already amazing program.

CloneKeenPlus/Commander Genius

clonekeen.sourceforge.net and
<http://clonekeenplus.sourceforge.net>

CloneKeen, an authentic rebuild of the classic PC platformer *Commander Keen*, has continued to become more stable and has been ported to more platforms. A separate project



CloneKeen

configuration options, such as search preferences, interface language (English or German) and colors. It has history and help functions and comes with useful key and mouse bindings for quick and easy lookups.

Installation Installing Ding is really easy. Head to the Web site, and you'll find a number of different packages along with a tarball. As usual, I'm running with the tarball for the sake

of neutrality. Download the tarball, extract it, and open a terminal in the new folder.

If your distro supports sudo, enter:

```
$ sudo ./install.sh
```

If not, enter:

```
$ su
# sudo ./install.sh
```

You'll want to install the German

aspell files for certain parts of program functionality. Once the installation has finished, run the program with:

```
$ ding
```

Usage Although I haven't much space to cover Ding's usage, the interface is pretty basic anyway. To start, enter a word in German or English, and either click Search or press Enter. At this point, any translated possibilities and variants show



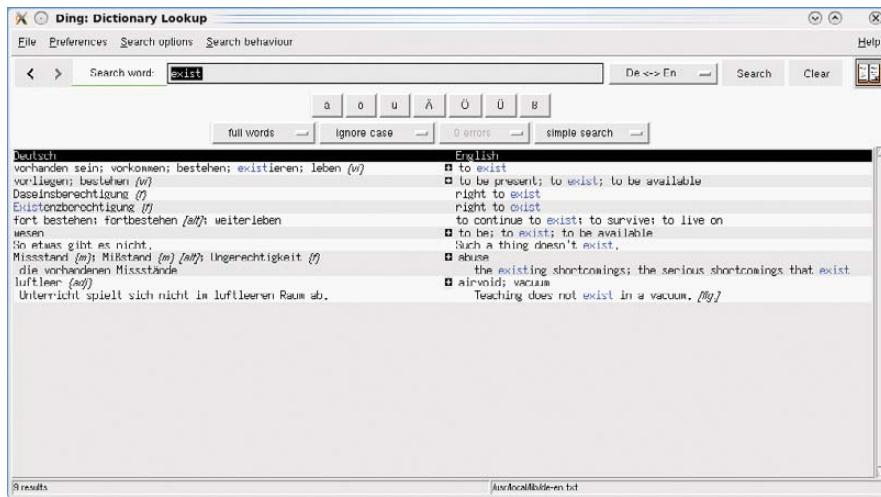
Vdrift

run by other fans also has started up, *Commander Genius* (aka *CloneKeenPlus*), which includes things like OpenGL 2.0 support, new graphical effects and a Normal and Hard mode, among other new features. Thankfully, *CloneKeen*'s creator Caitlin Shaw also has joined the project.

Vdrift

vdrift.net

This racing sim, aimed at the realistic racing crowd, has been gaining more realism and features over time. New features include car collision in Single Race mode, a much more capable AI (along with a new difficulty slider) as well as improved performance. But what really jumps out at me are tweaks to the actual feel of the game. It previously felt very detached, often making the process of driving more of an intellectual exercise than an intuitive one. Small touches, like tyre-spin noise when off-road, and bigger ones, such as a "bouncy" camera for hood view, should make for a game that is much more playable with a solid feel. I look forward to the full release.



A particularly handy feature is quick access to Germanic characters—something most English-based OS installations won't be set up for.

up below, with Deutsch (German) on the left and English on the right. You also may see a small cross icon next to each translation. Click it, and variants

are displayed in a collapsible menu from the root word, such as plurals, example usage and so on.

It's well worth looking in the

Preferences menu and clicking the Show umlaut buttons option. This shows the special Germanic characters most English editions of OSes won't be set up for. Other features include a spell checker, as well as orthography, but I'll let you explore things for yourself from here.

Although Ding may be rather gray and not pretty, it's nice and minimalist and easy to install with a minimum of fuss. I'm sure that 90% of its users will be those English-speaking Rammstein fans trying to work out what's being said, but why not? ■

John Knight is a 25-year-old, drumming- and climbing-obsessed maniac from the world's most isolated city—Perth, Western Australia. He can usually be found either buried in an Audacity screen or thrashing a kick-drum beyond recognition.

Brewing something fresh, innovative or mind-bending? Send e-mail to newprojects@linuxjournal.com.



Meet Victoria (on the right). She is the Silicon Mechanics marketing expert responsible for the events and promotions that keep our customers informed about new products and technologies. She's pictured here with her twin sister Veronica, an industrial designer, to help us make a point about what makes twin servers so popular. Victoria and Veronica are twins, but they don't look exactly alike and they don't do the same job. Twin servers are two servers in a single 1U chassis: they can be configured differently, and they handle their own individual workloads.



For more information about the Rackform iServ R4410 visit www.siliconmechanics.com/R4410



**Powerful.
Intelligent.**

With the introduction of the Rackform iServ R4410 from Silicon Mechanics, twin power has reached a whole new level: the twin². A twin² is a 2U 4-node system. It supports four swappable, full-featured nodes in a 2U chassis with redundant power. In each node you'll find 2 of the new Intel® Xeon® 5500 Series processors, 12 DDR3 DIMM slots, 3 hot-swap drives, and an integrated dual-port GigE adapter. Integrated InfiniBand is also available with the R4410-IB. Unmatched density and state-of-the-art processors make the R4410 a superior choice for high-performance computing, and Victoria is spreading the word with enthusiasm.

When you partner with Silicon Mechanics, you get more than the latest and greatest in density, performance, and energy efficiency—you get an expert like Victoria.

Expert included.

Silicon Mechanics and the Silicon Mechanics logo are registered trademarks of Silicon Mechanics, Inc. Intel, the Intel logo, Xeon, and Xeon Inside, are trademarks or registered trademarks of Intel Corporation in the US and other countries.

SOFTWARE

Axigen Mail Server

Axigen provides a professional e-mail server with AJAX Web interfaces for both administration and user Web mail. With Axigen, you'll have your e-mail server up and running in minutes, not days. MITCH FRAZIER

The **Axigen Mail Server** from Gecad Technologies is a pre-packaged e-mail and collaboration suite. For those who eschew closed-source, non-free software, you can stop reading now, but if you're looking for a full-featured, easy-to-install, easy-to-use e-mail setup that won't break the bank, read on.

A trial version of the Axigen suite is available for download. The first pleasing thing about Axigen is that there's no annoying registration required to download (it asks, but you can skip it and just download it). The download comes as a single install file for most popular Linux distributions as well as for Windows, Solaris and most flavors of BSD.

The second pleasing thing about Axigen is that it's easy to find the prices

Remember what Axigen is—an e-mail server with a Web mail interface—so Axigen includes a Web server, an SMTP server, a POP server and an IMAP server.

on the Web site (you know you're in trouble when a company won't tell you how much something costs). Axigen comes in three different versions: Business Edition, Enterprise Edition and Service Provider Edition. Within each edition, the pricing model is based on the number of e-mail users you have, although the pricing is tiered (that is, you buy a license for 25 users even if you need only 17). The initial purchase price starts out at about \$16 per user for the Business Edition, about \$30 per user for the Enterprise Edition, and

about a \$1.25 per user for the Service Provider Edition. Prices drop to about \$7, \$13 and \$0.89, respectively, for each edition at the top tier. If you need more users than the top tier provides, you have to contact the Axigen folks and arm-wrestle with them. The initial purchase price gives you a perpetual license to use the product. After that, if you choose to do so, you can buy an annual support contract for about 25% of the original purchase price. Note that there's also one more edition of Axigen available: an Office Edition that's free for up to five users.

Installing Axigen is simple: get the download appropriate for your distro, and as root, run the file as a script via the shell. The meat of Axigen is contained as a binary payload within the script. After the initial steps are completed (including accepting the EULA), the Axigen Config utility, a text-based GUI, starts (Figure 1). The Config utility allows you to configure things that must be known before the Axigen server can start, such as the administrator password; the

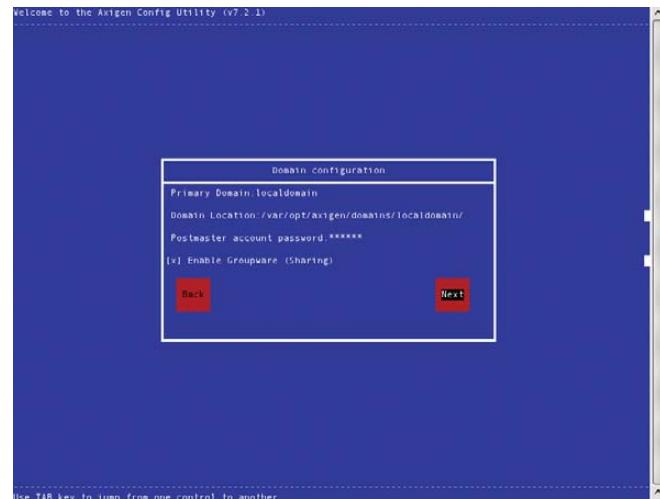


Figure 1. Text-Based GUI for Initial Install Steps



Figure 2. Install Requesting Shutdown of Already-Running Web Server

primary e-mail domain; the Web-based administration interface; what interfaces and ports to use for the SMTP server, the POP server, the IMAP server, and the Web server for the Web mail interface; and whether to install a wrapper for sendmail so that normal command-line tools can send mail via Axigen.

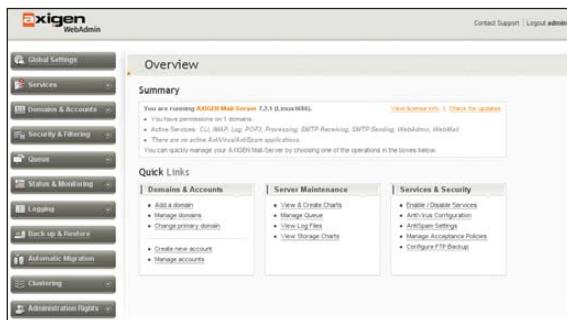


Figure 3. Axigen Web-Based Administration Interface



Figure 4. Adding a New E-Mail Account

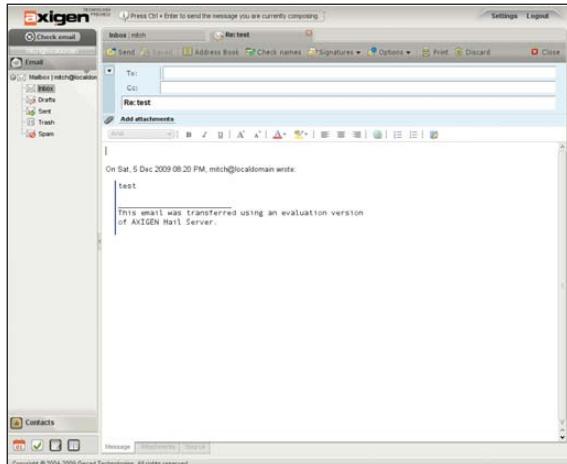


Figure 5. Sending an E-Mail from the Web Mail Interface

Remember what Axigen is—an e-mail server with a Web mail interface—so Axigen includes a Web server, an SMTP server, a POP server and an IMAP server. If you already have Apache, Postfix or other Web/mail servers running on your system, you need to shut them down and make sure they don't start when the system boots. If Axigen discovers a server running on a port that it needs to use, it will ask you to stop it before proceeding (Figure 2).

use. It's as AJAX-y of an interface as I've seen anywhere—while exploring the interface, I don't recall ever seeing a full page reload.

The first obvious step in testing, adding a new e-mail account, shows the flavor of the interface: click on the Add Account button, and a pop-up window appears where you can add the basic information for the new user (Figure 4).

After adding a user, I sent an e-mail

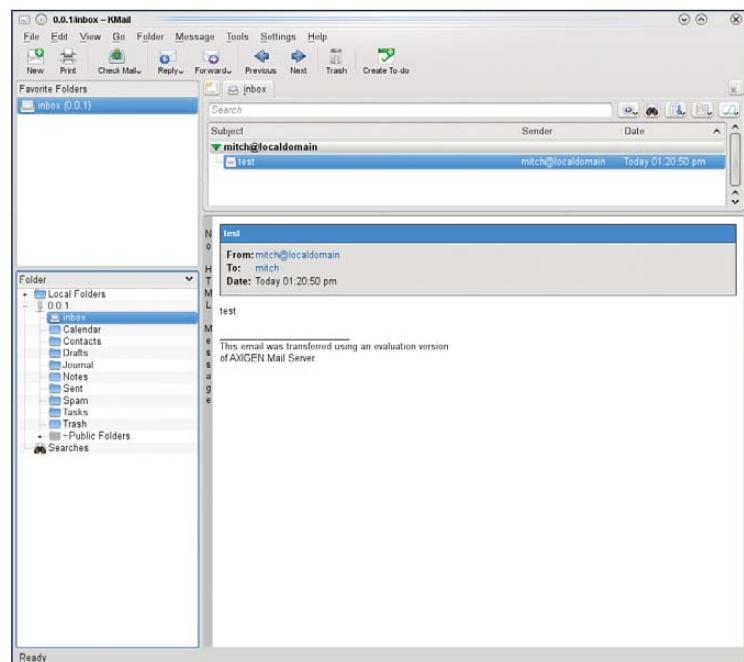


Figure 6. Axigen-Based E-Mail Account in KMail

After the command-line install steps are completed, you need to start the Axigen server. The install puts a startup script in init.d; run it, and the Axigen server should start. If you're not just testing, you also want to make sure that the server gets started when the system boots by configuring your system to run the init.d script when it boots.

Next, point your Web browser at 127.0.0.1:9000. After entering the admin password, you can get to Axigen's Web-based administration interface (Figure 3). Axigen's admin interface is attractive and easy to

from the command line to the newly added user (mitch@localdomain) and then opened a new browser tab, pointed it at 127.0.0.1 (no port) and logged in to my new e-mail account (Figure 5). My test mail was there waiting for me.

The e-mail account user interface follows in the footsteps of the admin interface; it's both attractive and easy to use. The e-mail interface includes folder management, contact management and access to all the normal operations one expects in an e-mail client. In your account settings, it even includes the ability to enable and disable shortcut keys, although it doesn't allow you to change them. E-mail sending includes a WYSIWYG editor (Figure 5). Notice also in Figure 5 that the interface itself is tabbed. There's a tab for the open Inbox and one for the e-mail that's being composed.

Also available from here (depending on the edition you have) are links to the included Calendar, Journal, Tasks and Notes applications. As with the e-mail application, these applications provide the normal features you'd expect. I did not test these applications extensively, but I did notice that when I assigned a task, Axigen automatically sent me an e-mail to notify me that I'd been assigned a task. All in all, there were no big surprises from the e-mail application or from any of the other included applications,

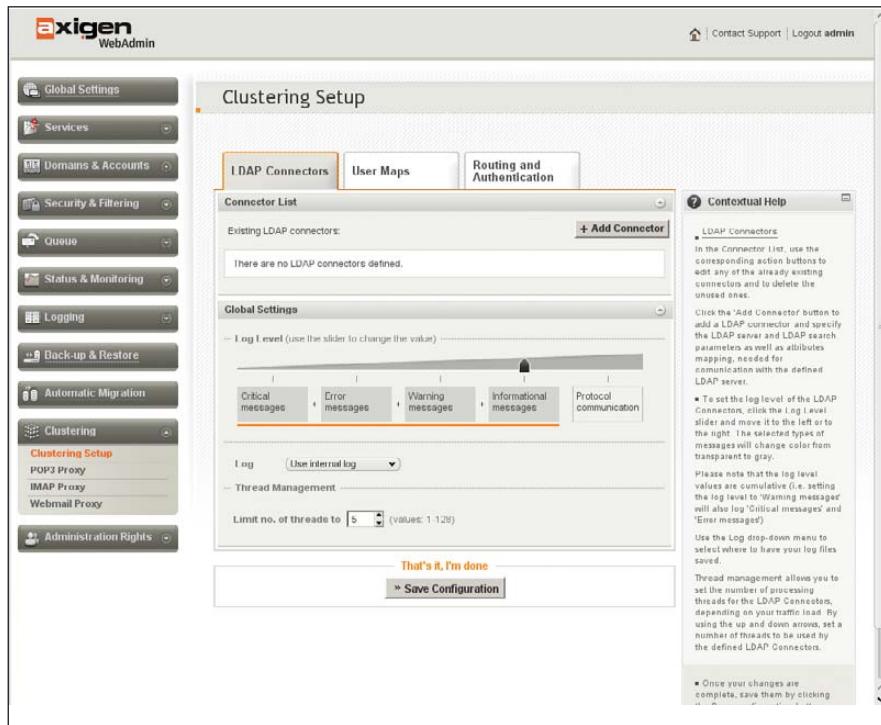


Figure 7. Context-sensitive help appears on the right-hand side of most pages.

and no big disappointments either.

