MongoDB | OpenVPN | Nmap | Pogoplug | MINIX | Tech Tips

LINUX JOURNAL

Since 1994: The Original Magazine of the Linux Community
JUNE 2010 | ISSUE 194 | www.linuxjournal.com

Organize Your Library

MongoDB's High-Powered Features

Automate Remote Backups

DISTRIBUTIONS

REVIEWED: Pogoplug File Server

LINUX ON NON-X86 COMPUTERS

QUICK DISTRO HISTORY LESSON

Philosophy of Distributions

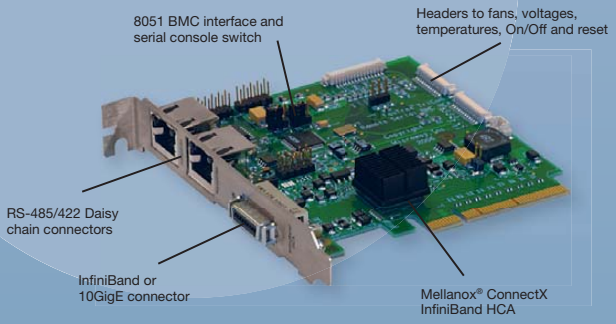
+ Point/Counterpoint: MOBILE PHONES

$5.99US $5.99CAN
06
0 09281 03102 4

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

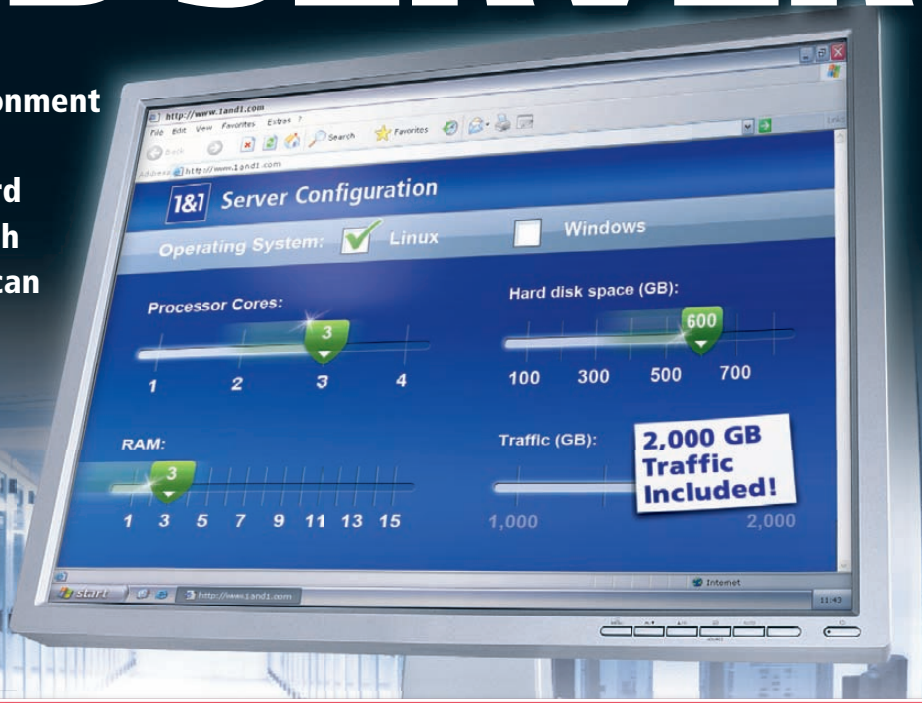
Sign up for Microway's Newsletter at
www.microway.com

Microway®
Technology you can count on℠

The 1&1 server totally configurable to your needs:

DYNAMIC CLOUD SERVER

A powerful virtual server environment with full root access. Adjust the processor core, RAM, and/or hard disk space to fit your needs. With the Dynamic Cloud Server, you can change your specifications at any time!



AMD Opteron™ 64

THIS COMPANY USES CERTIFIED CLEAN ENERGY

SPECIAL OFFER: 3 MONTHS FREE*

1&1® Dynamic Cloud Server – basic configuration includes:

- ✔ **1 Virtual Core of a Quad-Core AMD Opteron™ 2352 Processor**
- ✔ **1 GB RAM**
- ✔ **100 GB disk space**
- ✔ **Guaranteed resources** (just like a dedicated server!)

~~$49.99~~ per month

More server offers are available online. Visit our website for details.

*Offer valid as of May 1, 2010. Offer applies to Dynamic Cloud Servers only, up to a maximum discount of $149.97 per server. 12 month minimum contract term and setup fee apply. Prices valid for basic configuration only. For other configurations, additional costs apply. Visit www.1and1.com for full promotional offer details. Program and pricing specifications and availability subject to change without notice. 1&1 and the 1&1 logo are the trademarks of 1&1 Internet AG, all other trademarks are the property of their respective owners ©2010 Internet, Inc. All rights reserved.

MEMBER OF united internet

Call **1-877-GO-1AND1**

Visit us now **www.1and1.com**

1&1®

CONTENTS

JUNE 2010
Issue 194

FEATURES

DISTRIBUTIONS

50
DISTRIBUTIONS—A BRIEF HISTORY
The dawn of the distro.
Jes Fraser

54
PHILOSOPHY AND FANCY
The intellectual axes of distribution design.
Dan Sawyer

58
SMALL SYSTEMS AND BIG IRON: LINUX ON NON-X86 COMPUTERS
Linux distributions on "alternative" platforms.
Kira Scarlett

62
LINUX DISTRIBUTION CHART
See how your favorite distro ranks with other *LJ* readers.
Justin Ryan

ON THE COVER

- Organize Your Library, p. 34
- MongoDB's High-Powered Features, p. 18
- Automate Remote Backups, p. 68
- Linux on Non-x86 Computers, p. 58
- Quick Distro History Lesson, p. 50
- Philosophy of Distributions, p. 54
- Reviewed: Pogoplug File Server, p. 46
- Point/Counterpoint: Mobile Phones, p. 74

The latest 1&1 server solution for high performance needs:

HEXA-CORE TECHNOLOGY

NEW!

The ultimate in server technology, our powerful new hardware class is the perfect solution for running your resource-intensive applications.

THIS COMPANY USES CERTIFIED CLEAN ENERGY

SPECIAL OFFER: 3 MONTHS FREE*

1&1® Hexa-Core Servers – using the latest generation of AMD six-core processors:

- ✔ **2 x Six-Core AMD Opteron™ 2423 HE Processor**
- ✔ **Up to 32 GB memory**
- ✔ **Up to 2 TB of usable disk space with RAID 5**
- ✔ **Energy efficient, AMD-P technology**

Starting at ~~$499.99~~ per month

More server offers are available online. Visit our website for details.

*Offer valid as of May 1, 2010. 12 month minimum contract term and setup fee apply. Visit www.1and1.com for full promotional offer details. Program and pricing specifications and availability subject to change without notice. 1&1 and the 1&1 logo are the trademarks of 1&1 Internet AG, all other trademarks are the property of their respective owners.
©2010 Internet, Inc. All rights reserved.

MEMBER OF united internet

Call **1-877-GO-1AND1**

Visit us now **www.1and1.com**

1&1®

CONTENTS JUNE 2010 Issue 194

COLUMNS

18 REUVEN M. LERNER'S AT THE FORGE
Advanced MongoDB

24 DAVE TAYLOR'S WORK THE SHELL
Simple Scripts to Sophisticated HTML Forms

26 MICK BAUER'S PARANOID PENGUIN
Linux VPNs with OpenVPN, Part V

30 KYLE RANKIN'S HACK AND /
Dynamic Config Files with Nmap

34 DIRK ELMENDORF'S ECONOMY SIZE GEEK
Organizing a Library

74 KYLE RANKIN AND BILL CHILDERS' POINT/COUNTERPOINT
Mobile Phones

80 DOC SEARLS' EOF
Commons Interests

REVIEW

46 POGOPLUG
Mike Diehl

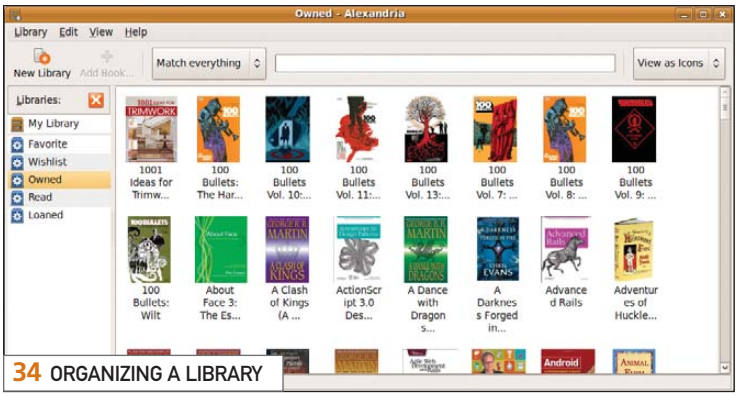
INDEPTH

64 AN INTRODUCTION TO MINIX
It's not just a filesystem supported by Linux.
Bruce Byfield

68 AUTOMATING REMOTE BACKUPS
Save or lose it!
Michael J. Hammel

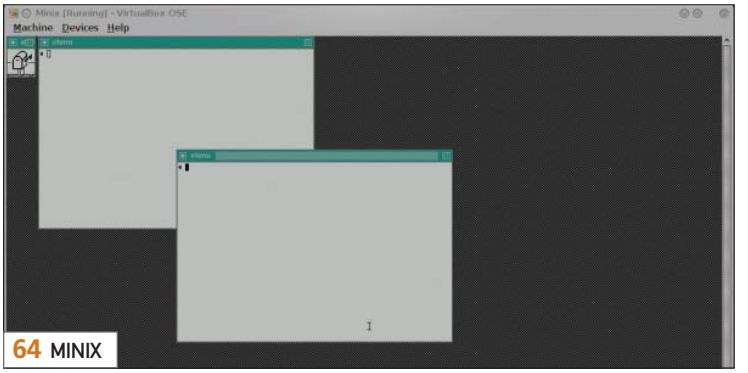
IN EVERY ISSUE

8 CURRENT_ISSUE.TAR.GZ
10 LETTERS
14 UPFRONT
40 NEW PRODUCTS
42 NEW PROJECTS
65 ADVERTISERS INDEX
73 TECH TIPS
78 MARKETPLACE