As a last check of the non-administration features, I tried to hook up KMail to my new e-mail account. This also was painless. I entered the server names, logged in, and there was my test e-mail (Figure 6). Somewhat unexpectedly, I also could see my Contacts, Tasks, Journals,

also supports interfacing with a number of third-party spam and virus filters, including SpamAssassin, AVG and ClamAV among others. It also provides the ability to back up data to an FTP server. A couple of the more interesting features I ran across were RPOP and Automatic Migration.

Axigen's admin interface is attractive and easy to use. It's as AJAX-y of an interface as I've seen anywhere—while exploring the interface, I don't recall ever seeing a full page reload.

Notes and Calendar events in KMail.

Task e-mails contain links in them that appear to allow interaction with the Axigen's database, but clicking on the links in KMail brings up the KDE Calendar and an error message. The links are generated from some vCalendar "text/calendar" content types in the e-mail, so perhaps there is a possibility to make external programs interact with the Axigen calendar, but I didn't pursue that any further.

Back to the admin interface to explore a bit more, I noticed that Axigen

RPOP is Remote POP, which allows fetching of mail from existing POP accounts. The interesting part here is that it allows individual users to define the accounts from which they'd each like to fetch mail.

Automatic Migration allows you to migrate your existing POP/IMAP e-mail server to Axigen. What's nice about this is that it's a lazy migration. You simply define where your old server is, and when a user logs in for the first time, if Axigen doesn't find the user in the current configuration, it checks the old server and migrates the user if the user has an account on the old server. Of course, if the user doesn't have an account on the old server, the login simply fails. Automatic Migration is a bit like RPOP but only for the installation as a whole. Plus, it's done only the first time a user connects.

The Axigen admin pages all include context-sensitive help along the right side of the page (Figure 7). The Axigen Web site also includes a number of downloadable documents and on-line resources for the product. Included in the documentation is a reference to the Axigen API, which allows you to write PHP programs to administer and access the Axigen server/data.

The Axigen server runs as two processes, the second a fork of the first. The first probably monitors the second to make sure it's still running. The second process contains a couple dozen threads. The memory footprint is between 300 and 400 megabytes.

Without a doubt, I'm a fan and an advocate of free and open-source software, but I'm not a zealot on those counts either. I have no objection to paying for software. And, you certainly could get most of what Axigen provides by assembling the right combination of FOSS products, but it's doubtful that the end result would be as professional-looking or consistent as the Axigen Mail Server. And, it almost goes without saying that you'd never get it all up and working as quickly as you can install Axigen and get it going. So, if you're willing to consider closed-source, non-free software, and you need an e-mail server that includes Web mail access for your users, give Axigen a look.■

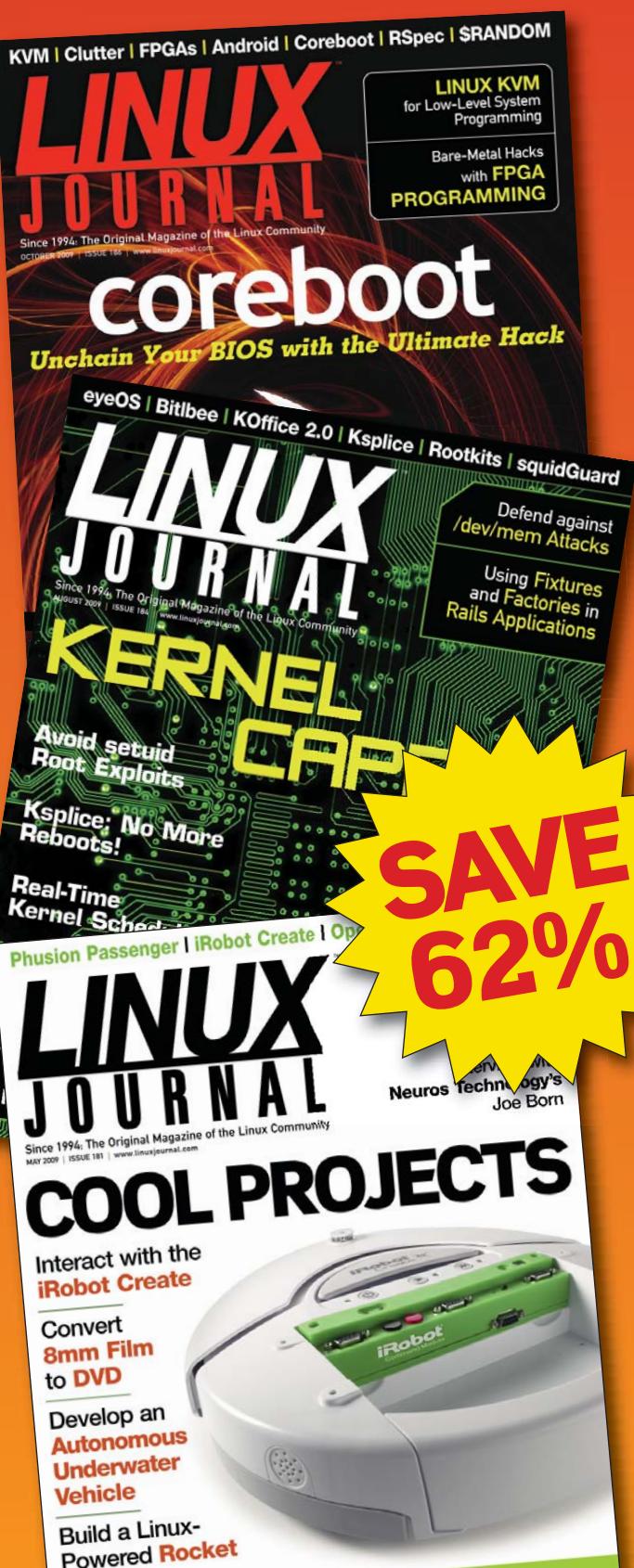
Mitch Frazier is an Associate Editor for *Linux Journal* and the Web Editor for LinuxJournal.com.

Resources

Axigen: www.axigen.com

Axigen Downloads: www.axigen.com/mail-server/download

If You Use Linux, You Should Be Reading **LINUX JOURNAL**™



- » In-depth information providing a full 360-degree look at featured topics relating to Linux
- » Tools, tips and tricks you will use today as well as relevant information for the future
- » Advice and inspiration for getting the most out of your Linux system
- » Instructional how-tos will save you time and money

Get *Linux Journal* delivered to your door monthly for 1 year for only \$29.50! Plus, you will receive a free gift with your subscription.

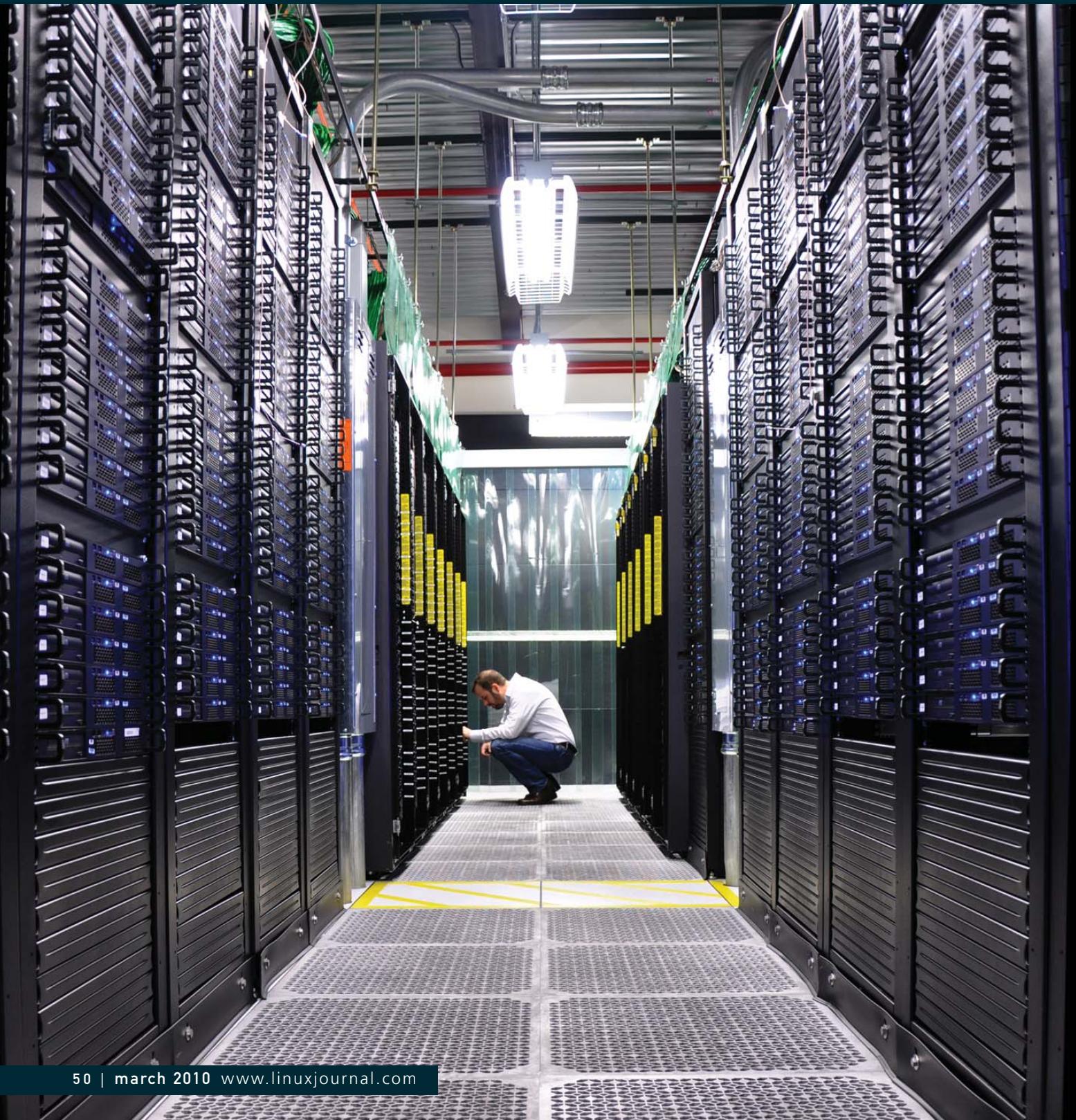
SUBSCRIBE NOW AT:
WWW.LINUXJOURNAL.COM/SUBSCRIBE

Offer valid in US only. Newsstand price per issue is \$5.99 USD; Canada/Mexico annual price is \$39.50 USD; International annual price is \$69.50. Free gift valued at \$5.99. Prepaid in US funds. First issue will arrive in 4-6 weeks. Sign up for, renew, or manage your subscription on-line, www.linuxjournal.com/subscribe.

TAMING the BEAST

The right plan can determine the difference between a large-scale system administration nightmare and a good night's sleep for you and your sysadmin team.

JASON ALLEN



As the appetite for raw computing power continues to grow, so do the challenges associated with managing large numbers of systems, both physical and virtual. Private industry, government and scientific research organizations are leveraging larger and larger Linux environments for everything from high-energy physics data analysis to cloud computing. Clusters containing hundreds or even thousands of systems are becoming commonplace. System administrators are finding that the old way of doing things no longer works when confronted with massive Linux deployments. We are forced to rethink common tasks because the tools and strategies that served us well in the past are now crushed by an army of penguins. As someone who has worked in scientific computing for the past nine years, I know that large-scale system administration can at times be a nightmarish endeavor, but for those brave enough to tame the monster, it can be a hugely rewarding and satisfying experience.

People often ask me, "How is your department able

to manage so many machines with such a small number of sysadmins?" The answer is that my basic philosophy of large-scale system administration is "keep things simple". Complexity is the enemy. It almost always means more system management overhead and more failures. It's fairly straightforward for a single experienced Linux sysadmin to single-handedly manage a cluster of a thousand machines, as long as all of the systems are identical (or nearly identical). Start throwing in one-off servers with custom partitioning or additional NICs, and things start to become more difficult, and the number of sysadmins required to keep things running starts to increase.

An arsenal of weapons in the form of a complete box of system administration tools and techniques is vital if you plan to manage a large Linux environment effectively. In the past, you probably would be forced to roll your own large-scale system administration utilities. The good news is that compared to five or six years ago, many open-source applications now make managing even large clusters relatively straightforward.

MONITORING

System administrators know that monitoring is essential. I think Linux sysadmins especially have a natural tendency to be concerned with every possible aspect of their systems. We love to watch the number of running processes, memory consumption and network throughput on all our machines, but in the world of large-scale system administration, this mindset can be a liability. This is especially true when it comes to alerting. The problem with alerting on every potential hiccup is that you'll either go insane from the constant flood of e-mail and pages, or even worse, you'll start ignoring the alerts. Neither of those situations is desirable. The solution? Configure your monitoring system to alert only on actionable conditions—things that cause an interruption in service. For every monitoring check you enable, ask yourself "What action must be taken if this check triggers an alert?" If the answer is "nothing", it's probably better not to enable the check.

I think it's smart to differentiate monitoring further into critical and noncritical alerts. E-mail and pager alerts should be reserved for things that require immediate action—for example, important systems that aren't pingable, full filesystems, degraded RAIDs and so on. Noncritical things, like NIS timeouts, instead should be displayed on a Web page that can be viewed when you get back from lunch. Also consider writing checks that automatically correct whatever condition they are monitoring.

Instead of your script sending you an e-mail when Apache dies, why not have it try restarting httpd automatically? If you go the auto-correcting "self-healing"

route, I'd recommend logging whatever action your script takes so you can troubleshoot the failure later.

When selecting a monitoring tool in

BLADE SERVERS

**DEDICATED SERVERS WITH
BLAZING FAST
CONNECTIONS**

Gigabit ports / MULTI-Gig options

High-capacity bandwidth plans, including:

- * 3000 GB/month for \$200
- * 5000 GB/month for \$375
- * 10000 GB/month for \$800

Custom clusters with private VLANs

Flexible storage and RAID options

Intel Premium Partner

Numerous OS choices (Linux or Windows)

FREE 24x7 "6-Star" support

www.CARI.NET/LJ
888.221.5902

carinet
Better Servers. Better Service

Monitoring Tools

If you were asked to name the first monitoring application that comes to mind, it probably would be Nagios. Used by just about everyone, Nagios is currently the king of open-source monitoring tools.

Zabbix sports a slick Web interface that is sure to make any manager happy. Zabbix scales well and might be poised to give Nagios a run for its money.

Ganglia is one of those must-have tools for Linux environments of any size. Its strengths include trending and performance monitoring.

a large environment, you have to think about scalability. I have seen both Zabbix and Nagios used to monitor in excess of 1,500 machines and implement tens of thousands of checks. Even with these tools, you might want to scale horizontally by dividing your machines into logical groups and then running a single monitoring server per group. This will increase complexity, but if done correctly, it will also prevent your monitoring infrastructure from going up in flames.

CONFIGURATION MANAGEMENT

In small environments, you can maintain Linux systems successfully without a configuration management tool. This is not the case in large environments. If you plan on running a large number of Linux systems efficiently, I strongly encourage you to consider a configuration management system. There are currently two heavyweights in this area, Cfengine and Puppet. Cfengine is a mature product that has been around for years, and it works well. The new kid on the block is Puppet, a Ruby-based tool that is quickly gaining popularity. Your configuration management tools should, obviously, allow you to add or modify system or application configuration files to a single system or groups of machines. Some examples of files you might want to manage are /etc/fstab, ntpd.conf, httpd.conf or /etc/password. Your tool also should be able to manage symlinks and software packages or any other node attributes that change frequently.

Regardless of which configuration management tool you use, it's important to implement it early. Managing Linux configurations is something that should be set up as the node is being

installed. Retrofitting configuration management on a node that is already in production can be a dangerous endeavor. Imagine pushing out an incorrect fstab or password file, and you get an idea of what can go wrong. Despite the obvious hazards of fat-fingering a configuration management tool, the benefits far outweigh the dangers. Configuration management tools provide a highly effective way of managing Linux systems and can reduce system administration overhead dramatically.

As an added bonus, configuration management systems also can be used as a system backup mechanism of sorts. Granted, you don't want to store large amounts of data in a tool like Cfengine, but in the event of system failure, using a configuration management tool in conjunction with your node installation tools should allow you to get the system into a known good state in a minimal amount of time.

PROVISIONING

Provisioning is the process of installing the operating system on a machine and

performing basic system configuration. At home, you probably boot your computer from a DVD to install the latest version of your favorite Linux distro. Can you imagine popping a DVD in and out of a data center full of systems? Not appealing. A more efficient approach is to install the OS over the network, and you typically do this with a combination of PXE and Kickstart. There are numerous tools to assist with large-scale provisioning—Cobbler and Spacewalk are two—but you may prefer to roll your own. Your provisioning tools should be tightly coupled to your configuration management system. The ultimate goal is to be able to sit at your desk, run a couple commands, and see a hundred systems appear on the network a few minutes later, fully configured and ready for production.

HARDWARE

When it's time to purchase hardware for your new Linux super cluster, there are many things to consider, especially when it comes to choosing a good vendor. When selecting vendors, be sure to understand their support offerings fully. Will they come on-site to troubleshoot issues, or do they expect you to sit for hours on the phone pulling your hair out while they plod through an endless series of troubleshooting scripts? In my experience, the best, most responsive shops have been local whitebox vendors. It doesn't matter which route you go, large corporate or whitebox vendor, but it's important to form a solid business relationship, because you're going to be interacting with each other on a regular basis.

The odds are that old hardware is more likely to fail than newer hardware.

Configuration Management Tools

Cfengine is the grandfather of configuration management systems. The project started in 1993 and continues to be actively developed. Although I personally find some aspects of Cfengine a little clunky, I've been using it successfully for many years.

Puppet is a highly regarded Ruby-based tool that should be considered by anyone considering a configuration management solution.

In my shop, we typically purchase systems with three-year support contracts and then retire the machines in year four. Sometimes we keep machines around longer and simply discard a system if it experiences any type of failure. This is particularly true in tight budget years.

Purchasing the latest, greatest hardware is always tempting, but I suggest buying widely adopted, field-tested systems. Common hardware usually means better Linux community support. When your network card starts flaking out, you're more likely to find a solution to the problem if 100,000 other Linux users also have the same NIC. In recent years, I've been very happy with the Linux compatibility and affordability of Supermicro systems. If your budget allows, consider purchasing a system with hardware RAID and redundant power supplies to minimize the number of after-hours pages. Spare systems or excess hardware capacity are a must for large shops, because the fact of the matter is regardless of the quality of hardware, systems will fail.

BACKUPS

Rethink backups. More than likely, when confronted with a large Linux deployment, you're going to be dealing with massive amounts of data. Deciding what data to back up requires careful coordination with stakeholders. Communicate with users so they understand backup limitations. Obviously, written policies are a must, but the occasional e-mail reminder is a good idea as well. As a general rule, you want to back up only absolutely essential data, such as home directories, unless requirements dictate otherwise.

SERIAL CONSOLE ACCESS

Although it may seem antiquated, do not underestimate the value of serial console access to your Linux systems. When you find yourself in a situation where you can't access a system via SSH or other remote-access protocol, a good-old serial console potentially could be a lifesaver, particularly if you manage systems in a remote data center. Equally important is the ability to power-cycle a machine remotely. Absolutely nothing is more frustrating than having to drive to the data center at 3am to push the power button on an unresponsive system.

Many hardware devices exist for

Provisioning Tools

Rocks is a Linux distribution with built-in network installation infrastructure. Rocks is great for quickly deploying large clusters of Linux servers though it can be difficult to use in mixed Linux distro environments.

Spacewalk is Red Hat's open-source systems management solution. In addition to provisioning, Spacewalk also offers system monitoring and configuration file management.

Cobbler, part of the Fedora Project, is a lightweight system installation server that works well for installing physical and virtual systems.

power-cycling systems remotely. I've had good luck with Avocent and APC products, but your mileage may vary. Going back to our "keep it simple" mantra, no matter what solution you select, try to standardize one particular brand if possible. More than likely, you're going to write a wrapper script around your power-cycling utilities, so you can do things like `powercycle node.example.com`, and having just a single hardware type keeps implementation more straightforward.

SYSTEM ADMINISTRATORS

No matter how good your tools are, a solid system administration team is essential to managing any large Linux environment effectively. The number of systems managed by my group has grown from about a dozen Linux nodes eight years ago to roughly 4,000 today. We currently operate with an approximate ratio of 500 Linux servers to every one system administrator, and we do this while maintaining a high level of user satisfaction. This simply wouldn't be possible without a skilled group of individuals.

When hiring new team members, I look for Linux professionals, not enthusiasts. What do I mean by that? Many people might view Linux as a hobby or as a source of entertainment, and that's great! But the people on my team see things a little differently. To them, Linux is an awesomely powerful tool—a giant hammer that can be used to solve massive problems. The professionals on my team are curious and always thinking about more efficient ways of doing things. In my opinion, the best large-scale sysadmin is someone who wants to automate any task that needs to be

repeated more than once, and someone who constantly thinks about the big picture, not just the single piece of the puzzle that they happen to be working on. Of course, an intimate knowledge of Linux is mandatory, as is a wide range of other computing skills.

In any large Linux shop, there is going to be a certain amount of mundane, low-level work that needs to be performed on a daily basis: rebooting hung systems, replacing failed hard drives and creating new user accounts. The majority of the time, these routine tasks are better suited to your junior admins, but it's beneficial for more senior people to be involved from time to time as they serve as a fresh set of eyes, potentially identifying areas that can streamlined or automated entirely. Senior admins should focus on improving system management efficiency, solving difficult issues and mentoring other team members.

CONCLUSION

We've touched a few of the areas that make large-scale Linux system administration challenging. Node installing, configuration management and monitoring are all particularly important, but you still need reliable hardware and great people. Managing a large environment can be nerve-racking at times, but never lose sight of the fact that ultimately, it's just a bunch of Linux boxes.■

Jason Allen is CD/SCF/FEF Department Head at Fermi National Accelerator Laboratory, which is managed by Fermi Research Alliance, LLC, under Management and Operating Contract (DE-AC02-07CH11359) with the Department of Energy. He has been working with Linux professionally for the past 12 years and maintains a system administration blog at savvysysadmin.com.

AlienVault

the Future of Security Information Management

Meet AlienVault OSSIM, a complex security system designed to make your life simpler.

JERAMIAH BOWLING

Security Information Management (SIM) systems have made many security administrators' lives easier over the years. SIMs organize an enterprise's security environment and provide a common interface to manage that environment. Many SIM products are available today that perform well in this role, but none are as ambitious as AlienVault's Open Source Security Information Management (OSSIM). With OSSIM, AlienVault has harnessed the capabilities of several popular security packages and created an "intelligence" that translates, analyzes and organizes the data in unique and customizable ways that most SIMs cannot. It uses a process called correlation to make threat judgments dynamically and report in real time on the state of risk in your environment. The end result is a design approach that makes risk management an organized and observable process that security administrators and managers alike can appreciate.

In this article, I explain the installation of an all-in-one OSSIM agent/server into a test network, add hosts, deploy a third-party agent, set up a custom security directive and take a quick tour of the built-in incident response system. In addition to the OSSIM server, I have placed a CentOS-based Apache Web server and a Windows XP workstation into the test network to observe OSSIM's interoperation with different systems and other third-party agents.

Installation

To keep deployment time to a minimum, I deployed OSSIM on a VMware-based virtual machine (VM). OSSIM is built on Debian, so you can deploy it to any hardware that Debian supports. I used the downloadable installation media from the



Figure 1. A little tough to read, but this is where everything starts.

AlienVault site in .iso form (version 2.1 at the time of this writing) and booted my VM from the media.

On bootup, you will see a rather busy and slightly difficult-to-read install screen (Figure 1). The default option is the text-based install, but by pressing the down arrow, you will see a graphical install option. Select the Text option and press Enter. If you've seen Debian install screens, the OSSIM installer will look very familiar. Set your language preferences and partition your hard drive(s). Configure your settings for Postfix if desired. Finally, set your root password, and enter a static IP address for the server when prompted. The installer will restart the



Figure 2. Main Login Screen



Figure 3. Main Dashboard

machine to complete the configuration.

Open a browser from a machine on the same network and enter the IP address of the OSSIM server in the URL field (Figure 2). Enter "admin" as the user and password to log in to the management site. Change your password under the Configuration→Users section. After logging in, the main dashboard view loads (Figure 3).

The next step is to add systems for the OSSIM server to monitor. Start by defining your local network and performing a cursory scan. On the Networks tab under Policy, click Insert New Network. Enter your LAN information in the fields provided. If you don't see a sensor listed, insert a new one using the hostname and IP address of your all-in-one OSSIM server. Leave the Nagios check box enabled, but the Nessus box unchecked (Figure 4) to reduce the time needed for the first scan. After the scan completes, several hosts should appear on the Hosts tab of the Policies section. OSSIM installs and auto-configures Nagios and ntop during installation, so you also can see basic network information by visiting the Monitors section of the management page (Figure 5). Once all hosts are found, find the CentOS Web server in the Hosts section under Policies, and modify its priority from 1 to 5 (Figure 6). You will use this later in the

article when I discuss correlation.

You now have an active OSSIM server using passive network monitors like snort, Nagios and ntop to report on your test network's activity. Next, let's add some client-based agents that feed data into the OSSIM server.

Installing the OSSEC Agent

Many client agents can communicate with OSSIM, but because of space limitations, I am covering the one I believe is the most valuable to security administrators: OSSEC. OSSEC is a freely available host intrusion detection system (HIDS) maintained by Trend Micro that performs a multitude of client security tasks, such as logging, alerting, integrity checking and rootkit detection. Additionally, a large number of OSSIM plugins for OSSEC already are installed with your server that can monitor virtually any part of a UNIX/Linux/Windows system.

First, let's install OSSEC on the CentOS Web server. Download and extract the client tar from the OSSEC Web site. If you have difficulty finding the OSSEC agent, or any other agent, links to OSSIM's supported third-party agents are available in the Tools/Downloads section of the management page. Next, run the install.sh script from the unpacked tar folder. Verify your machine information and select the agent install option. Accept the default install directory. Enter the IP address of the server (the OSSIM server). Run the integrity-check daemon and enable the rootkit-detect engine. When asked to enable active response, answer "no".

To start the agent, run:

```
/var/ossec/bin/ossec-control start
```

Now, from the CentOS Web server, ssh to the OSSIM server, and run the following command to add your client agent to the OSSEC server:

```
/var/ossec/bin/manage_agents
```

Select A to add an agent, and enter a unique name for it. Add the IP address of your CentOS Web server and give the agent a unique ID. The default ID usually

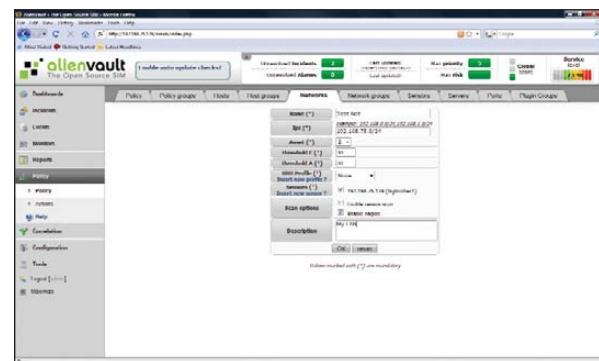


Figure 4. Setting Up the First Network Scan

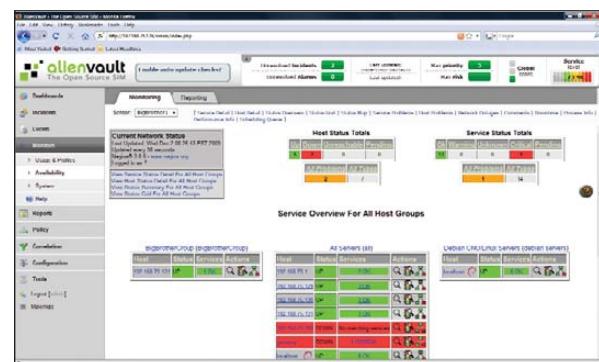


Figure 5. Nagios Working under the Hood

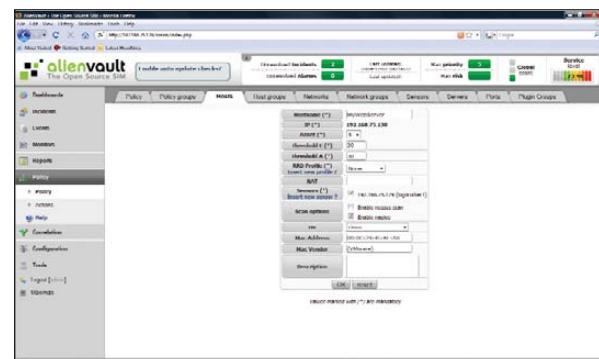


Figure 6. Changing the Web Server's Asset Value

Popular OSSIM Plugins

Some of the more popular plugins for OSSIM include the following:

- Snort
- Nagios
- OpenVAS
- Nessus
- ntop
- Nmap
- OSSEC
- Passive OS Fingerprinter (p0f)
- Osiris
- arpwatch
- syslog
- PAM
- Honeyd
- Passive Asset Detection System (pads)
- Cisco—Routers and Pix
- Multiple firewalls—iptables, sonicwall, monowall and pfSense
- Web servers—IIS and Apache
- Windows logs—Snare, OSSEC and ntsyslog
- OCS-NG—inventory software

is fine, unless you plan on implementing a naming convention for your OSSEC clients. Enter Y to confirm adding the agent. This returns you to the main menu. Select E to extract. Input the client ID you want to extract (the ID you assigned to the CentOS server). From another terminal window on the CentOS Web server, run the local manage_agents command. Select I to import the unique key. Copy and paste the unique key from the SSH window to the Web server's local prompt. Enter Y to confirm the key, and select Q to quit. Close the SSH connection, and from the local prompt, restart the agent by running the command:

```
/var/ossec/bin/ossec-control restart
```

On your XP client, download and install the OSSEC agent as well as the Putty SSH client. When finished, run the Putty client to SSH to the OSSIM server and repeat the same manage_agents command to generate and extract the XP client's unique key from the server. Once extracted, paste it into the XP client by opening the Manage Agent applet from the start menu under the OSSEC program group.

Finally, to begin receiving OSSEC events in OSSIM, open the file /etc/ossim/ossim_setup.conf on the OSSIM server and in the [sensor] section add ossec to the end of the line that begins with the word detectors. Save and exit the config file, and restart your OSSIM server using the shutdown -r now command. Upon reboot, you should start to see OSSEC events appear in OSSIM. To test this, restart the OSSEC agent on the XP machine and look in the Events→SIM Events section of the OSSIM management page. You should see messages related to the OSSEC agent (Figure

7). As you now have an external feed coming into your OSSIM server, let's look at how it digests and analyzes the data.

Events, Alarms, Directives and Correlation

For OSSIM to decipher data from any source, it first must have a plugin. A plugin is an XML-based configuration file that tells OSSIM how to read information from a particular data source and when to register a security event. According to the AlienVault site, more than 2,300 plugins currently are available (see the Popular OSSIM Plugins sidebar for a brief listing of the leading ones).

An event is any occurrence that a plugin's native software deems important enough to log or warn on. Events in OSSIM should be treated like log entries. They are not necessarily indicative of a problem, but should be reviewed nonetheless. When multiple events take place in such a way that an administrator has marked them as being "suspicious", OSSIM throws an alarm. It is also possible for a plugin to set a single event's settings high enough that it can throw an alarm when the single event occurs. The criteria used to trigger an alarm from multiple different events is known as a directive. The process of analyzing multiple events within a directive is called correlation. Correlation is central to OSSIM's operation. With correlation, administrators can take data from a multitude of disparate security devices and tailor directives to reduce false positives and extrapolate threat data in real time.

Take a typical IDS (Intrusion Detection System) device, for example. An improperly tuned IDS can record a large number of false positives. However, with OSSIM, you can create a directive that correlates your IDS

events with known vulnerabilities in Nessus. By doing so, you reduce false positives and refine questionable data into a valuable security check. As another example, you could correlate multiple port scans from Nmap with failed logins from syslog (or OSSEC, as I explain later) to detect break-ins. A third example would be to correlate

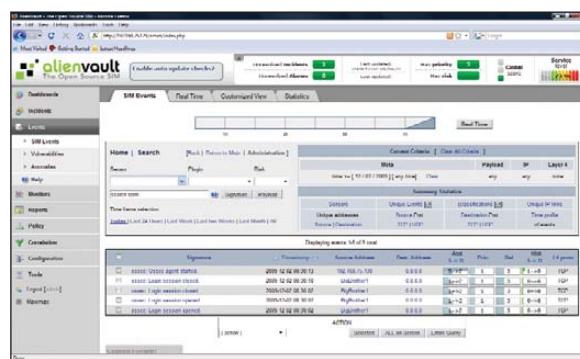


Figure 7. Verifying the OSSEC Agent Is Talking to OSSIM

aberrant network behavior using ntop with rootkit checks from OSSEC or virus detections from Sophos, ClamAV or McAfee to monitor for client-based threats. With the number of plugins available for OSSIM, the possibilities for correlation are almost limitless.

Custom Directives, Risk and Incident Response

Let's create a simple directive so you can see correlation in action. As an example, let's use a simple directive to monitor suspicious access to the Web server using two different plugins. In order to do so, first turn down the values for your OSSEC plugin. From the OSSIM management page, go to the Plugins section under Configuration. Scroll through the tables to find Plugin ID 7010, and click on the ID column to edit the plugin's values. On the resulting page, change the reliability values for the SIDs 5503 and 5716 from 5 to 1 (Figure 8). If you left these values at 5, they would send an alarm before the rule is processed.

Because the goal is to observe correlation, you need to turn them down.