34 ORGANIZING A LIBRARY

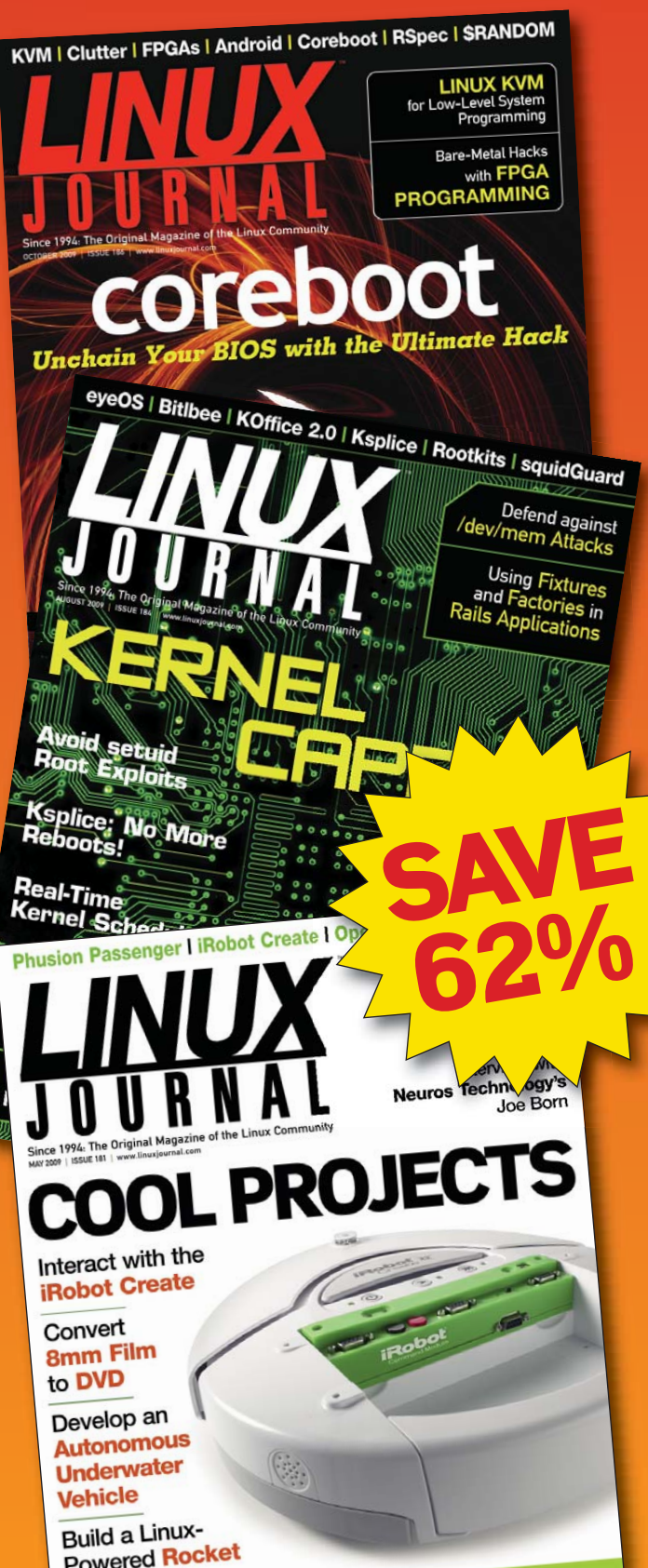
46 POGOPLUG



64 MINIX

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

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



SAVE 62%

- 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.

LINUX JOURNAL™
Since 1994: The Original Magazine of the Linux Community

DIGITAL EDITION NOW AVAILABLE!

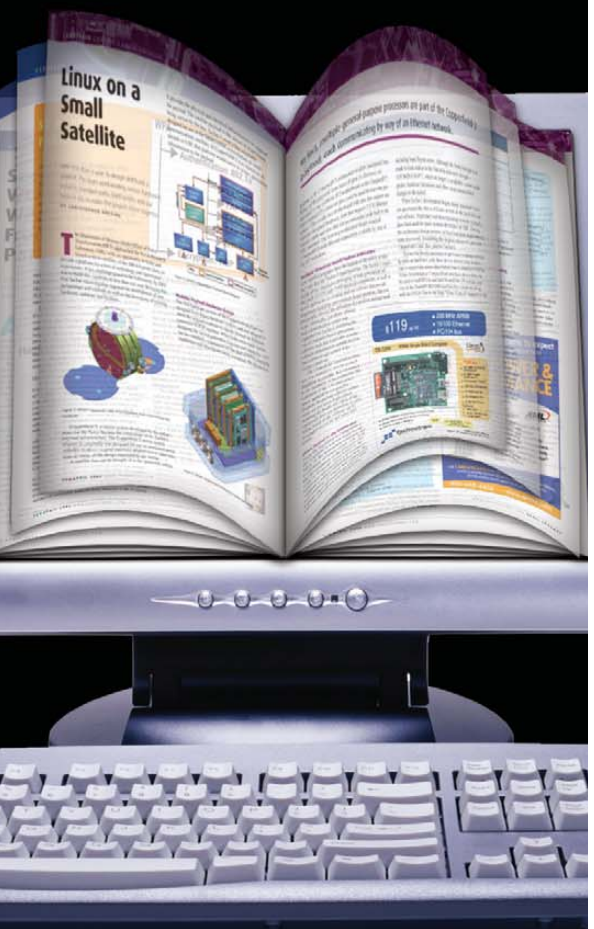
Read it first
Get the latest issue before it hits the newsstand

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	Garrick Antikajian garrick@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
Senior Print Media Sales Manager	Joseph Krack joseph@linuxjournal.com
Digital Media Sales Manager	Michael Beasley michael@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 Baillio • Nick Baronian • Hari Boukis • Steve Case
Kalyana Krishna Chadalavada • Brian Conner • Caleb S. Cullen • Keir Davis
Michael Eager • Nick Faltys • 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 Quinlan • 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.

PRINTED WITH SOY INK®

Are You Still Using Scripts to Lock Down Your OS?

Security Blanket ensures enterprise-wide policy configuration with fully automated OS lock down.

Using scripts to lock down your OS can be time consuming and error prone. And, keeping those scripts up-to-date as new guidelines are published requires a continuous cycle of research, interpretation, and script revision. ***Do you really have the time for this?***

Security Blanket automatically configures the OS using the most up-to-date industry standard security guidelines. Whether locking down one server or an entire enterprise, Security Blanket performs fast, consistent, and repeatable OS lock down.

Try it for free at www.trustedcs.com/SecurityBlanket or call us at 1-888-662-2724 for more information.

SHAWN POWERS

The Cake Is Not a Lie!

Linux is a kernel. Before I go any further, it's important to know that "Linux" is really just the kernel that powers our awesome computer systems. Before we get into heated debates about .deb and .rpm, or holy wars over GNOME and KDE, we never can forget that Linux really is just the kernel.

Likewise, cake is just the sweet bread stuff that hides under the frosting. Sure, icing, candles, plastic superheros and burning flames are what make a cake into a party, but at the base of things, cake is just that bread stuff. This issue, we talk about frosted cake and discuss the wonderful world of decorating. (If you prefer, feel free to use a mashed-potato-and-gravy metaphor instead. I realize we're not all cake fans.)

My first suggestion for this issue is that you turn to our centerfold. Granted, it's not really in the center (it's on pages 62 and 63), but it is a two-page spread that presents a ton of popular distributions. Our own Justin Ryan filled in the abundant blanks for us, and it's worth more than a cursory glance. If you like what you see, perhaps the next logical step would be to read Jes Fraser's article discussing the history of Linux distributions. Jes shows the whole gamut, starting from the Usenet post on the MINIX newsgroup.

What is MINIX, you ask? That's a good question. Bruce Byfield happily answers by discussing the present-day operating system that was the basis for Linus' early kernel. Although certainly not as popular as our coveted Linux, MINIX is a good way to learn about our roots without the complications of different colored frosting—er, distributions. Working with MINIX, while still being a Linux user, might start to make you think about the philosophy of your operating system. It certainly did for Dan Sawyer, and this month, he discusses some of the fundamental issues that Linux users face. What does it mean to be free? Is commercialism a good or bad thing? Is butter cream or Dream Whip the best icing for cake? However you slice it, Linux is many different things for many different people.

I know some of you are getting worried that this issue is starting to sound more like a college lecture than your normal monthly dose of tech goodies. Fear not; we have the perfect snack for your geeky sweet tooth as well. Mick Bauer wraps up his series on OpenVPN; Dave Taylor gets intense with some sophisticated HTML forms, and Reuven M. Lerner adds more about MongoDB.

Kyle Rankin proves that good workers are workers that can make computers do their work for them. He demonstrates making config files using nmap. It may sound strange, but it certainly will save you some grunt work and leave you time for other, more enjoyable things. If one of the things you enjoy doing is reading books, you'll want to read Dirk Elmendorf's article on library software. My wife is a librarian, and I can assure you that the days of the card catalog are over. Thankfully, Linux can step in and manage your books, whether you have only one shelf or enough books to overwhelm Mr Dewey himself.

No matter what your favorite cake is or what kind of frosting you prefer, everyone needs a pan to bake it in—that means hardware. With Linux, it's hard to find a limit to what we can use as an installation base. Kira Scarlett talks about a few of the less-common architectures and some reasons you might want to try them out. Mike Diehl reviews the Pogoplug device. It's tiny, cute, and it runs Linux. There are so many devices that run Linux, sometimes it's hard to choose a platform. Bill Childers and Kyle Rankin don't seem to have that problem, however, and this month they set out to prove why their cell-phone choice is best. As usual, I'm going to try to stay out of it.

So this month, whether you favor apt-get over emerge, or like fedoras instead of lizards, we all can agree to like cake—and the Linux kernel. Hopefully, you'll learn a few things, but more important, we hope you're inspired to try different distributions. Because really, it's hard to pick bad cake.■

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.

servers DIRECT®

MORE PRODUCTS, BETTER SERVICE, GUARANTEED.

GO STRAIGHT TO THE SOURCE!

1.877.727.7887 | www.ServersDirect.com

WANT YOUR BUSINESS TO BE MORE PRODUCTIVE?

The ServersDirect® Systems powered by Intel® Xeon® Processor provides the quality and dependability to keep up with your growing business.