Click on the Directives link found under the Correlation section of the navigation pane. From here, you get a brief description of how directives are ordered and processed. Click on the Add Directive line in the toolbar. In the resulting fields, enter "Unauthorized Access" as the Name. In the Id, enter 101, which is the value in the Generic directive. Set the Priority to 2 and click OK. Go to the next page (Figure 9), click on the plus sign symbol to add a rule directive. In the Name field,

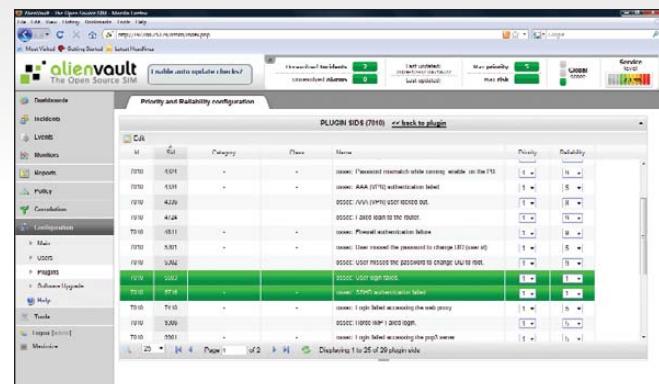


Figure 8. Adjusting the Reliability of Our Plugin's SIDs

Scan on Web Server from Foreign Host". Enter 1001 as the Plugin Id (snort). In the Plugin Sid field, type "2000537, 2000545", and under the Network section in the To field, type in the IP address of your CentOS server and the Port to List 22. In the Risk field, set Occurrence to 3, Reliability to 1. Set the Sticky field to True and Sticky Different to SRC_IP. Click the Save

ConFoo.ca

web techno conference

Work more efficiently with Web Technologies.

**All Linux Journal readers benefit
from a \$100 discount!**
Register online: <http://ConFoo.ca/lj2010>
Registration deadline: February 20th, 2010

PHP, Python, Ruby, Java, .Net,
Web Standards, Security,
Open Source, Databases,
Optimization, Web Services,
Design Patterns, RIA, Usability,
Project Management, SEO,
CMS, Frameworks, Ajax,
Testing, Social Networking

FEATURE AlienVault

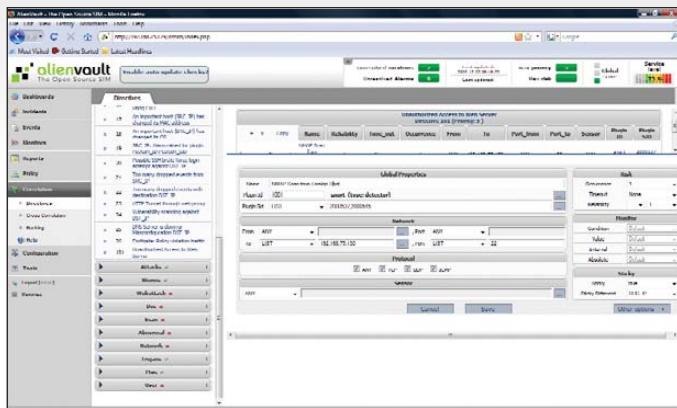


Figure 9. The First Rule of the Test Directive

button at the bottom of the page.

In theory, you have a directive that will send an alarm when a host runs an Nmap scan against port 22 on your Web server. However, you won't receive alerts yet. In order for a directive to send an alarm, the risk of the directive being tripped must be greater than 1.

Although I have not talked much about risk until now, it is integral to the function of correlation. Risk is the primary factor used by the correlation engine to determine when alarms are generated. It is calculated using a series of subjective numerical values assigned by the agents and administrators. Expressed in mathematical form, the formula for risk looks like this:

$$\text{Risk} = (\text{priority} \times \text{reliability} \times \text{asset}) / 25$$

Priority is the number OSSIM uses to prioritize rules. It is set at the Directive level. Priority can have a value of 0–5. 0 means OSSIM should ignore the alert. A value of 5 means OSSIM should treat this as a serious threat. Reliability refers to how reliable a rule is based on the chance that it's a false positive. It is set at the individual rule level and can be cumulative if there is more than one rule in a directive. Possible values for reliability are 1–10, and they equate to percentages, so 6 would mean a rule is reliable 60% of the time. Asset is the value that represents the importance of a host. You assigned the highest possible priority (5) to your CentOS server in the Policies section earlier in the article.

At this point, you have one rule under your directive, but no correlation, so you need to add another rule. Click on the + symbol on your directive. Give

the new rule a name of "Too Many Auth Failures". Set the Plugin ID to 7010 (OSSEC), and set the From field to the IP address of your Web server as the OSSEC agent will show the Web server as the source of the events. Set Occurrence to 4 and Reliability to 0 for now. Click Save. After adding the second rule, navigate to the row of the new rule and move the mouse over the directional arrows that control how rules are treated inside the directive. The up and down arrows are similar to AND statements, meaning both rules must match, and the left and right arrows nest rules within each other like nested IF statements. Move your second rule to the right. Open the second rule back up and change the reliability to +2, which will increase the reliability by 2 over the previously processed rule (3 if the first rule is met). Now, if both rules are met, the risk will be > 1 and an alarm will be generated. Listing 1 shows the directive in XML format.

To generate an alarm, log on to the XP client and download Nmap. Run four scans against the CentOS server using the zenmap GUI and the quick scan option. Then, ssh to the same server and attempt to log in as root, but

Listing 1. Directive in .xml Format

```
<directive id="101"
    name="Unauthorized Access to Web Server"
    priority="5">
    <rule type="detector"
        name="NMAP Scan from Foreign host"
        from="ANY"
        to="web.server.ip.address"
        port_from="ANY"
        port_to="22"
        reliability="1"
        occurrence="1"
        plugin_id="1001"
        plugin_sid="2000537,2000545"
        sticky="true"
        sticky_different="SRC_IP">
    </rule>
    <rules>
        <rule type="detector"
            name="Too Many Logins"
            from="web.server.ip.address"
            to="ANY"
            port_from="ANY"
            port_to="ANY"
            reliability="+2"
            occurrence="2"
            time_out="86400"
            plugin_id="7010"
            plugin_sid="5716"/>
    </rules>
    </rule>
</directive>
```

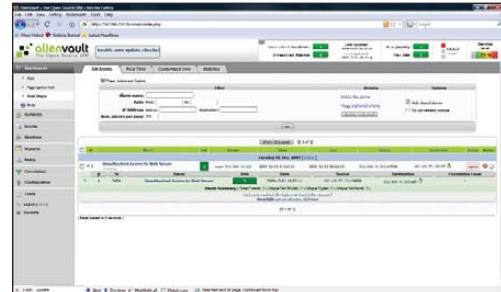


Figure 10. Test Directive Generating an Alarm

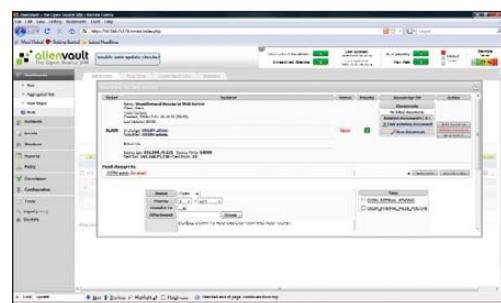


Figure 11. A New Ticket Generated by the Alarm

enter an incorrect password five times. You should see a new alarm in the Unresolved Alarms link at the top of the page. Access this link and find the alarm triggered by your test directive (Figure 10). Identify the row with your test alarm and click on the icon resembling a sheet of paper in the Action column to open a new Alarm Incident (Figure 11). A new window will pop up and display basic information about the incident that will be used to create a ticket. Click OK to confirm the information, and the full ticket editor will load. Add a description and any other pertinent information to this page, and click on the Add ticket button. You should see a new Unresolved Ticket on the indicator at the top of the page. To edit a ticket, navigate to the Tickets link in the Incidents section of the navigation pane. From here you can add notes, attach files and change the status of your tickets. A ticket will no longer show in the list once its status is set to Closed. Although quite simple, this built-in

ticketing system contains the necessary functionality to satisfy most enterprises' incident-response needs. OSSIM also contains a knowledge base that you can use to link tickets and external documents that adds another layer of depth to its incident response system.

The Sky's the Limit

This brief walk-through barely touches on the power of OSSIM. Its correlation abilities and its multitude of plugins make it an intriguing alternative to the traditional SIM. If you factor in the ability

to write your own plugins, you have a tool that is fully customizable for any environment and whose value is limited only by your creativity. The makers of OSSIM have given SIMs a new intelligence that hopefully will drive innovation in the field and take security management to the next level.■

Jeremiah Bowling has been a system administrator and network engineer for more than ten years. He works for a regional accounting and auditing firm in Hunt Valley, Maryland, and holds numerous industry certifications, including the CISSP. Your comments are welcome at jb50c@yahoo.com.

Resources

OSSIM Installer Download:

www.alienvault.com/opensourcesim.php?section=Downloads

OSSIM Wiki: www.ossim.net/wiki/doku.php

OSSEC: www.ossec.net

Expand Your Mind and Your Bottom Line

it 360°

Presenting
CIO
Thought Leader Forum

Co-located with
CLOUD COMPUTING CONFERENCE
AN IT PARADIGM SHIFT

ASTERISK OPEN TELEPHONY CONFERENCE

IT360 SILVER SPONSOR: MEUK
MEDIA PARTNER: CIO World Canada IDG CANADA
IN COOPERATION WITH: CIO Executive Council
PREMIER MEDIA SPONSORS: COMPUTERWORLD NetworkWorld THE GLOBE AND MAIL CDN
PLATINUM ASSOCIATION SPONSOR: CATAalliance
CERTIFICATION SPONSOR: Procept ASSOCIATES LTD.
CONFERENCE SPONSORS: INFO~TECH research group CIO CANADA CIO TORONTO Economic Development, Culture & Tourism

IT professionals and business executives maximize your success: www.it360.ca

METRO TORONTO CONVENTION CENTRE, CANADA
CONFERENCE & EXPO : 7 APRIL 2010

Use SSH to Cross a Suspect Host Securely

SSH tunnels can be used to connect securely to a server behind a firewall in such a way that the firewall can't observe the data. This is also true and useful if you know an attacker has gained control of the firewall.

der.hans

Recently at our local (Phoenix) Free Software Stammtisch, we were talking about security. I was surprised to find that no one else realized you can ssh safely across a compromised host to another machine behind it. A common example of "crossing" a host to get to another machine is accessing a machine on a private network protected by a firewall. The firewall is on the edge of the private network and provides the only means to get to the private network. So, you have to pass through the firewall to get to the private network.

But, what happens if the firewall has (or you believe it has) been compromised? How do you get to the private network without more security problems? It turns out that the answer is the same. You go through the firewall, but you do it in such a way that your connection remains secure even when the firewall itself may no longer be. The connection to the machine on the private network still can be made securely, even if you know the host you're passing through has been compromised and some rogue element has taken control of it.

As an example, let's say you need to connect from your Netbook to a server on your work's network. Let's also say the only way to get to your server is via a connection to your work's SSH/VPN gateway, but you think someone has broken into the gateway (yes, yes, get it off-line as soon as possible, but let's do that, and continue to be able to work and recover the gateway).

Now, let's consider an example scenario with three machines, named corresponding to what they are: Netbook,

Gateway and Server. If you already understand SSH tunneling and want the short story, see the Short Story sidebar.

For the long story, let's start with a description of some simple tunneling. The -L option on the command line allows you to create a tunnel:

```
ssh -N -f -L 2222:localhost:22 Gateway
```

The -L option allows you to specify the entry and exit points for a secure tunnel created by SSH. In the examples

SHORT STORY

The short description is that you initiate a connection to Gateway. With that connection, you create a tunnel to port 22 on Server using the -L option to ssh:

```
-L $local_port:Server:22
```

You then can connect to a local port on Netbook that is the entry point for a tunnel that comes out at port 22 of the destination machine, which is Server. The tunneled connection is never decrypted on Gateway, so it stays secure.

used in this article, it gets an argument that has three fields. The fields are separated by colons. The first field is the local port to bind (the local port to listen on). The next field is the host to connect to. This field is interpreted by the remote machine in the SSH connection, not by the local machine. The computer initiating the SSH connection doesn't need to know how to get to it. The third field is the port to connect to on the far end of the tunnel.

If you run this first command from Netbook, the command creates an SSH connection to Gateway and builds a tunnel that forwards port 2222 on Netbook to port 22 (the standard SSH port) on Gateway. The local port, 2222, can be almost any value as long as it's not already in use, although it must be above 1023 if you're not connecting as root. Now you can connect to SSH on Gateway by using that tunnel. Again, it's important to note that the tunnel destination hostname (the second field) is interpreted by Gateway, so "localhost" is instructing Gateway to connect to itself. localhost is *not* Netbook; it's Gateway. For more examples of using -L to create tunnels, see the SSH -L Examples sidebar.

The -N and -f options are useful when just creating a tunnel and not needing a remote shell. The -N option tells SSH not to run a remote command. The -f option puts the ssh command in the background, so you can regain control of the shell that ran the ssh command.

Now, run the following command on Netbook:

```
ssh -p 2222 localhost
```

This second command creates an SSH connection to port 22 on Gateway by using the tunnel created in the first command. In the second command, localhost is the destination machine, so it's interpreted by the SSH client on Netbook, which means the ssh command running on Netbook connects

SSH -L Examples

The following command connects to remote_host and builds a tunnel from port 8888 of your desktop to port 80 of my Web server. You then can point a Web browser at <http://localhost:8888/> to read my home page:

```
ssh -L 8888:www.LuftHans.com:80 remote_host
```

The following command connects to remote_host and builds a tunnel from port 9993 of your desktop to port 993 of mail_server's IMAP over SSL server. You then can configure your mail client to connect to port 9993 on your local system to read your mail:

```
ssh -L 9993:mail_server:993 remote_host
```

In both cases, the remote servers see the connection coming from remote_host, and in both cases, it doesn't matter whether your desktop can talk directly to the server at the far end of the tunnel.



Linux - FreeBSD - x86 Solaris - MS etc.



Proven technology. Proven reliability.

When you can't afford to take chances with your business data or productivity, rely on a GS-1245 Server powered by the Intel® Xeon® Processors.

Quad Core Woodcrest



2 Nodes & Up to 16 Cores - in 1U

Ideal for high density clustering in standard 1U form factor. Upto 16 Cores for high CPU needs. Easy to configure failover nodes. Features:

- 1U rack-optimized chassis (1.75in.)
- Up to 2 Quad Core Intel® Xeon® Woodcrest per Node with 1600 MHz system bus
- Up to 16 Woodcrest Cores Per 1U rackspace
- Up to 64GB DDR2.667 & 533 SDRAM Fully Buffered DIMM (FB-DIMM) Per Node
- Dual-port Gigabit Ethernet Per Node
- 2 SATA Removable HDD Per Node
- 1 (x8) PCI_Express Per Node



Servers :: Storage :: Appliances

Genstor Systems, Inc.

780 Montague Express. # 604
San Jose, CA 95131

Www.genstor.com
Email: sales@genstor.com

Phone: 1-877-25 SERVER or 1-408-383-0120



Intel®, Intel® Xeon®, Intel® Inside® are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

FEATURE Use SSH to Cross a Suspect Host Securely

to Netbook but on a nonstandard port. The tunnel is entered at port 2222 on Netbook and comes out at port 22 on Gateway. Because sshd is listening on port 22, the tunnel connects to the SSH daemon on Gateway.

Presuming Gateway can connect to Server even though Netbook can't (remember Server is "firewalled" behind Gateway), it's also possible to create a tunnel to Server using a command such as:

```
ssh -a -N -f -L 3333:Server:22 Gateway
```

In this case, the tunnel from port 3333 on Netbook doesn't connect to Gateway, rather it connects to port 22 on Server. Gateway is essentially forwarding packets from Netbook to Server. Like the first command, the tunnel destination (the second field, Server) is interpreted by Gateway, so this tunnel connects to Server. Netbook doesn't need to know how to get to Server; Gateway handles the routing.

The -a option here makes sure authentication agent

Authentication Agents

An authentication agent holds authentication credentials and allows other processes to use it for authenticating with SSH servers. It can be used for an X session allowing various commands and shells to authenticate automatically when logging in to remote services.

Authentication agents also can pass credentials on to remote servers allowing the remote shell to use it. The following command will connect to hostB and forward the authentication agent connection if you have one established:

```
ssh -A hostB
```

If hostC allows authentication with the same key, you then can ssh from hostB to hostC without having to authenticate manually.

A couple risks are involved in this type of authentication agent forwarding. In the example in the article, the risk is that if you forward the authentication agent connection to Gateway from Netbook, the attacker also could gain access to that authentication agent connection. If Server accepts the same key, the attacker could use your authentication agent connection to establish a connection to Server.

Another issue is that the authentication agent will forward all keys that it has. If you use one key for customerA and another key for customerB, you don't want the agent to forward the key for customerA to customerB's machine.

connections are not forwarded to Gateway. If you are concerned that Gateway is compromised, you don't want the attacker to gain control of your authentication agent connections. For more, see the Authentication Agents sidebar.

Figure 1 displays the example graphically. The initial ssh command builds the SSH connection for the tunnel, the tunneled connection and the forwarded connection. The second ssh command initiates a new connection through the tunnel and forwards the connection directly to the machine you are trying to reach. The forwarded connection is in red to show that it is unencrypted, but the blue SSH connection going through it is secure.

Your Machine

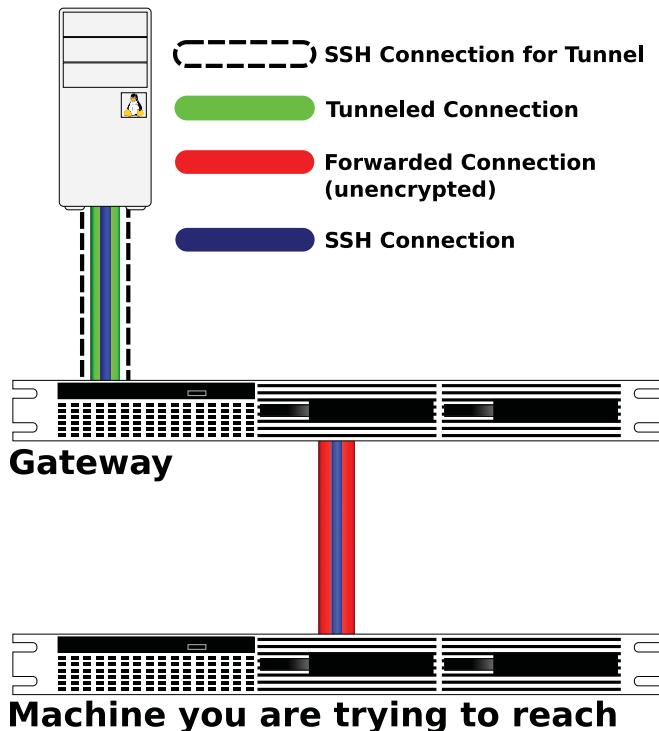


Figure 1. Tunneling across a Compromised Host (Brian Cluff, LJ2009@Macrosift.com, created the image for this article.)

This tunnel could work for any TCP protocol, but the packets from Gateway to Server and back to Gateway are unencrypted. That means unencrypted services would not be secure between Gateway and Server, even though you're using an SSH tunnel. You may want to review the -L option section of the SSH man page and consider this a bit to convince yourself that the Server/Gateway connection (the red part in Figure 1) is unencrypted. Anything that isn't secure or that gets decrypted on Gateway can be read by the attacker if the attacker has root access on Gateway. The attacker also can read much of that simply by having access to your account on Gateway even without root access on Gateway.

Before connecting to the tunnel, you need to make sure you have Server's host public key registered as a key for localhost. If you tried to pull the public key out of Gateway's known_hosts file, the attacker can give you a bogus key, so you need to get the key another way.

Therefore, it's best to have Server's public key already. If you don't already have the public key, make sure to acquire it or the fingerprint for the key from a secure, trusted source. For instance, you could list the SSH key fingerprints for all of your servers on a secure Web page.

I suggest registering Server's public key in your known_hosts file under the real server name as well as under localhost. By registering under the real server name, you can figure out to which server a key belongs with the ssh-keygen command.

The following command checks your \$HOME/.ssh/known_hosts file for a public key for Server. The command also reports the line number for the entry:

```
ssh-keygen -F Server
```

The entry then can be copied. By replacing the hostname portion of the copied entry with localhost, you create an entry for that key on localhost. See the Obtaining the Public Key sidebar for information on how to obtain a server's public key securely.

If your known_hosts file has hostnames that look like the keys, you have hashed hostnames. Use the -H option to ssh-keygen to hash your new localhost entry. The following command translates plain-text hostnames in your known_hosts file to hashed hostname entries:

Obtaining the Public Key

The public key for a host is usually in either /etc/ssh/ssh_host_rsa_key.pub or /etc/ssh/ssh_host_dsa_key.pub, depending on the type of key the host uses. You also can get the public key out of the known_hosts file of a computer you can trust. For instance, if you normally connect to the server from your work desktop, you could copy the entry for server out of your work desktop's known_hosts file.

Your site also might publish SSH public keys or public key fingerprints via a secure Web page or via DNS entries.

```
ssh-keygen -H
```

Now with the tunnel set up, you can connect to SSH on Server, and because it's an SSH connection, it's encrypted:

```
ssh -p 3333 localhost
```

Again, localhost here is from Netbook's perspective, so that command connects to port 3333 on Netbook. The last SSH

**FROM OPENNESS,
EVOLUTION**

ATTEND FLOURISH 2010
INSPIRING TALKS,
ILLUMINATING WORKSHOPS

**MARCH
19/20**

UIC STUDENT CENTER EAST
750 S. HALSTED STREET
CHICAGO, IL

FOUR YEARS OF SUPPORTING
OPEN-SOURCE IN THE MIDWEST

FLOURISH! 2010
OPEN SOURCE, OPEN FUTURE

ALWAYS FREE AS IN [SPEECH,BEER]. GET UPDATES AT [HTTP://FLOURISHCONF.COM/](http://FLOURISHCONF.COM/)

FEATURE Use SSH to Cross a Suspect Host Securely

"tunnel" command created a tunnel from that port over to the SSH port on Server via Gateway—meaning that this command uses the tunnel to connect to Server's SSH server via port 3333 on Netbook.

Even though the tunnel passes through Gateway, it is unreadable by Gateway. Just as SSH connections normally travel across untrustable routers, yet are secure, the connection through Gateway to Server is secure. The SSH connection via the tunnel is never unencrypted on Gateway, so the attacker can't compromise it. But again, remember to have verified the host key for Server already, or the attacker can compromise the connection with a man-in-the-middle attack.

With the connection to Server, you now could set up a SOCKS proxy to allow applications running on Netbook to be effectively on your work's network. See the SOCKS Proxy sidebar for more information.

If Netbook has a public IP, you also can set up a reverse tunnel from Server to Netbook. The reverse tunnel allows you to connect to Netbook from Server and then connect back to Server via the reverse tunnel. If the reverse tunnel doesn't need to go through Gateway, you then could take Gateway down for investigation and repair while still being connected to the internal network. See the Reverse SSH Tunnel sidebar for more information about reverse tunnels.

SOCKS Proxy

OpenSSH can set up a SOCKS tunnel when called with the -D option to contact services on Server.

One example would be to use the FoxyProxy add-on for Firefox to direct requests for Intranet servers to use the SOCKS tunnel. The http requests then would be sent from Server and would be able to contact Web servers on the Intranet. One such server could be a trouble-ticketing system to allow you to report that Gateway has been compromised.

The following command, when given on Netbook, would use the tunnel to Server by connecting to port 3333 on localhost and then create a SOCKS proxy on port 1080 of Netbook:

```
ssh -p 3333 -D 1080 localhost
```

FoxyProxy then could be configured to proxy Intranet requests via port 1080 on localhost.

Review

Connect to your possibly compromised machine, Gateway, and create a tunnel to the machine you ultimately want to reach, Server. Once the tunnel is up, use it to connect to SSH on Server:

```
ssh -a -N -f -L 3333:Server:22 Gateway  
ssh -P 3333 localhost
```

Reminders:

- Don't forget to block authentication agent forwarding.
- Don't forget to specify remote user names if you need them.
- Use a port above 1023 if you're not connecting as root.
- Make sure to have confirmed SSH host keys previously.
- Add the host key to your known_hosts file as an entry for localhost, because SSH will see the connection as being to localhost.
- Use -D to create a SOCKS proxy.
- None of the commands in this article require root access. All would work from your own account.
- The attacker can block the connection and can control the connection to Gateway, but the attacker can't compromise the connection to Server.

Any SSH connection across the Internet is crossing

101 Tech Tips DVD by *Linux Journal* is a series of one-minute how-to videos with shortcuts, tips and cool tricks for Linux users of all levels of expertise. Hosted by *Linux Journal* editor Shawn Powers, this DVD also features fun video extras, DRM-free copies of each tech tip for playing on your computer, as well as an entire year (2009 PDFs) of *Linux Journal* itself! Buy yours today for just \$29.95.

www.linuxjournalstore.com

Advertiser Index

Reverse SSH Tunnel

A reverse tunnel is just like the tunnel we set up with -L, except it allows the destination machine to connect to the client machine.

In the example from this article, you can create a reverse tunnel from Server to Netbook allowing Netbook to reconnect to Server. The following command, when run from Server, connects to Netbook and creates a tunnel from port 4444 on Netbook to the SSH daemon on Server. Any shell on Netbook then can connect to Server via port 4444:

```
ssh -R 4444:localhost:22 Netbook
```

The following command, when run from Netbook, would connect to Server via the reverse tunnel:

```
ssh -p 4444 localhost
```

If the outbound connections from Server don't go out via Gateway, the reverse tunnel can be used even after Gateway is shut down.

I suggest running the SSH command from within a screen session on Server to make sure the controlling shell doesn't exit.

questionable hosts/routers. If those were safe, you wouldn't need a secured connection. The tunnel is the same scenario, because it's just enabling a normal SSH connection. ■

der.hans is Vice President of Engineering at a startup in stealth-mode, cofounder of TwoGeekTechs, founder of the Free Software Stammtisch, an adjunct instructor at a local community college, Chairman of the Phoenix Linux Users Group, strategic advisor to TEDxPhoenix and founding member of League of Professional System Administrators (LOPSA). In his free time, he likes to go camping, annually not set himself on fire and brag about his grandma's cheesecakes and Spätzle. He can be reached via Commerz+LJ@LuftHans.com.

Resources

OpenSSH: www.OpenSSH.org

OpenSSH Manual Pages:
www.OpenSSH.com/manual.html

Free Software Stammtisch: www.LuftHans.com/
[Free_Software_Stammtisch](#)

FoxyProxy: <https://addons.mozilla.org/de/firefox/addon/15023?src=api>

CHECK OUT OUR BUYER'S GUIDE ON-LINE.
Go to www.linuxjournal.com/buyersguide where you can learn more about our advertisers or link directly to their Web sites.

Thank you as always for supporting our advertisers by buying their products!

Advertiser	Page #	Advertiser	Page #
1&1 INTERNET, INC. www.oneandone.com	1	IXSYSTEMS, INC. www.ixsystems.com	C3
ABERDEEN, LLC www.aberdeencinc.com	5	LINUX FEST NORTHWEST www.linuxfestnorthwest.org	7
CARI.NET www.cari.net	51	LOGIC SUPPLY, INC. www.logicsupply.com	37
CONF00 www.conf00.ca	57	MICROWAY, INC. www.microway.com	C4, 3
DEVICE SOFTWARE ORGANIZATION, LLC www.devicesoft.org	79	MIKROTIK www.routerboard.com	C2
DIGI-KEY CORPORATION www.digi-key.com	78	POLYWELL COMPUTERS, INC. www.polywell.com	78, 79
DRUPALCONF sf2010.drupal.org	73	POSSCON www.posscon.org	71
EMAC, INC. www.emacinc.com	23	SERVERS DIRECT www.serversdirect.com	9
EMPEROR LINUX www.emperorlinux.com	27	SHARE.ORG www.share.org	69
FLOURISH flourishconf.com	63	SILICON MECHANICS www.siliconmechanics.com	21, 45
FOURTH GENERATION SOFTWARE SOLUTIONS www.fourthgeneration.com	78	SXSW FESTIVALS AND CONFERENCES www.sxsw.com	39
GECAD TECHNOLOGIES/AXIGEN www.axigen.com	78	TECHNologic SYSTEMS www.embeddedx86.com	43
GENSTOR SYSTEMS, INC. www.genstor.com	61	TRUSTED COMPUTER SOLUTIONS www.TrustedCS.com/SecurityBlanket	13, 79
GUTSY GEEKS www.gutsygeeks.com	77	TUXERA LTD. tuxera.com	79
IT360 www.it360.ca	59	USENIX ASSOCIATION www.usenix.org/nsdi10/lja	29

ATTENTION ADVERTISERS

June 2010 Issue #194 Deadlines
Space Close: March 22; Material Close: March 30

Theme: Distributions

BONUS DISTRIBUTIONS: O'Reilly's Web 2.0 Expo, Linux and Open Source on Wall Street, VMworld

Call Joseph Krack to reserve your space
+1-713-344-1956 ext. 118, e-mail joseph@linuxjournal.com

Using an SMS Server to Provide a Robust Alerting Service for Nagios

How to implement a Nagios-to-SMS service.

ERIC PEARCE

I'm a big fan of the Nagios network monitoring system and rely on it to tell me if something goes wrong with the systems for which I am responsible. I have made a large investment in time configuring Nagios to monitor exactly what I am interested in, and this effort would be wasted if Nagios detected a problem, but failed to communicate that problem to me. To make Nagios more robust, I wanted to make sure that its alerting mechanism did not depend on connections to the Internet—this would include the physical connection itself and internal and external services, such as e-mail, routing and DNS.

I have relied on e-mail-based systems in the past to deliver alerts; however, my dilemma was that if I was not getting e-mail, I did not know if this meant everything was okay or if there was some problem preventing me from getting the e-mail alerts, such as a down Internet connection or another

kind of e-mail failure. I found that I became uneasy after long periods of silence and felt compelled to "poll" the system to make sure everything was okay.

On the other hand, I felt that if my alerting system was robust and I could trust it, my thinking would become "no news is good news", and the absence of alerts would mean everything was fine.

I've found that the Short Message Service (SMS) text service available on GSM cellular networks meets my requirements for a trusted alerting server. It is generally available and is unlikely to go down. A major disaster certainly could take down or overwhelm the cellular service, but I figure I would be aware of such an event and probably would have bigger and more pressing concerns than network management at that point.

There are several different ways to implement a Nagios-to-SMS service, and I certainly have not explored them all. This



Figure 1. MultiTech Systems MultiModem iSMS Intelligent SMS Server

article describes the system I am using, which is the MultiTech Systems MultiModem iSMS Intelligent SMS server (Figure 1) in combination with a public domain Perl script running on a Linux-based Nagios server.

I selected this hardware and software combination for the following reasons:

- Another company had done all the required work to integrate the iSMS device with Nagios, clearly documented the process and made this freely available on the Web, including the Perl script described in this article.
- A major feature of the Perl script is the ability to "ACK" or acknowledge a Nagios alert. This means you don't have to have any kind of IP connection to your Nagios server to perform acknowledgements. The ability to acknowledge alerts is helpful when you are off the IP network, as it stops any future alerts and can prevent the alerts from going to others if you have configured Nagios to do this. The script also can force a service or host back to an "OK" state if desired.
- The iSMS device is a standalone "appliance" and does not depend on any infrastructure other than a (local) Ethernet connection, GSM cellular service and electrical power. Most other products in this area are similar to traditional analog modems in that they have serial connections hard-wired to a specific host. As the iSMS is connected via Ethernet, it can be accessed and shared by multiple hosts. The particular model I used has a single GSM modem, but four- or eight-modem versions are also available.
- Other Nagios users are using conventional mobile phone handsets in this role, but I feel that consumer-level power supplies and some kind of jury-rigged mounting of a phone in a machine room would undermine the reliability I want. The iSMS has a robust metal case and can be attached securely to a rack. The power plug is threaded to the chassis to prevent accidental unplugging.
- The iSMS has a Web-based administration interface and

supports multiple methods of communication, including a "Telnet" interface to connect directly to the GSM modem for use of "AT" commands and multiple APIs. These include both TCP and HTTP APIs for sending and receiving SMS messages or querying the status of queued messages. Certainly, you could use Web-based or e-mail-based tools to create a similar alerting functionality, but SMS is somewhat unique in that it does not require an IP connection and is generally available wherever a modern cellular infrastructure exists.

As you can see in Figure 1, the iSMS is packaged in a sturdy metal enclosure. I used large plastic wire ties to mount the iSMS to a horizontal rack post, but it also can be mounted with screws. The antenna is visible on the top, and there is a little hatch on the bottom where the Subscriber Identity Module (SIM) card is plugged in.

Ordering the iSMS

The product is generally available, and I simply used a price comparison site to find the cheapest one, as I didn't feel I needed support from the vendor. MultiTech made several changes to its product while I was in the midst of writing this article. These changes included renaming the product, updating the firmware version and lowering the price. The iSMS previously was named SMSFinder, and you will see this reflected in the name of the Perl script and in other places. The firmware update required some changes in the Perl code. The original product was priced around \$700, but it's now available in the \$400 range. This article describes the more recent version of the iSMS and Perl script.

Ordering SMS Service

I did not shop around between the different carriers for SMS service, as my company already had a corporate account with AT&T. I initially tried to walk into an AT&T retail store to buy service for the iSMS, but I was unable to purchase a service package that did not include voice. I ended up doing all the ordering and setup over the phone with AT&T corporate. I was able to get the SIM card at the retail store, which saved me from having to wait for the card to be mailed to my location. AT&T calls its text-only SMS service telemetry. It may make the ordering process easier if you use this terminology with your carrier.

Once you reach the correct ordering department, all you should need to do is read them two numbers: the first identifies the iSMS, and the second identifies the SIM card. The number for the iSMS is the International Mobile Equipment Identity (IMEI) number printed on the iSMS chassis label. The second number, the Integrated Circuit Card ID (ICC-ID) is printed on the SIM card. Once I had communicated these numbers to the carrier, I was able to establish the service and send a test message within a few minutes. Make sure you make note of the subscriber number given to you, as this will be the source of SMS alerts and the destination for your "ACK" and "OK" responses. It is handy to associate a contact name with this number for caller-ID purposes on your mobile phone (for example, "Nagios"). With the service I purchased, the one-time setup fee was \$18, and the monthly charge is about \$9, depending on usage.