HIGH EFFICIENCY POWER & HIGH STORAGE CAPACITY SERVER
SDR-S4314-T36

ENTERPRISE-LEVEL HIGH CAPACITY STORAGE SERVER

+ Maximum 3.5"" hot-swap drives density
  - 36x (24 front + 12 rear) Hard Drive Bays
  - E1: Expander supports SAS
+ Redundant (1+1) 1400W Gold Level power supply with PMBus function
+ Redundant, Hot-pluggable cooling system, Power Supplies, Hot-swap drives
+ Support up to 192GB DDR3 1333/ 1066/ 800MHz ECC Reg DIMM
+ Intel® 82576 Dual-Port Gigabit Ethernet Controller

80 PLUS

SDR-S1343-T04

1U INTEL® XEON® PROCESSORS 5500 SERIES SERVER W/ 4X 3.5" HOT-SWAP SATA DRIVE BAYS

- Supermicro 1U Rackmount Server with HE 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

80 PLUS

SDR-S2318-T06

2U INTEL® XEON® PROCESSORS 5500 SERIES SERVER W/ 8X 3.5" HOT-SWAP SAS/SATA BAYS

- Supermicro 2U Rackmount Server w/650W HE 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 192GB DDR3 1333/ 1066 / 800MHz ECC Reg DIMM
- 6x 3.5" Hot-swap SAS / SATA Drive Bays
- Dual Intel® 82574L Gigabit Ethernet Controller

80 PLUS

SDR-S3308-T08

3U INTEL® XEON® PROCESSORS 5500 SERIES SERVER W/ 8 X 3.5" HOT-SWAP DRIVES TRAYS

- Supermicro 3U Rackmount Server w/650W HE 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 192GB DDR3 1333/ 1066 / 800MHz ECC Reg DIMM
- 8 x 3.5"" Hot-swap Drives Trays 6 x SATA Hard Drives Supported
- Dual Intel® 82574L Gigabit Ethernet

SDR-S4309-T08

TOWER/4U INTEL® XEON® PROCESSORS 5500 SERIES SERVER W/ 8 X 3.5" HOT-SWAP DRIVES TRAYS

- Supermicro Tower 800W 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 192GB DDR3 1333/ 1066 / 800MHz ECC Reg DIMM
- 24x 3.5" Hot-swap SATA Drive Bay
- Intel® 82576 Dual-port Gigabit Ethernet Controller

80 PLUS GOLD

SDR-S4314-T08

TOWER/4U INTEL® XEON® PROCESSORS 5500 SERIES SERVER W/ 8X 3.5" HOT-SWAP SATA BAYS (SUPPORT UP TO 4 DOUBLE-WIDTH GPU)

- 4U Rackmountable / Tower with 1400W Gold Level Redundant Power Supply Optional Rackmount Kit
- Supermicro Server Board w/ Dual Intel® Dual Chipset ICH10R
- Support up to Dual Intel® 5500 series Xeon® Quad/Dual-Core, with QPI up to 6.4 GT/s
- Support up to 192GB DDR3 1333/ 1066 / 800MHz ECC Reg DIMM
- "• 8 x 3.5"" Hot-swap Drives Trays Supports up to 6 SATA Drives"
- Intel® Dual 82574L Gigabit Ethernet
- **4x NVIDIA Tesla C1060 GPU Cards**

SDB-S7002-T00

9U INTEL® XEON® PROCESSORS 5500 SERIES SERVER W/ 50X HOT-SWAP SATA II / SAS BAYS

- Supermicro SBE-714D-D28 Enclosure chassis with two 1400W power supplies. Up to 14 hot-plug processor blades
- Support Intel based blades
- One Management modules comes standard with each enclosure
- One hot-plug management modules providing remote KVM and IPMI 2.0 functionalities
- Support one hot-plug Gigabit Ethernet switch, Pass - Through Module

10th year

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.

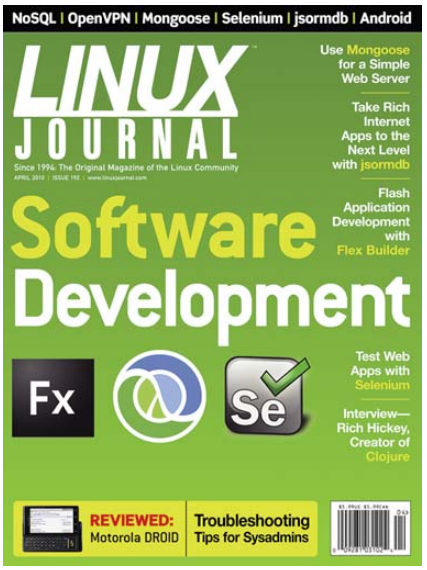
intel Xeon inside

Powerful. Intelligent.

1.877.727.7886 / www.ServersDirect.com

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

letters



Point/Counterpoint/Point

In the March 2010 Point/Counterpoint column, Kyle Rankin pretends that he has to use --prefix to install apache to /usr/local instead of /usr.

Well, he surely hasn't used any ./configure script in the past 15 years, because it goes to /usr/local by default, and apache2 uses /usr/local/apache2 as a default prefix.

One less "point" in favor of Kyle, heh?

--
Arnaud Launay

Kyle Rankin replies: *You are right. When Bill and I do the Point/Counterpoint column, it's all off-the-cuff responses, so I talked about --prefix with ./configure while I was really thinking about --prefix in building and installing RPMs. I didn't realize the default install location for ./configure was /usr/local (although in my opinion, that makes an even stronger argument for /usr/local: it's the default). To be honest, although I compiled many programs back when I started using Linux (it was almost a requirement to compile the kernel at the very least), I rarely compile anything anymore, especially when it's going to be used in production. I think the moment you go down that road, especially when you have a large environment, you lose all the testing benefits you get with distribution packages, and you become a QA department of one. Now, I know that many system administrators still like to do everything by hand, but to me, the risks and maintenance headaches of that far outweigh any benefits. It may be okay with one or two servers, but in my opinion, it doesn't scale. Thanks for the comment.*

Simplicity and Support

Mick Bauer's VPN series is long overdue [see Mick's Paranoid Penguin columns starting with the February 2010 issue through this current issue]. But in practice, few medium-to-large companies are going to deploy an OpenVPN server. Their corporate management has at least two requirements it can't meet: *simplicity* and *support*. To them, simplicity means a proprietary appliance from a name-brand vendor—they distrust software running on *nix in their infrastructure, never mind that the black-box appliance is probably just software running on embedded BSD or Linux. And, support is a euphemism for "somebody to sue".

This leaves those of us who need to make VPN connections to multiple clients/partners in the unenviable position of being stuck with one or more computers with a half-dozen incompatible proprietary VPN clients installed (most of which work only on MS Windows).

How about an article on making the OpenVPN client, and/or other open-source VPN clients, interoperate with servers from some of the popular VPN vendors, such as Cisco, Juniper, Nortel and so on, and the issues involved in connecting and authenticating to proprietary VPN servers?

--
Tom Uffner

Mick Bauer replies: *This particular series of articles was targeted at home/small-office/medium-office administrators; it was no coincidence that all my examples talked about connecting back to my house or that I showed a server configuration allowing only two concurrent connections. But by all means, you're correct, an article on connecting Linux clients to big commercial VPN concentrators would be useful. I'll consider that for a future column!*

In the meantime, I can offer a quick hint. Whereas in client mode, OpenVPN can connect only to OpenVPN server processes, the free utility vpnc can connect to Cisco and Juniper/NetScreen VPN servers. Thanks for writing!

Making Linux Known to Computer Haters

Microsoft spends hundreds of millions of dollars on advertising each year, reaching those in the community who own PCs but spend very little time with them. Microsoft's goal is to reinforce that Microsoft and Windows are what everyone uses on their PCs and that's just the way it is. I believe if a survey was taken world-wide on what operating systems were available to home PC users, a large number would say Windows and Mac. So, what's this letter all about? It's about finding ways without spending money to get Linux out to those who have no interest in operating systems in general. How (like Microsoft and Windows) do we reach people and make them aware of the existence of Linux without spending money?

I propose you write a article on the "Best Free Creative Ideas" that promote Linux. Here's mine. Get your readers to leave old copies of *Linux Journal* in doctors' waiting rooms (and similar places) where you have a captive audience looking for something to read. Maybe run a poll for the best ideas from your readers.

--
John Van Gaans

John, that's a great idea! Perhaps the Web site is a good place to get feedback.—Ed.

Send Letters to the Editor at
www.linuxjournal.com/contact

We Speak Your Language

```
> if ($ordered_online && $coupon_code == 'linux
> journal')
> {
> $server_discount = $server_price * .1;
> # 10% off for coupon code 'linuxjournal'
> }
>
```

Choose the dedicated and managed hosting provider that understands you.

1.877.999.2701
www.codero.com

Server Solutions

codero

Codero 2010. All Rights Reserved. 04.LJ.10

Linux across the Age Spectrum

This is a great magazine, and I have learned a great deal from it. I can safely say that with the help of this magazine, you made this 14-year-old learn a lot more about Linux, open source and computer management in general. Gotta say, mad props to you guys—keep the good news going. I look forward to reading the latest issue!

--

Alex Conrey

Aw shucks, Alex. It thrills me that a 14-year-old reads Linux Journal*! I put my monthly issues in our local school library, but they don't get read nearly as much as I'd like. The rest of the editorial staff isn't very keen on my idea of including a centerfold each month with hot new hardware. It's good to hear that even without such eye candy, the magazine is still appreciated.—Ed.*

dwm

On my PC running Debian GNU/Linux I use dwm (**dwm.suckless.org**) as my X window manager, and I like it very much. Now, on Wikipedia, there is an article about dwm (**en.wikipedia.org/wiki/Dwm**), but there is a discussion going on about deleting the dwm article (**en.wikipedia.org/wiki/Wikipedia:Articles_for_deletion/Dwm**). One of the reasons is "This article needs references that appear in reliable third-party publications."

So, I was wondering if *Linux Journal* could publish an article about this small and powerful window manager.

--

Jan Wagemakers

Thanks for bringing this to our attention. We'll take a look at it, and see if it inspires anyone on staff to write about it.—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.

McAfee enterprise products run on McAfee Linux. McAfee Linux team members must dress in their daily uniform to show their support for the product. This is me standing in front of the Alpharetta, Georgia, office. All hail Tux! Submitted by John Masinter.

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**.

The new Intel® Xeon® processor 5600 series increases performance, cuts operating costs, and helps deliver ROI. That's intelligent performance[1]. Check out the new intelligent features of the Xeon 5600 at intel.com/itcenter.

intel Xeon inside™

Powerful. Intelligent.

Performance:

Your processor gets the job done.

Intelligent performance:

Your processor gets the job done with 40% more performance by adapting to your workload.[2]

Servers from iXsystems feature the Intel® Xeon® processor 5600 series.

iXsystems™

Enterprise Servers
for Open Source
www.iXsystems.com
800-820-BSDi

Cleverly disguised as any other 2U server, the Gemini[2] houses 4 highly efficient, extremely powerful RAID 5 capable servers.

The Orion II 4U Storage Solution boasts energy-efficient technology and maximum, high-density storage capacity, creating a 4U powerhouse with superior cooling.

Intel is not responsible for and has not verified any statements or computer system product-specific claims contained herein.

Gemini[2]

- Intel® Xeon® processor 5600 series
- 4 Nodes in a 2U Form Factor
- Up to 192GB DDR3 RAM Per Node
- 3 x 3.5" SAS/SATA Drives/Node

Starting at **$3599**

Orion II

- Intel® Xeon® processor 5600 series
- Up to 72TB in 4U
- 36 x 3.5 Hot-Swap SAS/SATA HDDs
- 1400W Gold Level Power Supply

Starting at **$3199**

1. Increased performance tested when comparing to the previous generations of Intel® Xeon® processors. Performance tests measure approximate performance of Intel products on specific computer systems; any difference in hardware, software or configuration may affect actual performance. For more information, visit www.intel.com/performance/server.