FEATURE Using an SMS Server to Provide a Robust Alerting Service for Nagios

Physical Location of iSMS

I wanted the iSMS and the Nagios server to be able to send messages for as long as possible if there were a problem with network connectivity or power. In my situation, this meant locating the iSMS, the Nagios server and their shared Ethernet switch all in the same computer room and plugged in to the same redundant UPS. Of course, I could eliminate the switch from this configuration by using an Ethernet crossover cable to link the iSMS and the Nagios server directly, but that would limit communication to the one server. It also would eliminate most of the advantages over the hard-wired GSM modems that I was trying to improve upon.

The smsfinder Perl Script

The Perl script can be found on the MonitoringExchange site (www.monitoringexchange.org) or the Nagios Wiki (www.nagioswiki.org). Searching for "smsfinder" should get you to the correct place. The documentation for the script includes installation instructions, example Nagios configuration files and screenshots of the iSMS Web user interface. The author used an interesting approach to create a single script that serves three different purposes. The script checks to see which filename was used to call the script and then performs a completely different function depending on which name was used. The script has three names:

- smssend.pl: a Nagios "command" used to send messages about hosts and services via SMS.
- smsack.cgi: a CGI script used by the iSMS to acknowledge alerts it has received via an SMS message sent by a mobile phone.
- check_smsfinder.pl: a typical Nagios "plugin" or "check" script invoked by Nagios on a scheduled basis to monitor the health of the iSMS device itself.

Script Installation

I stored the actual script as /usr/local/nagios/smsack/smsack.cgi and made two symbolic links to it with the following names/paths:

- /usr/local/nagios/smsack/sendsms.pl
- /usr/local/nagios/libexec/check_smsfinder.pl

Apache will want to execute an actual file as a CGI, while Nagios will not care about the symbolic links. Figure 2 provides an overview of how the script interacts with the other components of the system.

The least complex use of the script is when it is called check_smsfinder.pl. Nagios runs this check script at scheduled intervals, and the script queries a status page on the iSMS via HTTP to make sure it is running okay. The script returns the exit status back to Nagios. The performance data includes the signal level for the GSM modem, the model number and the firmware version.

The next use of the script is when it is called as sendsms.pl by Nagios. In this form, it is used to send host and service alerts and acknowledgements to the iSMS for delivery to mobile users. The script uses the "HTTP Send API" to request

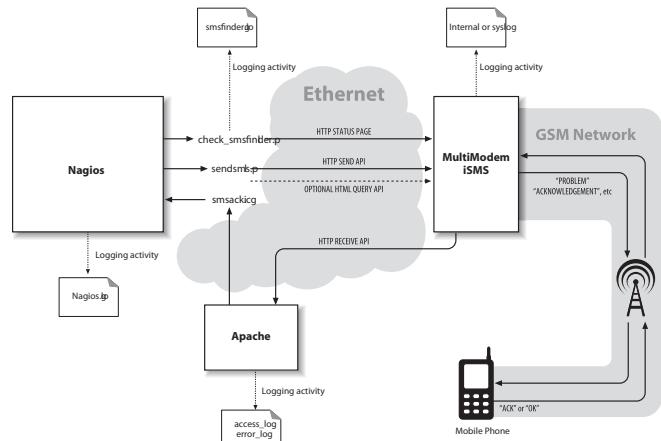


Figure 2. SMSFinder Method of Operation

that it transmits the SMS message. The iSMS queues the request and returns a message ID. The end user can query the iSMS with the message ID to find out if the SMS message was sent successfully or failed for some reason. When invoked as sendsms.pl, the script has a --noma option, which will query the iSMS after queuing the message to get its status. This takes slightly longer to execute, as the script has to wait for confirmation that the message actually was sent (or failed). The documentation refers to "NoMa" but does not explain why the option was named that way.

The most complicated use of the script is when it runs as the smsack.cgi CGI script under Apache. The recipient of an SMS alert can send the entire message back to the iSMS with the string "ACK" or "OK" prepended to the message text. When this SMS message is received by the iSMS, it uses the "HTTP Receive API" to call the smsack.cgi CGI script with the ACK message. The smsack.cgi CGI script parses the message text, determines whether it is a host or service being acknowledged, verifies that the sender's phone number is in the Nagios object cache and then uses the Nagios "external command" interface to signal Nagios. Nagios then tries to match the host or service name with one it knows about, and if this match is successful, it acknowledges the problem. The script also creates a note on the host or service page indicating that the problem was acknowledged by the sender's mobile number.

The acknowledge feature expects the entire SMS alert message to be sent back to the iSMS with the



Figure 3. Apple iPhone Screenshot

"ACK" text prepended. I found the best way to do this on both older text-based and newer graphical mobile phones was to "forward" the entire message received from Nagios back to the Nagios phone number and then insert the ACK text at the beginning of the message.

Figure 3 shows a screenshot from an Apple iPhone. The initial "PROBLEM" alert received from the iSMS is at the top (shown in gray). The message forwarded back to the iSMS with the prepended "ACK" is in the middle (shown in green), and the receipt of the acknowledgment sent from the iSMS is at the bottom (shown in gray). This entire transaction can be accomplished in less than a minute.

Debugging Installation and Runtime Problems

I was able to get everything running in a day or two, but I did have to resolve several issues as part of the installation. I also discovered several problems that required changes to the Perl script. Therefore, it's important to test the scripts.

You can run the check_smsfinder.pl and sendsms.pl scripts on the command line to view their output directly. For example:

```
% /usr/local/nagios/libexec/check_smsfinder.pl \
-H 192.168.1.50 -u nagios -p secret
OK: GSM signal strength is 100.0% - \
model: SF100-G - \
firmware: 1.31|loginID=1607132337 strength=100.0%;40;20;;
```

```
% /usr/local/nagios/smsack/sendsms.pl \
--noma -H 192.168.1.50 -u nagios -p secret \
-n 14155551212 -m 'this is a SMS from nagios'
>this%20is%20a%20SMS%20from%20nagios" to 14155551212 \
via 192.168.1.50 send successfully. MessageID: 37
```

The smsack.cgi script is a little harder to debug than the command-line scripts, but the usual Apache log files access_log and error_log are useful in that they will contain the HTTP response codes when the CGI is invoked by the iSMS. You also can use the method described below under "Network Capture" to look for problems with the CGI script.

Logging

In many places within Nagios, the Perl script and the iSMS device contain debugging information. Knowing where those are will help you with your installation.

The iSMS can send helpful debugging messages to a remote host via syslog. The Nagios server would be an ideal destination for the messages, as all logging can be consolidated in once place. The remote syslog host is specified in the iSMS Web GUI. The iSMS syslog messages use the LOG_LOCAL0 facility. I added a local0.* /var/log/isms entry to my /etc/syslog.conf file to capture all messages. The log file will record all SMS messages sent and received by the iSMS,

SHARE in Seattle

March 14–18, 2010 | Washington State Convention and Trade Center | Seattle, Washington

Hot topics at SHARE in Seattle

In Seattle, SHARE will focus on the following to assist your organization's engagement in these industry hot topics.

- **Enterprise Virtualization & Cloud Computing**
- **IT & the Bottom Line**

Here's why SHARE is one of the most valuable enterprise IT technical events:*

- **93% of attendees** would recommend attending SHARE to a friend or colleague
- **85% of attendees** stated SHARE's technical program provided solutions to their current business problems.
- **92% of attendees** rated their return on time invested as "excellent" or "good" for technical knowledge gained

*Feedback received from the SHARE in Denver conference evaluation, August 2009.



Register at seattle.share.org/2010

FEATURE Using an SMS Server to Provide a Robust Alerting Service for Nagios

for example:

```
Nov 23 09:27:59 smsgw MultiModemISMS modem: sentlog:  
[SENT TO] : 14155551212 : [MSG] : this is a SMS from Nagios
```

The log also contains any authentication failures. This is useful because the check_smsfinder.pl and sendsms.pl scripts authenticate themselves to the iSMS every time they run.

The iSMS has a concept of an "Inbox" for SMS messages received from mobile users and an "Outbox" for SMS messages being sent out from the iSMS. You can examine these boxes via the iSMS Web interface to find out whether a message actually was received or transmitted.

Nagios logs to the file nagios.log, which is typically found in the /usr/local/nagios/var directory. You can use this log to verify that Nagios is generating an alert for a problem and that a command has been used to send an SMS (notify-host-by-sms):

```
[1258664139] HOST NOTIFICATION:  
 epearce-sms;mailserv2;DOWN;notify-host-by-sms;CRITICAL -  
 Host Unreachable (192.168.1.250)
```

The Nagios log also will show the results of smsack.cgi running after getting the "ACK" back from a mobile user:

```
[1258500602] EXTERNAL COMMAND:  
 ACKNOWLEDGE_HOST_PROBLEM;mailserv2;1;1;1;14155551212;  
 Acknowledged by 14155551212 at 09/11/17 15:29:57  
 ACK PROBLEM mailserv2> is DOWN /11-17-2009 15:28:21/ CRITICAL -  
 Host Unreachable(192.168.1.250)
```

The smsfinder scripts log to smsfinder.log (also in the Nagios var directory). This file will contain debugging information for the sendsms.pl and smsack.cgi uses of the script. The lines containing "SMSsend" show the status of sendsms.pl when it is being run by Nagios. For example:

```
2009/11/19 12:55:39 SMSsend:  
 "PROBLEM...mailserv...is...DOWN...CRITICAL..."  
 to 14155551212 via 192.168.1.250 queued successfully.  
 MessageID: 14
```

Lines containing "SMSreceived" and "SMSverify" will show the progress in parsing any acknowledgement SMS messages received by the smsack.cgi script:

```
2009/11/12 09:15:41 SMSreceived:  
 username=nagios&password=secret&XMLDATA=  
<?xml version="1.0" encoding="ISO-8859-1"?>  
<Message Notification>  
<SenderNumber>14155551212</SenderNumber>  
<Message>  
 ACK PROBLEM HostAlert mailserv2 192.168.1.250  
 /AllServices is DOWN  
 /11-12-2009 09:11:46/ CRITICAL -  
 Host Unreachable (192.168.1.250)  
</Message>  
<Date>09/11/12</Date>  
<Time>09:15:36</Time>  
</Message Notification>
```

```
2009/11/12 09:15:41 SMSverify  
 status = ACKed - ACCEPTED:  
 From=14155551212 Received=09/11/12 09:15:36  
 Status=ACK Host=mailserv2 Service=AllServices  
 MSG="ACK PROBLEM ... Host Unreachable (192.168.1.250)"
```

Apache

I initially had multiple overlapping "Directory" statements in the Nagios section of the Apache configuration file. The net effect was a "Permission denied" when the CGI was being run. I figured this out by using the method described below and by looking at the Apache access_log and error_log files.

Network Capture

If you think there is some communication problem with the script, you can monitor the traffic between Nagios, Apache and the iSMS by listening on the network. I used tcpdump to capture the HTTP traffic and see error messages:

```
% tcpdump -v -s 0 -w /tmp/cap host 192.168.1.50
```

In this example, I used the -v option for verbose output, the -s 0 option to capture as much of the packet as possible and the -w option to write the captured traffic to the /tmp/cap file. The "host" keyword indicates that I want all traffic to and from the IP address of the iSMS (192.168.1.50). I ran this command on the machine hosting both Nagios and Apache, so it should see all communication between these services and the iSMS. I then generated some SMS messages traffic by causing Nagios to send out a "PROBLEM" message, which I then acknowledged via my mobile phone. You should see the number following "Got" incrementing as packets are being captured:

```
tcpdump: listening on eth0, link-type EN10MB (Ethernet),  
 capture size 65535 bytes  
Got 22
```

I then interrupted the capture and converted the captured data to plain text:

```
% tcpdump -A -r /tmp/cap > /tmp/txt
```

The -A option writes out ASCII text, and the -r option reads capture data from a file. Examining the /tmp/txt file allows you to see the entire HTTP transaction between Nagios, the iSMS and the CGI script:

```
12:50:09.434851 IP nagios.46058 > smsgw.http:  
 P 1:266(265) ack 1 win 46  
<nop,nop,timestamp 2801435752 1987587011>  
  
GET /sendmsg?user=nagios&passwd=secret...text=ACKNOWLEDGEMENT...  
  
12:50:09.501524 IP smsgw.http > nagios.46058:  
 P 1:29(28) ack 266 win 6432  
<nop,nop,timestamp 1987587017 2801435752>  
  
HTTP/1.0 200 OK
```

ID: 2078

In this capture, you can see that the sendsms.pl script invoked by Nagios (hostname nagios) has sent an HTTP GET to the iSMS (hostname smsgw) containing the Nagios "ACKNOWLEDGEMENT" message. The "ID: 2078" response from the iSMS back to Nagios indicates that the message has been queued for sending and that the ID for this SMS message is 2078. You also might note that the user name and password for the iSMS "nagios" account is being sent in the clear—not perfect, but I think this is a pretty low security risk, as this transaction is internal to the company network.

Firmware Version of iSMS

My original iSMS came with version 1.20 firmware. This worked fine with the original Perl script, but it had a problem in that it was somewhat "single user". For example, if you happened to be logged in to the iSMS Web user interface while the check_smsfinder.pl script ran, it would return a bad status, and Nagios would create an alert for the device. Upgrading to the newer firmware fixed this problem, but broke the check_smsfinder.pl script. The Perl script has been updated, but the version of the script now is tied to the firmware version running on the iSMS.

Because this is a Perl script, it can be modified easily. If you do not like the format of the message being sent out by

Nagios, you can change this in the Nagios "command" definition—for example, "notify-host-by-sms" and also change the Perl script to parse whatever format of a message you want to send back from your phone. The script authors have changed their message format over time to make it easier to parse, as problems were discovered with whitespace in service names and host alerts that would change format depending on whether the host definition contained an IP address (such as in the case of DHCP clients).

Conclusion

I am very pleased with how this alerting service has worked out. The iSMS has been solid since the moment I installed it, and the associated script has worked perfectly once I tweaked my Nagios setup to match it. I have high confidence that I will get alerts regardless of the nature of the problem.

Acknowledgements

Thanks to Birger Schmidt and his colleagues from NETWAYS GmbH (www.netways.de) for writing the original script, updating it and taking the time to review this article, and to Chris Reilley (www.reilleydesign.com) for Figure 2.■

Eric Pearce is the IT Lead for AmberPoint, Inc., an Application Management and Governance software company based in Oakland, California. He has authored several books on UNIX and Windows system administration for O'Reilly & Associates.

3RD ANNUAL
**PALMETTO OPEN
SOURCE SOFTWARE
CONFERENCE**
COLUMBIA, SOUTH CAROLINA

columbia metropolitan convention center

**FOR MORE INFORMATION
AND TO REGISTER, GO TO
WWW.POSSCON.ORG**

presented by
CONSORTIUM |
**FOR ENTERPRISE
SYSTEMS MANAGEMENT**

POSSCON
PALMETTO OPEN SOURCE SOFTWARE CONFERENCE

**APRIL 15, 16 & 17
2010**
at the
COLUMBIA CONVENTION CENTER

**PLEASE JOIN US AS TOP
F/OSS SPEAKERS
DISCUSS TODAY'S HOTTEST OPEN
SOURCE TOPICS.**

**SO POPULAR WE HAD TO
MOVE IT TO THE COLUMBIA
CONVENTION CENTER!**

**WITHIN EASY DRIVING
DISTANCE OF MOST MAJOR
CITIES IN THE SOUTHEAST.**

Running Ubuntu 9.10 under Amazon's Elastic Cloud

Put your servers in the cloud with Amazon EC2 and Ubuntu. BILL CHILDERS

Cloud services are all the rage today, although some of my fellow *Linux Journal* staffers may scoff when they hear me say that. Cloud services is a nebulous term that can mean anything from completely hosted services (like Gmail) to virtualized, leased servers, such as those provided by Amazon's EC2 service. And, the latter is the subject of this article. Recently, with the advent of Ubuntu 9.10 (Karmic Koala), Canonical has added support for pre-baked Amazon EC2 images. This makes spinning up your own personal cloud servers fast and easy—although not necessarily economical (see the Amazon EC2 Economics sidebar for a quick cost breakdown of EC2).

The Ubuntu EC2 Starters Guide (see Resources) should be your first stop. This document guides you through the process of creating your own EC2 instance. Before you can do anything at all with EC2, you need to set up an Amazon EC2 account. Go to the URL listed in the Resources section,

Because the Ubuntu EC2 images are publicly available, you need to generate an SSH key to access them.

and either sign in with your existing Amazon account or create a new one. Then, click on the EC2 link and sign up for the EC2 service. You need to provide a credit card for billing purposes. Once you've done that, select the Create an X.509 Certificate link that's presented on the thank-you page. Select yes to create a new certificate, and then download your certificate and private key to your hard disk. Finally, make a note of your AWS account ID number, as you may need it later.

Now that you've got your Amazon account and are enrolled in the EC2 program, it's time to start installing the management tools on your local system. If you're running Ubuntu, simply type `sudo apt-get install ec2-api-tools` to download the EC2 management tools. You also need the Sun 1.6 JDK (installable via `sudo apt-get install sun-java6-jdk`). If you're using another distro or Mac OS X, you can get the toolset by following the link in the Resources section of this article. The tools are nothing more than a .zip file full of Java binaries and shell scripts, so they're fairly portable.

Now that you've got the tools, the next step is to edit your environment variables. Simply adding the following lines to

your `~/.bashrc` file will do the trick (make sure to edit the path and filename to suit your own setup):

```
export EC2_PRIVATE_KEY=$HOME/<path-to-your-private-key>/pk-XX.pem
export EC2_CERT=$HOME/<path-to-your-certificate>/cert-XX.pem
export JAVA_HOME=/usr/lib/jvm/java-6-sun/jre
```

To test the tools, run the following from a new terminal:

```
ec2-describe-images -o self -o amazon
```

You should receive a list of all the publicly available EC2 images published by Amazon.

Because the Ubuntu EC2 images are publicly available, you need to generate an SSH key to access them. Do this on your local machine by running the command:

```
ec2-add-keypair ec2-keypair > ec2-keypair.pem
```

Make sure the permissions on the file are read-write for your user, with no access for anyone else (`chmod 600 ec2-keypair.pem`).

Now that you've finished the groundwork, you're ready to start up your first Ubuntu 9.10 instance in the cloud. The `ec2-run-instances` command will start your instance for you, but you need to feed it an AMI ID and your keypair. The AMI ID is nothing more than a unique identifier that is associated with a published EC2 image. Because I want to instantiate the "small" Ubuntu 9.10 EC2 image for this example, I use the "ami-52be5d3b" AMI ID. The links to publicly available Ubuntu 9.10 images and their AMI IDs are available in the Resources section of this article. Once you run the `ec2-run-instances` command, you will see output similar to the following:

```
$ ec2-run-instances ami-ab15f6c2 \
    --key ec2-keypair --instance-type m1.small
RESERVATION r-d8b376b0 748502897232 default
INSTANCE   i-bc9913d4 ami-ab15f6c2 \
            pending   ec2-keypair 0 \
            m1.small  2009-11-02T22:23:12+0000 \
            us-east-1d aki-76be5d1f ari-4cbe5d25
```

Instances may take a few minutes to start up (shown by the "pending" status in the output above), particularly if this is your first one. To check on the status of your instance, you can run the `ec2-describe-instances`

Amazon EC2 Economics

Amazon's servers are priced on an hourly basis, for every hour the instance is running. The baseline server, otherwise known as the "small" configuration, is basically a Xen virtual machine with 1.7GB of RAM, one core of CPU measuring one EC2 unit (approximately a 1.7GHz CPU) and 160GB of storage. The "large" configuration is 7.5GB of RAM, two cores with two EC2 units each and 850GB of storage. The small instance is \$0.085 per hour, and the large instance is \$0.34 per hour. Amazon also charges for data sent and received. Incoming data is billed at \$0.10 per GB, and outgoing data is billed at

\$0.17 per GB. Larger plans are available for more compute-intensive applications, but the cost for those is even higher. This means that the average small instance probably will run around \$70 per month to operate, and the large instance will run somewhere around \$250 per month. Of course, this also depends on the amount of bandwidth consumed. Although the monthly costs are a little on the pricey side, there are no setup fees, equipment maintenance costs or capital expenses needed to run this solution. EC2 isn't a one-size-fits-all solution, but it may make sense for some applications.

command. When your instance is running, you'll see something similar to the following:

```
$ ec2-describe-instances
RESERVATION r-d8b376b0 748502897232 default
INSTANCE   i-bc9913d4 ami-ab15f6c2 \
            ec2-72-44-62-167.compute-1.amazonaws.com \
```

```
domU-12-31-39-00-ED-A3.compute-1.internal \
running      ec2-keypair 0 \
m1.small    2009-11-02T22:23:12+0000 \
us-east-1d  aki-76be5d1f  ari-4cbe5d25
```

The "running" tag in the output denotes that the instance is up and running. Also listed is the Internet-facing DNS name as well

The Moscone Center

Join us for DrupalCon. Learn about how open source is reshaping the internet and meet the community that is leading the way.

Visit us online: sf2010.drupal.org



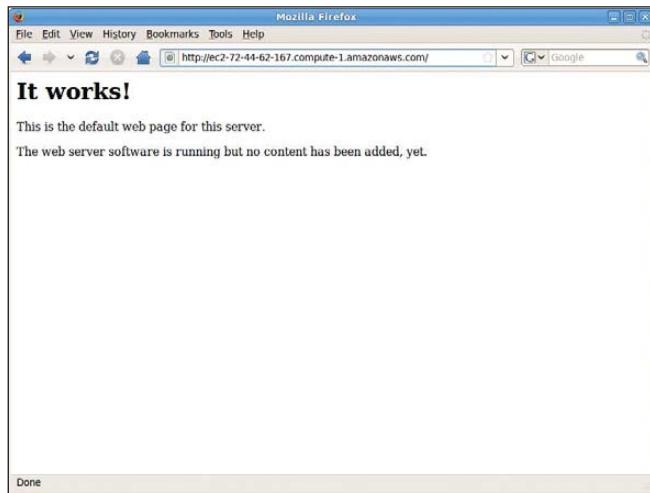


Figure 1. My Web server's in the cloud!

as the internally facing Amazon DNS name. The default image has SSH running, but the EC2 firewall rules deny inbound port 22 access. To enable SSH access to the EC2 instance, you need to run `ec2-authorize default -p 22`. Once that is done, you can ssh to your instance using the SSH keys generated earlier:

```
$ ec2-authorize default -p 22
GROUP      default
PERMISSION default  ALLOWS  tcp  22  22  FROM  CIDR  0.0.0.0/0

$ ssh -i ~/.ec2/ec2-keypair.pem \
      ubuntu@ec2-72-44-62-167.compute-1.amazonaws.com
Linux domU-12-31-39-00-ED-A3 2.6.31-300-ec2 #3-Ubuntu SMP \
      Sat Sep 26 10:31:44 UTC 2009 1686
```

To access official Ubuntu documentation, please visit:
<http://help.ubuntu.com/>

```
System information as of Mon Nov  2 22:45:44 UTC 2009

System load: 0.0          Memory usage: 1%  Processes:      56
Usage of /:  7.6% of 9.92GB Swap usage:  0%  Users logged in: 0
...
ubuntu@domU-12-31-39-00-ED-A3:~$
```

EC2 instances can be a great way to extend your own computing platform, provide capacity on demand to a service or enable you to spin up a replacement server.

How Can I Figure My Costs, Exactly?

It's rather tough to nail down cost for an EC2 instance, as it depends on things like bandwidth. However, there is a small dashboard-like application you can install that can help you track your costs. The package byobu (formerly known as screen-profiles) can help. After apt-getting byobu and running byobu, you'll get a screen session with a small two-line dashboard at the bottom of your terminal window. To add the EC2 information, press the F9 key to bring up the byobu menu, select Toggle Status Notifications, then select the `ec2_cost` notifier, and press Apply. Then you'll have a neat little running total of the approximate cost for your EC2 instance, as shown in Figure 2. You can see this particular run cost me 40 cents! Byobu has all kinds of other useful little widgets too; check it out if you do a lot of management via a terminal.

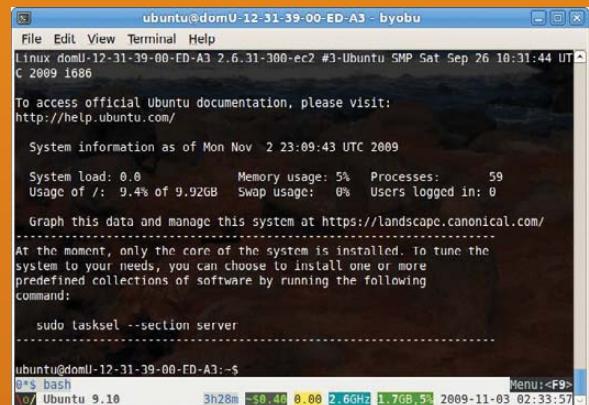


Figure 2. Pennies Here, Pennies There.

At this point, your instance is ready for you to start configuring whatever software you choose to run on it. Because it's essentially an Ubuntu machine, administration and package management is done just like on the systems you're used to (assuming you use Ubuntu). You simply can use `apt-get`.

As an example, I'm going to get a quick Apache server running. First, I update the apt indexes on the instance by running `sudo apt-get update`. Next, I install Apache on the instance by running `sudo apt-get install apache2`. Apache is installed and running using the default Ubuntu configuration. However, I can't actually hit the Web server from my desktop here, as port 80 and 443 are disallowed by the EC2 firewall. I do a quick `ec2-authorize default -p 80`, and now Firefox on my local machine can hit the

LINUXTM
JOURNAL

Linux News and Headlines Delivered To You



Linux Journal
topical RSS feeds
AVAILABLE

http://www.linuxjournal.com/rss_feeds

Web server I just installed on the EC2 instance, as shown in Figure 1. It's not the most exciting of Web pages, but it's something!

Last but not least, you'll want to terminate or shut down your instances when they're not in use to save money. That's done via the `ec2-terminate-instances` command. Simply run it with the ID number of your instance (which can be determined via the `ec2-describe-instances` command), and your instance will terminate:

```
$ ec2-describe-instances
RESERVATION r-d8b376b0 748502897232 default
INSTANCE   i-bc9913d4 ami-52be5d3b \
            ec2-72-44-62-167.compute-1.amazonaws.com \
            domU-12-31-39-00-ED-A3.compute-1.internal \
            running   ec2-keypair 0 \
            m1.small   2009-11-02T22:23:12+0000 \
            us-east-1d aki-76be5d1f ari-4cbe5d25

$ ec2-terminate-instances i-bc9913d4
INSTANCE   i-bc9913d4      running shutting-down
```

There you have it. You can run your own servers "in the cloud", thanks to Canonical and Amazon. It's certainly a lot faster than installing your own OS, and it requires no physical equipment nor the need to buy anything. EC2 instances can be a great way to extend your own computing platform, provide capacity on demand to a service or enable you to spin up a replacement server. Whatever you use EC2 for, keep in mind the hourly rate, or you run the risk of getting an unexpected bill at the end of the month.■

Bill Childers is an IT Manager in Silicon Valley, where he lives with his wife and two children. He enjoys Linux far too much, and probably should get more sun from time to time. In his spare time, he does work with the Gilroy Garlic Festival, but he does not smell like garlic.

Resources

Ubuntu EC2 Starters Guide: <https://help.ubuntu.com/community/EC2StartersGuide>

Amazon AWS Portal: <aws.amazon.com>

Amazon EC2 Signup: <aws.amazon.com/ec2>

Amazon EC2 Getting Started Guide:
<docs.amazonwebservices.com/AWSEC2/latest/GettingStartedGuide>

Amazon EC2 API/Management Tools:
<developer.amazonwebservices.com/connect/entry.jspa?externalID=351&categoryID=88>

Ubuntu EC2 Image List: <uec-images.ubuntu.com/releases/karmic/release>



KYLE RANKIN



BILL CHILDERS

/opt vs. /usr/local

Should a sysadmin put additional software in /usr/local or /opt? Bill and Kyle argue the one true location for third-party software.

This month, Bill and I take on one of the classic holy wars between Linux geeks: /opt vs. /usr/local. If you look at the current Linux Filesystem Hierarchy Standard, you will see that both /opt and /usr/local are represented there. If you read further, you will see that they both seem to have a similar purpose: a place to put software that's not part of the standard system. Even though both directories are designed for similar purposes, they each go about that task in different ways. Well, there's nothing quite like two very similar things with subtle nit-picky differences to spark a debate between geeks.

BILL: So what's wrong with /opt?

KYLE: There was nothing wrong with /opt, back when it was set up. You know, back in the days when tar was your package manager and dinosaurs roamed the earth.

BILL: "Back when it was set up." Oh, here we go again, another "Bill's older than dirt" comment.

KYLE: I'm not saying you're older than dirt, I'm just saying I've seen your yearbook pictures with dirt. Anyway, back then, it made sense for someone

It is nice having a distinct delineation of what's part of the distribution and what the admin installs—for me, it's just in /usr/local.

to package everything up in one big directory and dump it under /opt. But some time in the last two decades, most Linux distributions started using more sophisticated package managers.

BILL: Ahem. I rather *like* using /opt. It's nice having a distinct delineation as to what's installed by the distribution and what's installed by the admins after the fact.

KYLE: Wow, I totally agree with half of that statement.

BILL: Hey, there's a first. And it's in print too. Whoohoo!

KYLE: It is nice having a distinct delineation of what's part of the distribution and what the admin installs—for me, it's just in /usr/local.

BILL: This is the *first* time I've heard you, of all people, advocate more typing.

KYLE: Your system packages can dump their files in /usr, and any third-party packages can put things in an identical directory structure under /usr/local; however, because these days we aren't using tar, but programs like rpm and dpkg to install packages (and their yum and apt front ends), we have a much more sophisticated way to see what is installed and where it's installed, beyond just the ls command. Even then, using ls, I can see that a particular binary is in /usr/local/bin and, therefore, must be part of a third-party package.

BILL: I may be arguing semantics here, but that's what Point/Counterpoint is about. To me, /opt is for "options"—stuff that's post-install. /usr/local implies that its local to that machine. To me. Your "ls" point also applies to /opt, except the path is shorter, and you can't assume that everyone will be using rpm and dpkg.

KYLE: The path is shorter, eh? Well Bill, thanks for the setup.

BILL: What if you compile things from source, and don't want to go through the added steps of making a .deb? The bottom line is that there is no real "standard". All the admins I've seen tend to have their own spin on this.

KYLE: Once you start using /opt, you can count on your system paths increasing exponentially. With /usr/local, my library paths and my binary paths need to add only one entry (an entry that probably is already there).

BILL: Exponential? Only if you're installing a crazy amount of software, man. I rather like knowing that if I'm going to be building, say, a Java application

server, that my JDK is always in /opt/jdk (I typically have a symlink that points to the real JDK, like /opt/jdk_sun_1.6.0.17. That way, JAVA_HOME is always /opt/jdk. Any other packages, say a custom-compiled apache, can live in /opt/apache.

KYLE: But if you installed the JDK in /usr/local (not that Sun would ever approve), you could have all the libraries in /usr/local/lib and Java binaries in /usr/local/bin, and you could just use your regular package manager to see where things are installed.

BILL: That's only a couple paths. You're assuming that these things are packaged by the software maintainer or that I want to go through with making packages. Lots of times software's not packaged.

KYLE: It's an extra (and in my opinion, proper) step when you are deploying software to your servers, but it's a step that makes sure you can automatically handle dependencies and can use standard tools and not tar, cp and rm to add and remove packages.

BILL: Whoa, you're calling tar and cp "not standard tools"?

KYLE: Standard packaging tools. Let's take your apache example. If I wanted a custom apache, I'm going to have to compile it, right? All I have to do is use the --prefix option to change where it is installed from /usr to /usr/local. In my opinion, that's easier than the /opt approach.

BILL: It's rather nice to be able to take a completely working server and just rsync its directory to another box.

KYLE: Again, I suppose if you are a closet Solaris or Slackware admin, tar, cp and rm are your packaging tools, but if your add-on programs are in packages, you can just use a standard package manager.

BILL: Yes, if there's a package for it, or you want to go through the work of making one.

KYLE: That's what I think this argument ultimately comes down to: the traditional (and ancient) approach to install software before proper package managers came on the scene versus the modern way to deploy software on a server.

BILL: There are times when packaging is appropriate. If you've got lots of time to prepare, or require a lot of validation and control, then sure, package away.

KYLE: The modern way is to use package managers, so dependencies are easily managed—adding, removing and updating packages is managed, and there's an audit trail. The traditional way is just to untar or copy files around and hope they work. Plus, with the traditional way, you tie up extra space by sharing fewer libraries and just bundling everything together in each package, even if another package might use the same

libraries. The work of doing it "right" is work up front that saves you a lot of work down the road. I think what it comes down to is that Bill has a soft spot for /opt from all his years at Sun.

BILL: Hey, just because I feel nostalgic when I see an /opt/SUNW type path doesn't mean I don't have a point. On a development system, or a sandbox, having an /opt directory where you can just toss things and see if they work makes a whole lot of sense. I know I'm not going to go through the effort of packaging things myself to try them out. If the app doesn't work out, you can simply rm the /opt/mytestapp directory and that application is history. Packaging may make sense when you're running a large deployment (there are times when I do package apps), but lots of times, it's nice to toss stuff in /opt.■

Kyle Rankin is a Systems Architect in the San Francisco Bay Area and the author of a number of books, including *The Official Ubuntu Server Book*, *Knoppix Hacks* and *Ubuntu Hacks*. He is currently the president of the North Bay Linux Users' Group.

Bill Childers is an IT Manager in Silicon Valley, where he lives with his wife and two children. He enjoys Linux far too much, and he probably should get more sun from time to time. In his spare time, he does work with the Gilroy Garlic Festival, but he does not smell like garlic.



The first and only radio show broadcast in the USA dedicated exclusively to spreading the word about the LINUX OPERATING SYSTEM and FOSS.