2. When compared to the previous generations with servers based on Intel® 32nm microarchitecture. Based on results on a server side Java* benchmark in conjunction with power consumption across a load line. Intel internal measurement (Jan. 15, 2010). Configuration details: server side Java benchmark in conjunction with power consumption across a load line.

© 2010, Intel Corporation. All rights reserved. Intel, the Intel logo, Intel Xeon, and Xeon Inside are trademarks of Intel Corporation in the U.S. and other countries.
*Other names and brands may be claimed as the property of others.

Copyright 2010 iXsystems, Inc. All rights reserved. Pricing will vary based on the specific configuration needed.

diff -u

WHAT'S NEW IN KERNEL DEVELOPMENT

Paul E. McKenney has worked up a patch to include a more precise **version number** in the config data, so if you're running a kernel built from a git repository, you'll be able to identify the source tree precisely, even if it's in between officially released versions. In this case, the version number will look something like 2.6.33-01836-g90a6501. Isn't it beautiful? His code actually went through numerous revisions to make sure it derived the version number in a safe way that wouldn't cause other scripts to choke and to give users the option of setting environment variables to control whether full version information should be included.

Dave Young has posted patches to change the patch submission documentation to list **Gmail** as no longer useful for sending patches. In the past, Gmail apparently could be made to send patches cleanly by jumping through a couple hoops, but now that's no longer the case. Gmail converts tabs to spaces, automatically wraps long lines and will 64-bit encode messages that have non-ASCII characters. Any one of those features would be enough to corrupt a patch file. Now, it's possible to configure **Firefox** to edit the e-mail with an external editor, and in the past, Gmail would send the edited text instead of using its own editor. But, with the introduction of the line-wrapping feature, Gmail apparently wraps lines even when an external editor is used. The documentation used to explain the workaround involving the external editor, but Dave's patch now simply lists the various issues and states that Gmail shouldn't be used for sending patches to the **linux-kernel mailing list**.

Eric W. Biederman has changed the way **/dev/console** is created. The old way was to wait until the filesystem containing the /dev directory had been mounted and then mount /dev/console there. The problem with that is if you ever want to unmount the filesystem, you can run into problems if /dev/console is still open. Eric's patch mounts /dev/console as part of **rootfs**—still in the same location, still called /dev/console, but just as part of rootfs instead of whatever filesystem you choose to mount for your running system. Very, very few power users may have to adjust the way they do things slightly as a result of this patch. Everyone else should notice nothing at all, or they may notice in some situations, certain problems that used to crop up don't anymore.

Christine Caulfield has marked herself as no longer maintaining the **DECnet** networking layer and has marked that code as orphaned instead of maintained. With the decnet mailing list totally silent, her theory is that the only users are running older kernels and are happy with it as is. The DECnet networking protocols originally were used in the 1970s to connect **PDP-11s**. They were published as open standards, paving the way for Linux's DECnet implementation decades later.

—ZACK BROWN

Create BillyBobBuntu with Reconstructor

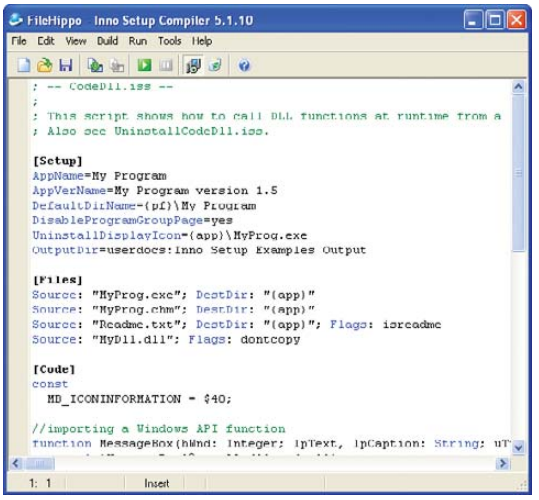
One glance at DistroWatch will prove that Linux users like to roll their own distributions. Heck, there's even a distribution called Linux From Scratch, which you'd think would just be called Linux! If you have been itching to roll your own distribution but feared it was too complicated, Reconstructor (**www.reconstructor.org**) might be exactly what you need.

I've written about Reconstructor before on the *Linux Journal* Web site (**www.linuxjournal.com/content/reconstructor-when-you-lose-your-restore-cd**), and more recently, Ross Larson wrote a follow-up on how the project has progressed (**www.linuxjournal.com/content/howto-customized-live-dvds-reconstructors-web-ui**). One interesting new feature is that you can build your own distribution from a Web-based distro builder. Surfing over to **build.reconstructor.org** (and creating an account) allows you to build a custom Linux distribution and then download it.

I do have one request: please don't start a new Linux distribution to compete with all the others. We already have plenty!

—SHAWN POWERS

NON-LINUX FOSS



Inno Setup Install Script

Whether you think making each program have its own installer is a bug or a feature, in the Windows world, it's the norm. So, if you're porting open-source code to Windows, at some point, you have to think about creating an installer.

Inno Setup is a free and open-source installer for Windows programs. It's been around since 1997 and is written in Delphi Pascal. Inno Setup is driven by a script that you provide, allowing Inno Setup to create an installer for your program. The script is much like an INI file. You provide simple name/value pairs that drive the creation of the installer. For more complex scenarios, Inno Setup contains its own built-in Pascal compiler for creating real "code" sections in the script.

Inno Setup has a long list of supported features: support for 64-bit applications, customizable setup types, integrated uncompressing of installed files, creation of shortcuts, creation of registry entries, running programs before/during/after the install, password protection, digital signing and much more. See the Web site (**www.jrsoftware.org/isinfo.php**) for more information.

Inno Setup runs on all modern versions of Windows. It creates an uninstaller as well as an installer and packages it all up in a single EXE for easy distribution. At the time of this writing, Inno Setup is at version 5.3.8, released February 18, 2010.

—MITCH FRAZIER

Save Your Favorite Articles

Did you know you can save your favorite LinuxJournal.com articles to reference later? Just click "Mark this as a favorite" at the bottom of any post, and you'll see it on your user profile. When you click your favorites tab, you can search your favorites for easy reference. Now, you can keep track of all the useful articles you come across on LinuxJournal.com in a sort of recipe box. Visit any author or reader profiles to see their favorite articles as well. We hope this makes it easier for you to recall specific info on the site. I'd love to hear how this feature is working for you, so feel free to drop me a line at webmistress@linuxjournal.com. See you on-line!

—KATHERINE DRUCKMAN

LJ Index June 2010

1. Millions of developers in the world: **15.2**
2. Number of lines of code produced per developer per day: **10**
3. Millions of lines of code produced per year by all developers: **31,616.0**
4. Millions of lines of code produced per minute by all developers: **0.32**
5. Millions of lines of code in kernel version 2.6.32: **12.99**
6. Minutes required to rewrite the Linux kernel if all developers pitched in: **41**
7. Millions of lines of code in the average Linux distro: **204.50**
8. Hours required to rewrite the average Linux distro if all developers pitched in: **10.6**
9. Number of the top 10 fastest computers in the world that run Linux: **10**
10. Number of the top 10 fastest computers in the world that run UNIX: **0**
11. Number of the top 10 fastest computers in the world that run Microsoft Windows: **0**
12. Number of the top 10 fastest computers in the world built by Cray: **2**
13. Number of the top 10 fastest computers in the world built by IBM: **4**
14. Number of the top 10 fastest computers in the world built by Sun: **2**
15. Number of the top 10 fastest computers in the world built by SGI: **1**
16. Number of the top 10 fastest computers in the world built by NUDT (China): **1**
17. Teraflop speed of world's fastest computer (Cray Jaguar at ORNL): **1,750**
18. Terabytes of memory in the world's fastest computer: **362**
19. Petabytes of disk storage in the world's fastest computer: **10**
20. Number of Opteron processor cores in the fastest computer in the world: **224,256**

Sources: *1: Evans Data / 2: Frederick P. Brooks in "The Mythical Man Month" / 3: #1 * #2 * 208 (208 working days/year) / 4: #1 * #2 / 8 / 60 (8-hour workday) / 5: www.h-online.com / 6: #5 / #4 / 7: Linux Foundation / 8: #6 * #4 / 60 / 9–16: TOP500 / 17–20: www.ornl.gov*

Maintaining Your System from the Command Line

Many Linux distributions use some form of packaging system to organize applications installed on a system. A formal packaging system lets you install, remove and, in general, maintain your software in a controlled and coherent way. The three main packaging systems that most distributions currently use are the Debian deb package, the Red Hat rpm package and the Slackware pkg package. They all have graphical utilities to interact with the packaging system, but what if you want to deal with the system on the command line? What if you're running a server or accessing a distant machine through SSH and don't want to deal with the overhead of X11? Let's look at how to do this for Debian-based systems.

First, you probably will want to install some software. The preferred way to do this is with the apt-get utility. apt-get is aware of the chain of dependencies between packages. If you want to install stellarium, simply run `apt-get install stellarium`, which downloads the relevant package file and all of its dependencies from a repository. What if you don't know the exact package name? Use the dpkg-query utility to query the package management system. So, if you know the package name has "kde" in it, you can list all the matching packages with `dpkg-query -l "*kde*"`. Remember, quote any search strings that have an asterisk (*), so you don't inadvertently make the shell try to expand them.

This works great for software available in the given repository. But, what if you want something not available? If you have a .deb file available for download, you can download it and install it manually. After downloading the file, install it by running `dpkg -i file_to_install.deb`.

dpkg works with the deb packaging system at a lower level than apt-get. With it, you can install, remove and maintain individual packages. If you have a group of packages to install, you might want to add the relevant repository to your list so that apt-get knows about it. The list of repositories is stored in the configuration file /etc/apt/sources.list. Each line has the form:

```
deb http://us.archive.ubuntu.com/ubuntu/ karmic main restricted
```

The first field tells apt-get what is available at this repository: deb is for binary packages and deb-src is for source packages. The second field is the URL to the repository (here, the Ubuntu repository). The third field is the repository name (in this case, the repository for Ubuntu's karmic version). The last fields are the sections from which to install packages. This example looks at the main and restricted sections when trying to install applications or resolve dependencies.

Now that you have installed some applications, you probably want to maintain and keep them updated, because every piece of software will have bugs or security issues that come to light over time. Developers always are releasing new versions to fix those issues and updating the relevant packages in the repositories. To update the list of software and versions on your system, run `apt-get update`. Once you've updated the list, tell apt-get to install the updates with `apt-get upgrade`. If you want a list of what is about to be upgraded, add the `-u` option: `apt-get upgrade -u`.

Sometimes, when a new version for a package comes out (like when a distribution releases a new version), the dependencies for said package might change too. In such cases, a straight upgrade might be confused, so use `apt-get dist-upgrade`. This command tries to deal with these changes in dependencies intelligently, adding and removing packages as necessary.

What if you've installed a package just to try it out and don't want it anymore? Remove a package with `apt-get remove stellarium`. This removes all the files installed as part of the stellarium package, but it leaves any configuration files intact and also doesn't deal with any extra packages installed because stellarium depended on them. If you want to remove a package completely, including all configuration files, run `apt-get purge stellarium`.

Installing and removing all this software can result in space-wasting cruft accumulating on your system. To recover some space, run `apt-get autoclean`. This removes the package .deb files from the local cache for packages that no longer can be downloaded (mostly useless packages). If you want to clean out the local cache completely and recover more space, run `apt-get clean`.

Although `remove` and `purge` will remove a package, what can you do about any dependencies installed for this package? If you run `apt-get autoremove`, you can uninstall all packages that were installed as dependencies for other packages and aren't needed anymore.

Another way of finding packages that are no longer needed is with the deborphan utility. First, you need to install it, with `apt-get install deborphan`. (Most distributions don't install it by default.) Once installed, running it with no command-line options gives a list of all packages in the libs and oldlibs sections that have no dependencies. Because no other package depends on those packages, you safely can use apt-get to remove or purge them. If you want to look in all sections, use the `-a` option. If you're trying to save space, ask deborphan to print out the installed sizes for these orphan packages by using the `-z` option. Then, you can sort them with `deborphan -z -a | sort -n`, which gives a list of packages you can safely uninstall, sorted by installed size from smallest to largest.

Each of the tools discussed above has many other options that you should research in the relevant man pages. Also, Red Hat-based systems have equivalent commands to help you manage rpm files.

—JOEY BERNARD

They Said It

We live in a society exquisitely dependent on science and technology, in which hardly anyone knows anything about science and technology.
—Carl Sagan

The most overlooked advantage to owning a computer is that if they foul up, there's no law against whacking them around a little.
—Porterfield

Any science or technology which is sufficiently advanced is indistinguishable from magic.
—Arthur C. Clarke

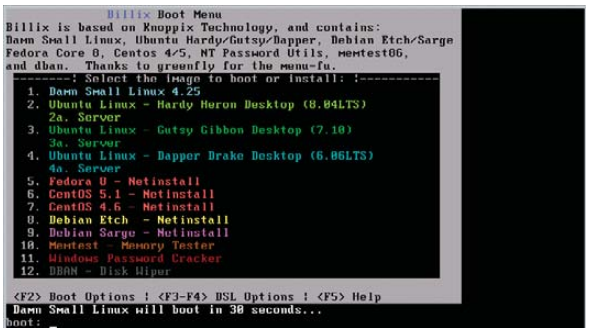
Any technology that is distinguishable from magic is not sufficiently advanced.
—Gregory Benford

Microsoft once made the mistake of broad-brushing Linux as an intellectual property quagmire. It made Microsoft headlines, but few friends: lawyers didn't believe it, customers didn't want to hear it, and competitors dared it to sue.

Years later, Microsoft still hasn't sued, but instead plods away at convincing the world, one patent cross-licensing agreement at a time, that everyone, everywhere owes it money for alleged violations of its IP in Linux.
—Matt Asay, Chief Operating Officer at Canonical

A year spent in artificial intelligence is enough to make one believe in God.
—Alan J. Perlis

Dual Booting, Not Just for Windows Users

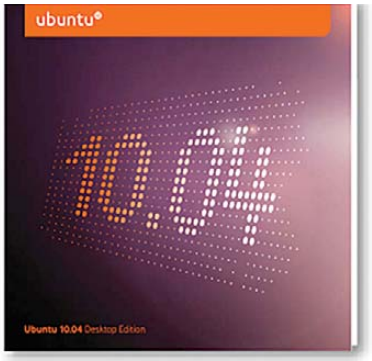


This is *LJ*'s Distribution issue, and it seems fair to note that programs like GRUB aren't only for those of us with one foot in the Windows world. Did you know you can run Fedora and Ubuntu on the same machine? Did you know you can run Fedora 10, Fedora 12, Ubuntu 8.04, Ubuntu 9.10, Slackware and Linux Mint all on the same machine?

One of the many great things about Linux is that it makes multiple installs simple! During the install process, carve off a hunk of hard drive, and most distributions happily will honor and respect your existing GRUB install. So if you can't decide which distribution you want to try, install them all! (Okay, if you install 20 distributions on one computer, you may start to run into problems keeping them straight!)

—SHAWN POWERS

One-Eyed, One-Horned, Flying Purple...Ubuntu?



With the latest iteration of its Linux distribution, Canonical has changed its branding a bit. Although we might all speculate why it has moved on from its traditional brown themes, sadly the reality often is less exciting than speculation. True, the rebranding is due to years of planning, research and marketing decisions, but I suspect a strong underlying set of reasons:

- UPS already had the corner of the brown market.
- Ubuntu's "Human" theme limited its interplanetary domination strategy.
- Mark Shuttleworth heard enough "scat" jokes as they pertain to the color brown.
- The color brown would clash with the upcoming orange overtones of the 10.10 version of Ubuntu, Marauding Marmaduke.

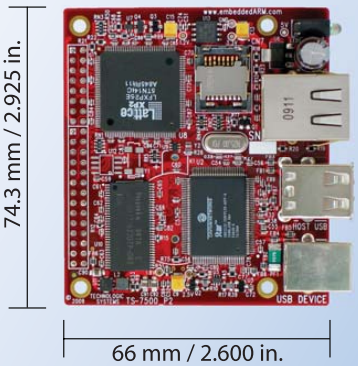
All joking aside, the rebranding is a refreshing new look for Ubuntu. Whether it will have any effect on the marketability of Canonical's flagship product remains to be seen. For those of us who were just about browned-out though, I think it's safe to say, "Bring on the purple!"

—SHAWN POWERS

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

Advanced MongoDB

A look at some of MongoDB's features, such as indexing and object relationships.

REUVEN M. LERNER

Last month, I started discussing MongoDB, an open-source non-relational "document-based" database that has been growing in popularity during the past year. Unlike relational databases, which store all information in two-dimensional tables, MongoDB stores everything in something akin to a set of hash tables.

In a relational database, you can be sure that every record (that is, row) in a table has the same number and set of columns. By contrast, MongoDB is schema-less, meaning there is no enforcement of such rules on columns. Two records in a MongoDB collection might have identical keys, or they might have no two keys in common. Ensuring that the keys are meaningful, and that they will not be prone to abuse or error, is the programmer's responsibility.

Working with MongoDB turns out to be fairly straightforward, as I showed in several examples last month. Once you have set up a database and a collection, you can add, remove and modify records using a combination of objects in your favorite language and the MongoDB query language.

The fact that it's easy to work with MongoDB doesn't mean that it's lacking in high-powered features, however. This month, I describe some of the features you're likely to use if you incorporate MongoDB into your applications, such as indexing and object relationships. If you're like me, you'll see there is a lot to like; plus, using MongoDB prods you to think about your data in new and different ways.

Indexing

As I explained last month, MongoDB has its own query language, allowing you to retrieve records whose attributes match certain conditions. For example, if you have a book database, you might want to find all books with a certain title. One way to perform such a retrieval would be to iterate over each of the records, pulling out all those that precisely match the title in question. In Ruby, you could express this as:

```
books.find_all {|b| b.title == search_title}
```

The problem with this approach is that it's quite slow. The system needs to iterate over each of the items, which means as the list of books grows, so too will the time it takes to find what you're seeking.

The solution to this problem, as database programmers have long known, is to use an index. Indexes come in various forms, but the basic idea is that they allow you to find all records with a particular value for the title immediately (or any column field), without having to scan through each of the individual records. It should come as no surprise, then, that MongoDB supports indexes. How can you use them?

Continuing with this book example, I inserted about 43,000 books into a MongoDB collection. Each inserted document was a Ruby hash, storing the book's ISBN, title, weight and publication date. Then, I could retrieve a book using MongoDB's client program, which provides an interactive JavaScript interface:

```
   ./bin/mongo atf
> db.books.count()
   38202
> db.books.find({isbn:'9789810185060'})
   { "_id" : ObjectId("4b8fca3ef23f3c614600a8c2"),
     "title" : "Primary Mathematics 4A Textbook",
     "weight" : 40,
     "publication_date" : "2003-01-01",
     "isbn" : "9789810185060" }
```

The query certainly seems to execute quickly enough, but if there were millions of records, it would slow down quite a bit. You can give the database server a speed boost by adding an index on the isbn column:

```
> db.books.ensureIndex({isbn:1})
```

This creates an index on the isbn column in ascending order. You also could specify -1 (instead of 1) to indicate that the items should be indexed in descending order.

Just as a relational database automatically puts an index on the "primary key" column of a table, MongoDB automatically indexes the unique _id attribute on a collection. Every other index needs to be created manually. And indeed, now if you get a list of the indexes, you will see that not only is the isbn column indexed, but so is _id:

```
> db.books.getIndexes()
   [
      {
```

```
        "name" : "_id_",
        "ns" : "atf.books",
        "key" : {
                "_id" : ObjectId("000000000000000000000000")
        }
    },
    {
        "ns" : "atf.books",
        "key" : {
                "isbn" : 1
        },
        "name" : "isbn_1"
    }
]
```

Now you can perform the same query as before, requesting all of the books with a particular ISBN. You won't see any change in your result set; however, you should get a response more quickly than before.

You also can create a compound index, which looks at more than one key:

```
> db.books.ensureIndex({title:1, weight:1})
```

Perhaps it doesn't make sense to combine the index for a book's title with that of its weight. Nevertheless, that's what I have now done in the example. If you later decide you don't want this index, you can remove it with:

```
> db.books.dropIndex('title_1_weight_1')
    { "nIndexesWas" : 3, "ok" : 1 }
```

Because I'm using the JavaScript interface, the response is a JSON object, indicating that there used to be three indexes (and now there are only two), and that the function executed successfully. If you try to drop the index a second time, you'll get an error message:

```
> db.books.dropIndex('title_1_weight_1')
    { "errmsg" : "index not found", "ok" : 0 }
```

Enforcing Uniqueness

Indexes not only speed up many queries, but they also allow you to ensure uniqueness. That is, if you want to be sure that a particular attribute is unique across all the documents in a collection, you can

SILICON MECHANICS

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

intel Xeon inside™

Powerful.
Intelligent.