GUTSY GEEKS
COMPUTER SHOW

gutsygeeks.com

**ERP/Accounting
Distribution/Mfg**

*Software for Linux
Source Code, Development Tools*

>> Free Eval License <<

Fitrix
Affordable, Adaptable ERP Software

fitrix.com/LJ1003
800.374.6157

axigen
MAIL SERVER

The Mail Server
for IT Professionals

www.axigen.com

INNOVATION ON THE GO
ORDER YOUR BEAGLE BOARD FROM DIGIKEY.COM

AVAILABLE EXCLUSIVELY AT DIGI-KEY
beagleboard

LOW-COST, NO FAN,
SINGLE-BOARD
COMPUTER

Digi-Key
CORPORATION

www.digikey.com

Polywell Storage Servers

More Choices, Excellent Service, Great Prices!

9015N

Quiet Storage NAS/SAN/iSCSI

8TB \$1,999
12TB \$2,599
30TB \$6,599

- Dual Gigabit LAN
- RAID-5, 6, 0, 1, 10
- Hot Swap, Hot Spare
- Linux, Windows, Mac
- E-mail Notification
- Tower or Rackmount

5048A

**5U-48Bay 96TB
Storage Server**

4U24A

4U-24Bay 48TB
RAID-6, NAS/iSCSI/SAN Storage
Mix SAS / SATA, 4 Giga / 10Gb LAN

2012A

2U-12Bay 24TB
RAID-6, NAS/iSCSI/SAN Storage
Mix SAS / SATA, 4 GigaLAN

Silent Eco Green PC
The Best Terminal PC
Intel® / AMD® x86 Processor
Energy efficient, Quiet and Low
Voltage Platform. starts at \$199

LD-001

1U945GCL2

Mini-1U Server \$499
Intel Dual-Core Processor, 2 x 500G RAID
Dual GigaLAN, 4GB DDR2 RAM

Polywell OEM Services, Your Virtual Manufacturer
Prototype Development with Linux/FreeBSD Support
Small Scale to Mass Production Manufacturing
Fulfillment, Shipping and RMA Repairs

■ 20 Years of Customer Satisfaction
■ 5-Year Warranty, Industry's Longest
■ First Class Customer Service

888.765.9686
linuxsales@polywell.com
www.polywell.com/us/Lx

POLYWELL

TUXERA

Leading NTFS and exFAT Interoperability



Every device with mass storage, at home and at work, needs an interoperable file system.

Your set-top-box runs Linux but needs to write the media in a Windows and Mac compatible format.

Tuxera NTFS for Embedded Systems

Performance increase 10-100x compared to our open source NTFS-3G. Proven reliability and data integrity. Low CPU usage, small memory footprint. Available for any system.

Tuxera exFAT for Embedded Systems

exFAT is part of SDXC and Memory Stick standards. New product available now for Linux.

+358 50 5980498 sales@tuxera.com www.tuxera.com

WAN Toolkit

Featuring

- * Secure, distributed file system
- * Centralized, web-based administration
- * Easy installation
- * Free 30-day trial with permanent results

<http://www.devicesoft.org>

Automated OS Lock Down for Linux and Solaris

Are you using scripts to lock down your operating systems? Security Blanket automatically locks down your OS to meet industry (DISA STIGS, CIS, SANS, etc.) or customized standards.



For a Free Trial of

Security Blanket visit

<http://www.trustedcs.com/>
SecurityBlanket/SecurityBlanket-Try-Out.html

Polywell Mini-PCs



ITX with VESA / Wallmount



ITX-50C



ITX-10A 1.4" ThinPC



ITX-20A with SlimDVD



Fanless Slim PC with Intel® 1.6GHz 1W Mobile Atom™ Processor, DC12V Power-in, 1GB DDR2 RAM starts at \$199
Supports SATA Hard Drive or Solid State Drive (SSD), Optional PCI RISER Slot for TV Tuner or other Add-on Device on ITX-30A



ITX-1000C with 4LAN and WiFi Option

NVIDIA® ION™

The World's Small, Greenest, Fanless PC with Blu-Ray Ready

ITX-40A with NVIDIA® ION™ Graphics \$199 Barebone system
4GB RAM, 1.6GHz Intel® 4W Processor \$599 with Blu-ray, 500G HD



ITX-30A with PCI Riser



Over 250 Mini-ITX Models Available:

- NVIDIA® GeForce 8200/8100 with AMD® Athlon/Phenom Processor
- NVIDIA® GeForce 9300/7100/7050 with Intel® Core 2 Duo Processor
- PCI, PCIe, MiniPCIe Slot for TV Tuner or Industrial Add-on
- Custom Design Chassis for Small to Mid Size OEM Project

888.765.9686

linuxsales@polywell.com

polywell.com/us/Lx



■ 23 Years of Customer Satisfaction

Polywell Computers, Inc.

NVIDIA, ION, nForce, GeForce and combinations thereof are trademarks of NVIDIA Corporation. Other names are for informational purposes only and may be trademarks of their respective owners.

■ 5-Year Warranty, Industry's Longest

1461 San Mateo Ave. South San Francisco, CA 94080 650.583.7222 Fax: 650.583.1974

■ First Class Customer Service

A Cloud of One's Own

The true nature of supply and demand. DOC SEARLS



The phrase "supply and demand" first appeared as "demand and supply" in *An Inquiry into the Principles of Political Economy* by James Denham-Steuart. It was published in 1767, nine years before Adam Smith, a fellow member of the Scottish Enlightenment, came out with *The Wealth of Nations*. Among Denham-Steuart's many points about demand, two stand out. One is that demand "must constantly appear reciprocal. If I demand a pair of shoes, the shoemaker either demands money, or something else for his own use." The other is that "The nature of demand is to encourage industry."

And, perhaps it did. Not long after Denham-Steuart's time, Industry won the Industrial Revolution, and it has continued winning ever since. This has had the effect not only of putting supply ahead of demand as a central concern of economists, but also to reverse Denham-Steuart on the matter of who encourages whom. For example, the entire marketing industry—and advertising in particular—is premised on a belief that it is the nature of industry to encourage demand, which it does. But, does that mean demand must do only what supply wants? Or that demand can't often supply itself? Or that demand can't encourage industry in ways other than those that industry supplies?

The Internet has begun to obsolete this one-sided view (it's now plain that consumers do far more than consume). But most of us still anchor our economic perspective on the supply side. That is, we still look from supply toward demand, rather than from demand toward supply. Thus, we help demand by improving supply, not by helping demand—in Denham-Steuart's terms—to encourage industry. And, since every supplier is different, we end up with a choice as customers among many different ways for suppliers to encourage us. That's why, for example, we walk around with dozens of loyalty cards for different retailers, and why business in general remains a world of silos, even as the technology world is gradually getting

hip to the silo-busting advantages of free and open-source software.

In fact, open-source code is possibly the best example we have of demand-and-supply. Is there a single open-source code base that was not created to overcome the absence of code with its functions already in the marketplace? Would Sergey Brin and Larry Page have bothered making Google if free and useful code were not out there to help make it possible?

Still, there is a difference between supplying building material and looking for more direct effects.

This brings me to the matter of cloud computing. It's a subject on which bedfellows no less odd than Larry Ellison and Richard M. Stallman find themselves in degrees of agreement. According to a September 2008 report in *The Guardian*, Ellison said, "Maybe I'm an idiot, but I have no idea what anyone is talking about. What is it? It's complete gibberish. It's insane. When is this idiocy going to stop?" Stallman said, "It's worse than stupidity: it's a marketing hype campaign....Somebody is saying this is inevitable—and whenever you hear somebody saying that, it's very likely to be a set of businesses campaigning to make it true." He added: "One reason you should not use Web applications to do your computing is that you lose control....It's just as bad as using a proprietary program. Do your own computing on your own computer with your copy of a freedom-respecting program. If you use a proprietary program or somebody else's Web server, you're defenseless. You're putty in the hands of whoever developed that software."

True: the PCs of today might be a lot smarter than the dumb terminals of computing's mainframe and minicomputer ages, but in respect to clouds, they're still terminals. That is, they are still remote: architecturally peripheral to the cloud itself.

Now, we could argue about what clouds are good for and have deep digressive exchanges about the premises (or even the facts) in that last paragraph, but instead, let's address this question: Why not

have your own cloud? That is, why not be what Joe Andrieu calls the point of integration for your own data and the point of origination about what gets done with it?

One answer I like comes from The Mine! Project (themineproject.org), an open-source effort conceived and named by Adriana Lukas, and turned into code by Alec Muffett and a crew of allied geeks. As a name, Mine! refers to the first-person singular possessive pronoun, but its symbol is a miner's pickaxe: a tough real-world tool. The project's purpose, say Adriana and Alec, is to equip individuals with "tools and functionalities" that give them ways to "take charge of their data (content, relationships, transactions, knowledge); arrange (analyse, manipulate, combine, mash-up) it according to their needs and preferences and share it on their own terms whilst connected and networked on the Web."

Although not exactly a cloud of one's own (which may be a contradiction in terms), it's close enough to obey RMS's admonitions. That is, it gives you control of your data and what can be done with it by others.

In a London gathering in early December 2009, Alec and Adriana showed Pymine, which they called "the first implementation of The Mine! Project software, in Python/Django". As far as I know, it's the first purely open-source project that directly supports the mission I called out for ProjectVRM in 2006: to equip individuals with tools that make them both independent of vendors and better able to engage with vendors. Although it's not limited to commercial purposes. Any individual can use it for anything they like. "It's meant as infrastructure", Alec said. What matters is that the infrastructure is yours. You don't have to build it on somebody else's platform. Or cloud. ■

Doc Searls is Senior Editor of *Linux Journal*. He is also a fellow with the Berkman Center for Internet and Society at Harvard University and the Center for Information Technology and Society at UC Santa Barbara.

iX-Athena Workstation

- ✓ Powerful Performance
- ✓ (93%+) Energy Efficiency
- ✓ Super-Quiet Operation

Highest
Level MTBF
Cooling

To order today call: **1-800-820-BSDi**



Notable features include:

- Dual 64-Bit Socket 1366 Quad-Core or Dual-Core, Intel® Xeon® Processor 5500 Series
- Eight 3.5" Hot-swap SAS/SATA HDDs in a 4U/Tower Configuration (Optional 4U Rackmount Rail Kit Available)
- Dual Intel® 5520 Chipsets with Quick-Path Interconnect (QPI) up to 6.4 GT/s
- Up to 144GB DDR3 1333/1066/800MHz ECC Registered DIMM/24GB Unbuffered DIMM (18 DIMM Slots)
- Two (x16) PCI-E 2.0 slots, Four (x8) PCI-E 2.0 slots (1 in x16 slot), and One (x4) PCI-E slot (in x8 slot)
- Intel® 82576 Dual-Port Gigabit Ethernet Controller
- Matrox G200eW Graphics Support
- Integrated IPMI 2.0 with Dedicated LAN
- Realtek ALC888 7.1 HD audio
- 1400W Redundant High Efficiency Power Supply (Gold Level 93%+ power efficiency)

iXsystems Introduces the iX-Athena Workstation

The iX-Athena showcases amazing computing performance and energy efficiency, while keeping noise levels to a minimum.

The iX-Athena delivers the most powerful performance available on the market today. Dual Intel® Xeon® 5500 series Quad-Core or Dual-Core processors boost performance for specific workloads by increasing processor frequency. Next-generation Intel® Virtualization Technology enhances performance in virtualized environments by up to 2.1x with new hardware-assist capabilities. Up to 144GB of DDR3 memory with eighteen DIMM sockets supports higher performance for data-intensive applications and makes it easy for the iX-Athena to handle any workload.

In terms of energy efficiency, the iX-Athena also leads the pack. The automated low-power states of the Intel® Xeon® 5500 series processors intelligently save power during low-use periods and increase performance when the system requires it. The iX-Athena also features an FCC Class B certified power supply with gold level (93%+) energy efficiency to provide 1400W of power and minimize impact on the environment.

The Super-Quiet operation of the iX-Athena allows users to spend less time distracted by a loud machine, and more time focusing on its powerful computing capabilities. At normal operation levels, the iX-Athena workstation's 5,000 RPM cooling and exhaust fans perform at a hushed 38 decibels to make this an ideal machine for any office or lab environment.

With eight 3.5" hot-swappable SAS/SATA hard drive bays, the iX-Athena also offers ample storage for all conceivable technical computing and graphics applications. The iX-Athena even includes four dedicated power connectors for high-end graphics cards, all contained in a stylishly sleek, high-end quality, dark gray chassis.

For more information about the iX-Athena visit:

<http://www.iXsystems.com/Athena>



More GFLOPS, Less WATTS

Intel® Nehalem is here!

**Higher Memory Bandwidth with DDR3 and QPI
Clusters and Servers Consume Less Power**

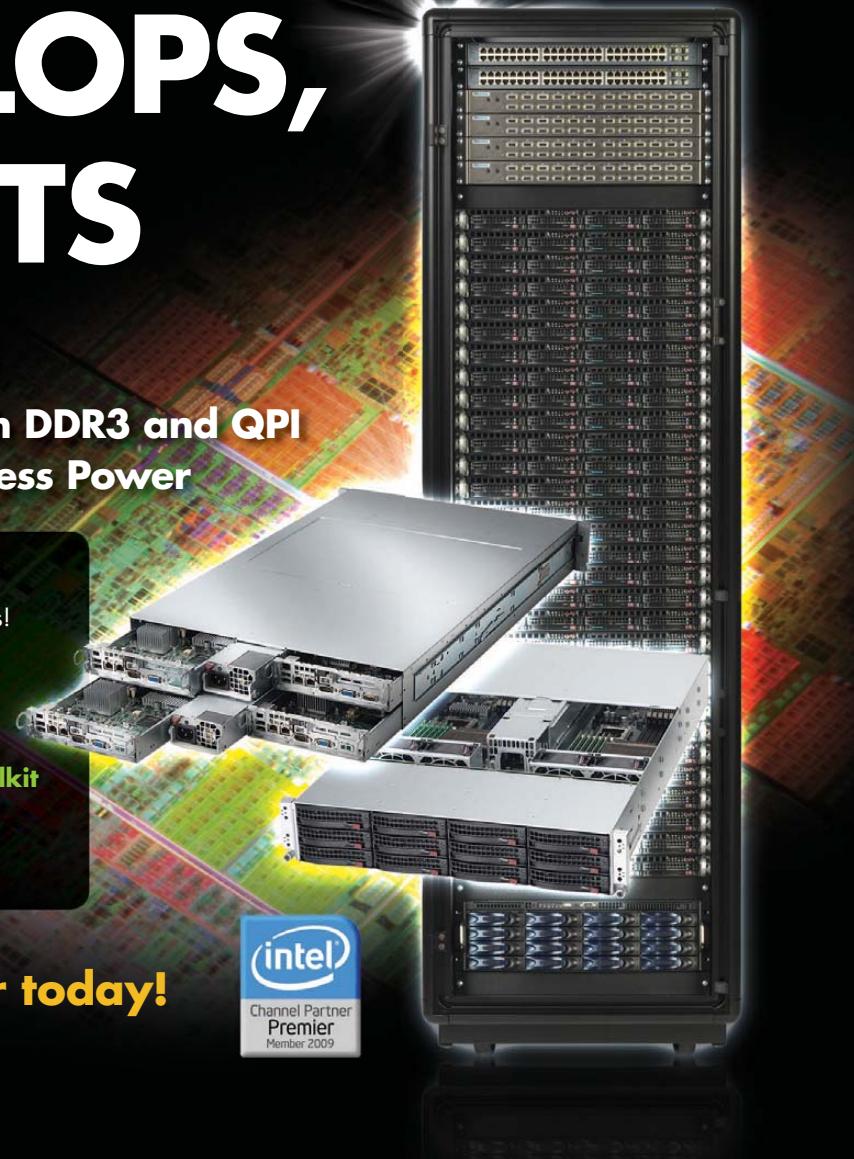
Four Servers in a 2U Chassis with all Hot-Swap:

- ▶ 1200 Watt 1+1 supply, 12 Drives, and Server Modules!

**FasTree™ ConnectX® QDR and DDR InfiniBand
Switches and HCAs**

Intel Professional Compiler Suite and Cluster Toolkit

- ▶ Version 11 with Nehalem Enhancements
- ▶ Academic Pricing Available



Configure your next Cluster today!

www.microway.com/quickquote

GPU Computing



WhisperStation™ With 1 to 4 Tesla GPUs

Tesla C1060 GPU Performance:

- ▶ 1 TFLOPS per GPU
- ▶ 4 GB DDR3 per GPU
- ▶ 102 GB/Sec Bandwidth
- ▶ CUDA SDK

Run MATLAB® on Tesla with "Jacket"

Clusters With Tesla™ S1070 - 4 GPU Servers

- ▶ 36 GPUs + 36 CPUs + 24 TB in 24U
- ▶ 40 Gbps FasTree™ InfiniBand
- ▶ InfiniScope™ Network Monitoring

FREE

15-day trial available
[at microway.com](http://microway.com)

Microway
Technology you can count on™

508-746-7341
microway.com



GSA Schedule
Contract Number:
GS-35F-0431N