Dominic and Maddison are logistics and shipping Experts for Silicon Mechanics. That's especially important recently. They have geared up to deliver the newest Silicon Mechanics rackmount servers and storage products with next-generation Intel CPU technology: the Intel® Xeon® Processor 5600 Series.

They are both excited to be shipping products that take advantage of the increased performance and decreased energy consumption made possible by features like 6-core CPUs with up to 12 threads and 12 MB of cache, and Intel® Turbo Boost Technology. No, we can't put the Experts themselves into the boxes we deliver, but you can be sure that every one of our products contains our Experts' commitment to quality and service, and that includes packaging and shipping.

When you partner with Silicon Mechanics, you get more than increased performance and improved energy efficiency — you get Experts like Maddison and Dominic.

For more information about products featuring the Intel Xeon Processor 5600 Series, visit www.siliconmechanics.com/5600

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.

define the index with the "unique" parameter.

For example, let's get a record from the current collection:

```
> db.books.findOne()
   {
      "_id" : ObjectId("4b8fc9baf23f3c6146000b90"),
      "title" : "\"Gateways to Academic Writing: Effective Sentences,
                  Paragraphs, and Essays\"",
      "weight" : 0,
      "publication_date" : "2004-02-01",
      "isbn" : "0131408887"
   }
```

If you try to insert a new document with the same ISBN, MongoDB won't care:

```
> db.books.save({isbn:'0131408887', title:'fake book'})
```

But in theory, there should be only one book with each ISBN. This means the database can (and should) have a uniqueness constraint on ISBN. You can achieve this by dropping and re-creating your index, indicating that the new version of the index also should enforce uniqueness:

```
> db.books.dropIndex("isbn_1")
   { "nIndexesWas" : 2, "ok" : 1 }
> db.books.ensureIndex({isbn:1}, {unique:true})
   E11000 duplicate key errorindex: atf.books.$isbn_1
   ➥dup key: { : "0131408887" }
```

Uh-oh. It turns out that there are some duplicate ISBNs in the database already. The good news is that MongoDB shows which key is the offender. Thus, you could go through the database (either manually or automatically, depending on the size of the data set) and remove this key, re-try to create the index, and so on, until everything works. Or, you can tell the ensureIndex function that it should drop any duplicate records.

Yes, you read that correctly. MongoDB will, if you ask it to, not only create a unique index, but also drop anything that would cause that constraint to be violated. I'm pretty sure I would not want to use this function on actual production data, just because it scares me to think that my database would be removing data. But in this example case, with a toy dataset, it works just fine:

```
> db.books.ensureIndex({isbn:1}, {unique:true, dropDups:true})
   E11000 duplicate key errorindex: atf.books.$isbn_1
   ➥dup key: { : "0131408887" }
```

Now, what happens if you try to insert a non-unique ISBN again?

```
> db.books.save({isbn:'0131408887', title:'fake book'})
   E11000 duplicate key errorindex: atf.books.$isbn_1
   ➥dup key: { : "0131408887" }
```

You may have as many indexes as you want on a collection. Like with a relational database, the main cost of an index is obvious when you insert or update data, so if you expect to insert or update your documents a great deal, you should carefully consider how many indexes you want to create.

A second, and more subtle, issue (referenced in David Mytton's blog post—see Resources) is that there is a namespace limit in each MongoDB database, and that this namespace is used by both collections and indexes.

Combining Objects

One of the touted advantages of an object database—or a "document" database, as MongoDB describes itself—is that you can store just about anything inside it, without the "impedance mismatch" that exists when storing objects in a relational database's two-dimensional tables. So if your object contains a few strings, a few dates and a few integers, you should be just fine.

However, many situations exist in which this is not quite enough. One classic example (discussed in many MongoDB FAQs and interviews) is that of a blog. It makes sense to have a collection of blog posts, and for each post to have a date, a title and a body. But, you'll also need an author, and assuming that you want to store more than just the author's name, or another simple text string, you probably will want to have each author stored as an object.

So, how can you do that? The simplest way is to store an object along with each blog post. If you have used a high-level language, such as Ruby or Python before, this won't come as a surprise; you're just sticking a hash inside a hash (or if you're a Python hacker, then a dict inside of a dict). So, in the JavaScript client, you can say:

```
> db.blogposts.save({title:'title',
                     body:'this is the body',
                     author:{name:'Reuven',
                     ➥email:'reuven@lerner.co.il'} })
```

Remember, MongoDB creates a collection for you if it doesn't exist already. Then, you can retrieve your post with:

```
> db.blogposts.findOne()
   {
          "_id" : ObjectId("4b91070a9640ce564dbe5a35"),
          "title" : "title",
          "body" : "this is the body",
          "author" : {
```

```
                "name" : "Reuven",
                "email" : "reuven@lerner.co.il"
        }
}
```

Or, you can retrieve the e-mail address of that author with:

```
> db.blogposts.findOne()['author']['email']
  reuven@lerner.co.il
```

Or, you even can search:

```
> db.blogposts.findOne({title:'titleee'})
  null
```

In other words, no postings matched the search criteria.

Now, if you have worked with relational databases for any length of time, you probably are thinking, "Wait a second. Is he saying I should store an identical author object with each posting that the author made?" And the answer is yes—something that I admit gives me the heebie-jeebies. MongoDB, like many other document databases, does not require or even expect that you will normalize your data—the opposite of what you would do with a relational database.

The advantages of a non-normalized approach are that it's easy to work with in general and is much faster. The disadvantage, as everyone who ever has studied normalization knows, is that if you need to update the author's e-mail address, you need to iterate over all the entries in your collection—an expensive task in many cases. In addition, there's always the chance that different blog postings will spell the same author's name in different ways, leading to problems with data integrity.

If there is one issue that gives me pause when working with MongoDB, it is this one—the fact that the data isn't normalized goes against everything that I've done over the years. I'm not sure whether my reaction indicates that I need to relax about this issue, choose MongoDB only for particularly appropriate tasks, or if I'm a dinosaur.

MongoDB does offer a partial solution. Instead of embedding an object within another object, you can enter a reference to another object, either in the same collection or in another collection. For example, you can create a new "authors" collection in your database, and then create a new author:

```
> db.authors.save({name:'Reuven', email:'reuven@lerner.co.il'})
```


Small, Portable Devices with Ubuntu Linux

ubuntu solution provider

Small Form Factor Intel® Atom™ Platform

No fans, no moving parts. Just quiet, reliable operation.
Incredibly tiny (0.6 L); takes up minimal desktop space.

Low-Profile Intel® Atom™ Industrial System

Small footprint platform featuring solid state storage.
System is less than 1.5" thick, yet rugged and sturdy.

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.

LOGIC SUPPLY

Learn More > www.logicsupply.com/linux

© 2010 Logic Supply, Inc. All products and company names listed are trademarks or trade names of their respective companies.

```
> a = db.authors.findOne()
   {
            "_id" : ObjectId("4b910a469640ce564dbe5a36"),
            "name" : "Reuven",
            "email" : "reuven@lerner.co.il"
   }
```

Now you can assign this author to your blog post, replacing the object literal from before:

```
> p = db.blogposts.findOne()
> p['author'] = a

> p
   {
            "_id" : ObjectId("4b91070a9640ce564dbe5a35"),
            "title" : "title",
            "body" : "this is the body",
            "author" : {
                     "_id" : ObjectId("4b910a469640ce564dbe5a36"),
                     "name" : "Reuven",
                     "email" : "reuven@lerner.co.il"
            }
   }
```

Although the blog post looks similar to what you had before, notice that it now has its own "_id" attribute. This shows that you are referencing another object in MongoDB. Changes to that object are immediately reflected, as you can see here:

```
> a['name'] = 'Reuven Lerner'
   Reuven Lerner
> p
   {
            "_id" : ObjectId("4b91070a9640ce564dbe5a35"),
            "title" : "title",
            "body" : "this is the body",
            "author" : {
                     "_id" : ObjectId("4b910a469640ce564dbe5a36"),
                     "name" : "Reuven Lerner",
                     "email" : "reuven@lerner.co.il"
            }
   }
```

See how the author's "name" attribute was updated immediately? That's because you have an object reference here, rather than an embedded object.

Given the ease with which you can reference objects from other objects, why not do this all the time? To be honest, this is definitely my preference, perhaps reflecting my years of work with relational databases. MongoDB's authors, by contrast, indicate that the main problem with this approach is that it requires additional reads from the database, which slows down the data-retrieval process. You will have to decide what trade-offs are appropriate for your needs, both now and in the future.

Resources

The main site for MongoDB, including source code and documentation, is at **mongodb.org**. A reference guide to the interactive, JavaScript-based shell is at **www.mongodb.org/display/DOCS/dbshell+Reference**.

For an excellent introduction to MongoDB, including some corporate background on 10gen and how it can be used in your applications, listen to episode 105 of the "FLOSS Weekly" podcast. I found the podcast to be both entertaining and informative.

Another good introduction is from John Nunemaker, a well-known blogger in the Ruby world: **railstips.org/blog/archives/2009/06/03/what-if-a-key-value-store-mated-with-a-relational-database-system**.

Mathias Meyer wrote a terrific introduction and description of MongoDB on his blog: **www.paperplanes.de/2010/2/25/notes_on_mongodb.html**.

Because MongoDB is a "document" database, you might be wondering if if there is any way to generate a full-text index on a document. The answer is "kind of", with more information and hints available at **www.mongodb.org/display/DOCS/Full+Text+Search+in+Mongo**.

Finally, David Mytton recently wrote a blog post, in which he described some of the issues he encountered when using MongoDB in a production environment: **blog.boxedice.com/2010/02/28/notes-from-a-production-mongodb-deployment**.

Conclusion

MongoDB is an impressive database, with extensive documentation and drivers. It is easy to begin working with MongoDB, and the interactive shell is straightforward for anyone with even a bit of JavaScript and database experience. Indexes are fairly easy to understand, create and apply.

Where things get tricky, and even sticky, is precisely in the area where relational databases have excelled (and have been optimized) for decades—namely, the interactions and associations among related objects, ensuring data integrity without compromising speed too much. I'm sure MongoDB will continue to improve in this area, but for now, this is the main thing that bothers me about MongoDB. Nevertheless, I've been impressed by what I've seen so far, and I easily can imagine using it on some of my future projects, especially those that will have a limited number of cross-collection references.■

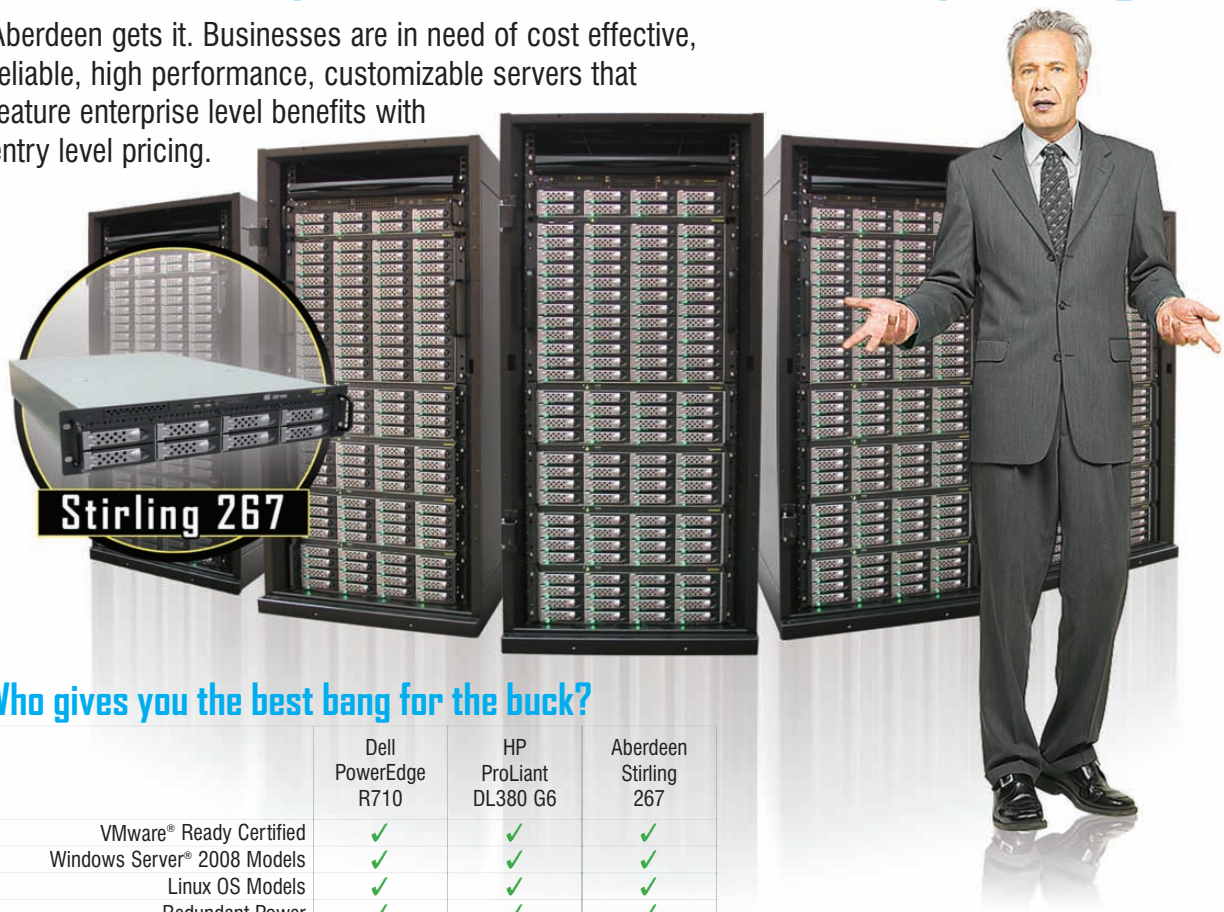
Reuven M. Lerner is a longtime Web developer, trainer, and consultant. He is a PhD candidate in Learning Sciences at Northwestern University. Reuven lives with his wife and three children in Modi'in, Israel.

ABERDEEN
SERVERS AND STORAGE

WHAT'S THE DEAL WITH THOSE GUYS?

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

Aberdeen gets it. Businesses are in need of cost effective, reliable, high performance, customizable servers that feature enterprise level benefits with entry level pricing.



Who gives you the best bang for the buck?

	Dell PowerEdge R710	HP ProLiant DL380 G6	Aberdeen Stirling 267
VMware® Ready Certified	✓	✓	✓
Windows Server® 2008 Models	✓	✓	✓
Linux OS Models	✓	✓	✓
Redundant Power	✓	✓	✓
Hardware RAID 0, 1, 5 & 6	✓	✓	✓
SAS / SATA Drive Support	✓	✓	✓
Available with 2TB Drives	✓	✗	✓
Out of Band RAID Management	✗	✗	✓
JBOD Storage Expansion	✗	✗	✓
Dual Intel® Xeon® Processors	E5504 2GHz	E5504 2GHz	E5504 2GHz
Memory	6GB	6GB	6GB
PCI-E Expansion Slots	4	6	7
Hot-Swap Drive Bays	6	6	8
Maximum Capacity	12TB	6TB	16TB
Configured Capacity	3TB	3TB	3TB
Warranty	3 Years	3 Years	5 Years
Price	**$4,462**	**$5,338**	**$3,995**

intel Xeon inside

Powerful.
Intelligent.

Prices for the above specific configurations obtained from the respective websites on Jan. 27, 2010. 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.aberdeeninc.com/abpoly/abterms.htm. lj033

888-297-7409
www.aberdeeninc.com/lj033

DAVE TAYLOR

Simple Scripts to Sophisticated HTML Forms

Building on the Yahoo Movies search form script.

Last month, we looked at how to convert an HTML form on a page into a shell script with command flags and variables that let you have access to all the features of the search box. We tapped into Yahoo Movies and are building a script that offers up the key capabilities on the search form at **movies.yahoo.com/mv/advsearch**.

The script we built ended up with this usage statement:

```
USAGE: findmovie -g genre -k keywords -nrst title
```

So, that gives you an idea of what we're trying to do. Last month, we stopped with a script that offered the capabilities above and could open a Web browser with the result of the search using the open command.

Now, let's start with a caveat: open is a Mac OS X command-line script that lets you launch a GUI app.

What if the user wants the option of dumping the data to the command line instead of launching a browser?

Just about every other Linux/UNIX flavor has a similar feature, including if you're running the X Window System. In fact, with most of them, it's even easier. A typical Linux version of "open a Web browser with this URL loaded" might be as simple as:

```
firefox http://www.linuxjournal.com/ &
```

That's easily done, even in a shell script.

Actually, if you're going to end a script by invoking a specific command, the best way to do it is to "exec" the command, which basically replaces the script with the app you've specified, so it's not still running and doesn't even need to exit. So in that case, it might look like `exec firefox "$url"` as the last line of the script.

This month, I want to go back and make our script do more interesting things. For now, an invocation like:

```
./findmovie.sh -g act evil
```

produces a command from the last few lines in the script:

```
echo $baseurl${params}\&p=$pattern
exec open -a safari "$baseurl${params}\&p=$pattern"
```

that ends up pushing out this:

```
http://movies.yahoo.com/mv/
➥search?yr=all&syn_match=all&adv=y&type=feature&gen=act&p=evil
```

It's pretty sophisticated!

Letting the User Dump the Resultant Data

What if the user wants the option of dumping the data to the command line instead of launching a browser? We can address that by adding a -d dump command flag into the getopt block:

```
while getopts "dg:k:nrst" arg
do
  case "$arg" in
    d ) dump=1 ;;
    g ) params="${params:+$params&}gen=$OPTARG" ;;
```

To dump the data, we'll enlist the powerful curl command, as we've done in the past. The program has zillions of options, but as we're just interested in the raw output, we can ignore them all (fortunately) except for --silent, which hides status updates, leaving the conditional:

```
if [ $dump -eq 1 ] ; then
  exec /usr/bin/curl --silent
"$baseurl${params}\&p=$pattern"
else
  exec open -a safari "$baseurl${params}\&p=$pattern"
fi
```

But, that generates a huge amount of data, including all the HTML needed to produce the page

in question. Let's spend just a minute looking closely at that output and see if there's a way to trim things at least a bit.

It turns out that every movie title that's matched includes a link to the movie's information on the Yahoo Movies site. Those look like:

```
<a href="http://movies.yahoo.com/movie/1809697875/info">Resident Evil
```

So, that's easy to detect. Better, we can use a regex expression with grep and skip a lot of superfluous data too:

```
cmd | grep '/movie/.*info'
```

That comes close to having only the lines that match individual movies, but to take this one step further, let's remove the false matches for dvdinfo, because we're not interested in the links to DVD release info. That's a `grep -v`:

```
cmd | grep '/movie/.*info' | grep -v dvdinfo
```

Now, let's have a quick peek at comedies that have the word "funny" in their titles:

```
./findmovie.sh -d -g com funny | grep '/movie/.*info'
 ➥| grep -v dvdinfo |  head -3

<td><a href="http://movies.yahoo.com/movie/1810041785/info">
<b>Funny</b> People (2009)</a><br>

<td><a href="http://movies.yahoo.com/movie/1809406735/info">What's So
 <b>Funny</b> About Me? (1997)</a><br>

<td><a href="http://movies.yahoo.com/movie/1808565885/info">That
 <b>Funny</b> Feeling (1965)</a><br>
```

Okay, so the first three films in that jumble of HTML are *Funny People*, *What's So Funny About Me?* and *That Funny Feeling*.

From this point, you definitely can poke around and write some better filters to extract the specific information you want. The wrinkle? Like most other sites, Yahoo Movies chops the results into multiple pages, so what you'd really want to do is identify how many pages of results there are going to be and then grab the results from each, one by one. It's tedious, but doable.

How Many Matches?

Let's look at a more interesting subset, instead, by adding a -c flag to have it output just a count of how many films match the specified criteria, you've given the command instead.

To do that, we don't need to go page by page, but just identify and extract the value from the match count on the page. For the comedies with "funny" in the title, the line on the page looks like this: "< Prev | 1 - 20 of 37 | Next 17 >".

What we need to do is crack the HTML and look at the source to the link to "next 17" and see if it's extractable (is that a word?):

```
./findmovie.sh -d -g com funny | grep -i "next 17" | head -1
<td align=right><font face=arial size="-2"><nobr>
➥&lt; Prev | <b>1 - 20</b>
➥ of <b>37</b> | <span
➥class="yperlink"><ai href="/mv/search?p=funny&yr=all
➥&gen=com\&syn_match=all&adv=y&type=feature
➥&n=17&b=21&h=s">Next 17</a> &gt;
➥ </nobr></span></span></font></td></tr>
```

Well that's ugly. You'd think Yahoo didn't want to make this easy or something! It turns out though that this is a pretty tricky task, because if there are no matches, the link doesn't show up, and instead you see "Sorry, no matches were found". If there are less than 20 matches, you see "Next >", but it's not a clickable link, so it's not going to be so easy!

Given that I'm out of space, let's defer this topic until next month. Meanwhile, look at the source to various searches yourself and see if anything comes to mind. Otherwise, it'll be brute force!■

Dave Taylor has been hacking shell scripts for a really long time, 30 years. 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.

MICK BAUER

Linux VPNs with OpenVPN, Part V

Tips for success in using OpenVPN for secure remote access.

In my four previous columns, I showed, in painstaking detail, how to set up OpenVPN to allow remote users to create secure remote-access connections—Virtual Private Network (VPN) tunnels—over the Internet back to your personal or corporate network. By now, you should understand how VPN technologies in general, and TLS/SSL-based VPNs in specific, work and how to create working server and client configurations for OpenVPN.

This month, I wrap up the series, with some miscellaneous but important notes about the previous columns' client-server scenario, including instructions on enabling IP forwarding, some tips on using a Web proxy and enforcing DNS use through the tunnel, and on "hiding" all VPN clients' IP addresses behind that of your OpenVPN server.

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
```

Review

Throughout this series, I've been implementing the OpenVPN server configuration shown in Listing 1, which causes OpenVPN to run in server mode. In my example scenario, I've got only one remote user connecting to this OpenVPN server, but if you have more, you should edit the `max-clients` parameter accordingly. Remember, because I've also set fairly liberal tunnel timeouts in order to minimize the odds that a tunnel will go down due to network problems, you should add 1 or 2 to the actual number of maximum concurrent client connections you expect.

The other setting in Listing 1 that I need to review is `push "redirect-gateway def1 bypass-dhcp"`, which pushes the OpenVPN's local default gateway setting to all clients. This has the effect of causing VPN clients to route all their Internet traffic through the VPN tunnel, which (as I discuss shortly) has important security benefits.

The client configuration file that corresponds to Listing 1 is shown in Listing 2. This file works equally well on Linux and Windows client systems. Remember that the parameter `remote` specifies the IP address or hostname of your OpenVPN server and the port on which it's accepting connections.

Remember also that the files ca.crt, minion.crt, minion.key and ta.key specified by the parameters `ca`, `cert`, `key` and `tls-auth`, respectively, need to be generated beforehand and placed alongside the configuration file itself in /etc/openvpn. The certificate and key specified by `ca` and `cert` should be unique for each client system!

Again, the purpose of the server configuration in Listing 1 and the client configuration in Listing 2 is to allow a remote user to connect from over the Internet back to the "home" network on which the OpenVPN server resides. (This may or may not be your residence. By home network, I mean "trusted corporate or personal network", as opposed to the remote network from which you're trying to connect.) Last month, however, I forgot to mention a critical step that you must perform on your OpenVPN server if you want remote clients to be

Listing 2. Client's client.ovpn File

```
client
proto udp
dev tun

remote 1.2.3.4 1194

nobind

ca ca.crt
cert minion.crt
key minion.key

ns-cert-type server
tls-auth ta.key 1

cipher BF-CBC
comp-lzo

user nobody
group nogroup
persist-key
persist-tun

mute-replay-warnings

verb 3
mute 20
```

able to communicate with anything besides the server itself: enabling IP forwarding.

Enabling IP Forwarding

By default, almost any Linux system is configured not to allow network packets entering one network interface to be forwarded to and sent out of a different network interface. This is a Linux security feature. It helps reduce the likelihood of your Linux system linking different networks together in undesirable or unintended ways.

But, generally you do want an OpenVPN server to link different networks. The exceptions to this are if:

1. All services and resources your remote users need are housed on the OpenVPN server itself.

2. It's possible to run proxy applications on the OpenVPN server that can proxy connections to services not hosted on it.

In the first case, once remote users have connected to the OpenVPN server successfully, they can connect to other services hosted on that server by targeting the server's real/local IP address rather than its Internet-facing address. For example, the client configuration in Listing 2 is targeting a server address of 1.2.3.4, which is Internet-routable. Suppose that this is actually a router or firewall address that is translated to your OpenVPN server's address 10.0.0.4.

To `ssh` to the OpenVPN server after you've established a tunnel to it, you'd target 10.0.0.4, not 1.2.3.4. The same would apply to Samba, NFS, HTTP/S or any other service running on the OpenVPN server.

In the second case, to reach other resources on the remote network, you would configure the applications running on your client system to use the OpenVPN server's real/internal address (10.0.0.4) as its proxy address. The best example of this is Squid. If all the resources you wanted to reach on your remote network involve Web services, you could run Squid on the OpenVPN server and configure your client's Web browser to use 10.0.0.4 as its proxy address (although this will work only when the tunnel is up).

In either of the above scenarios, you *don't* need IP forwarding enabled on the OpenVPN server, because all direct communication between VPN clients and your home network terminates on the OpenVPN server. If, however, your clients need to reach other things on the home network or beyond, *without* using the OpenVPN server as a proxy, you do need to enable IP forwarding.

This is very simple. To turn on IP forwarding without having to reboot, simply execute this command:

```
bash-$ sudo sysctl -w net.ipv4.ip_forward=1
```

To make this change persistent across reboots, uncomment the following line in /etc/sysctl.conf (you'll need to `su` to root or use sudo to edit this file):

```
net.ipv4.ip_forward=1
```

Web Proxies and VPN Clients

In talking about the value of using VPN software when using untrusted networks like WLAN hot spots, I've described the benefits of using your home network's Web proxy rather than surfing the Web directly through the untrusted network. From a policy-enforcement standpoint, this allows you to enforce whatever URL or content filtering with which your home network's proxy is configured; from an endpoint-security standpoint, it makes phishing and man-in-the-middle attacks harder.

On the downside, it also results in a somewhat slower Web browsing experience, because each user's Web traffic must traverse a longer, slower path than without the VPN tunnel in place. Also, making remote users use your corporate Web proxy without also configuring them to use your corporate DNS servers may fail to prevent man-in-the-middle attacks (in which DNS redirection is a common technique), giving a false sense of security.

I return to the DNS problem shortly, but how do you use Web proxies with OpenVPN? It's quite

simple. On the Web proxy itself, you simply need to make sure there's an Access Control List (ACL) allowing client connections from tunnel IPs. This is a moot question if your Squid server is running on a different box from the OpenVPN server, *and* the OpenVPN server is using Network Address Translation (NAT) to "hide" all tunnel-originated packets behind its own IP address (I discuss NAT shortly).

If, however, you are running the Web proxy on the OpenVPN server itself, you need an ACL. For Squid, you need to add something like this to /etc/squid/squid.conf:

```
acl openvpn_tunnels src 10.31.33.0/24
http_access allow openvpn_tunnels
```

The `acl` line defines an object named `openvpn_tunnels`, representing transactions whose source IP addresses fall between 10.31.33.1 and 10.31.33.254. The `http_access` line allows transactions initiating from this IP range. As with any other change you make to this file, you need to restart Squid for this ACL to take effect (`sudo /etc/init.d/squid restart`).

Your clients will, of course, need to be configured to use your Web proxy, but they target the same IP address regardless of whether they're connecting from afar via OpenVPN or connecting directly to your LAN. That is, if you're already having your users proxy all their Web traffic, no change to their Web browser settings should be necessary for them to use the same proxy through OpenVPN.

Enforcing DNS

If you're requiring all remote users to route all their Internet traffic through the VPN tunnel, it isn't enough to force them to use the remote network's default gateway. You also need to force them to use the remote network's DNS servers. Otherwise, a man-in-the-middle attack that involves DNS spoofing on the client side of the tunnel will succeed. Once a remote user's browser has been handed a phishing site's IP address for a given URL, it doesn't matter whether it connects to that IP directly or through the VPN tunnel (unless, perhaps, the phishing site's IP address is on a blacklist enforced by your corporate Web proxy or firewall).

If your remote clients all run Windows, it's easy to enforce server-side DNS settings. Simply add the following line to your OpenVPN server's OpenVPN configuration file:

```
push "dhcp-option DNS 10.0.0.100"
push "dhcp-option DNS 10.0.0.120"
```

Of course, you should replace 10.0.0.100 and 10.0.0.120 with the addresses of the DNS servers you want your clients to use.

Unfortunately, this won't work for non-Windows clients. For Linux and other UNIX clients, you'll need to edit those client systems' /etc/resolv.conf files either manually or dynamically. The server-side configuration parameter `foreign_option_<I>n<I>` lets you pass data to tunnel-initiation scripts (`--up` scripts); for example, the line `foreign_option_1='dhcp-option DNS 10.0.0.100'` sends the line `dhcp-option DNS 10.0.0.100` to any defined "up" scripts, which can then act on that data.

The details of how all this works are out of the scope of this article. Suffice it to say that the OpenVPN man page describes how "up" scripts work, and the link to the update-resolv-conf script in the Resources for this article provides a script you can alter to rewrite /etc/resolv.conf to give precedence to your "home" network's DNS servers.

NAT and iptables on the OpenVPN Server

There's one more critical step necessary to allow remote users to route packets to the Internet through their VPN tunnels. You need to set up Network Address Translation (NAT) so that traffic entering your "home" network from VPN tunnels appears to originate from the OpenVPN server.

This is because the networks from which remote clients connect will have either different network IP addresses than your "home" network, in which case the odds are your "home" network infrastructure won't have a route to the remote clients, or they'll have the same network IP addresses, in which case it's quite possible that different hosts on opposite ends of the VPN tunnels will have the *same* host IP addresses!

Note that this problem plays out differently on "bridging" (Layer 2) VPN tunnels than on "routing" (Layer 3) VPN tunnels. Because all my examples so far have involved a routing VPN scenario, what I'm about to say regarding NAT applies to routed VPN tunnels.

So, the way to sidestep the problem of foreign IP addresses on remote clients' packets completely is simply to rewrite all packets entering the OpenVPN server's local network with the OpenVPN server's local IP address. To do so, add just one firewall rule, like this:

```
bash-$ sudo iptables -t nat -A POSTROUTING
➥-s 10.31.33.0/24 -o eth0 -j MASQUERADE
```

Note that as with any other time you execute the command `iptables`, this is not a persistent change. To make this rule persistent across reboots, you need to add an equivalent line to whatever configuration file or script controls firewalling on your OpenVPN server.

The OpenVPN man page has an entire section on firewalls (called "FIREWALLS") that contains lots of good information about managing iptables firewall rules on your OpenVPN server. Remember, any

VPN server is a security device. It's a good idea to run not just a single NAT rule, but a detailed set of filtering rules that restrict how people can connect to the server and to what systems your VPN clients may connect.

(Speaking of iptables, it's been a long time since I covered Linux's powerful firewall capabilities in this column. Look for an in-depth article on writing your own Linux firewall rules in a future column.)

Conclusion

This article and my previous four columns covered Virtual Private Network principles and architectures; described a few VPN technologies available for Linux and how SSL/TLS solutions differ from IPsec; covered OpenVPN server configuration, including how to generate and manage digital certificates; and described client configuration and usage; all for a simple remote-access usage scenario.

With all of that plus the practical use details I covered this month, you should be well on your way to a secure remote-access VPN solution using OpenVPN. If you decide to use OpenVPN instead or additionally to build network-to-network VPNs or to do a "bridging" OpenVPN solution, the OpenVPN man page, HOWTO and FAQ should make more sense to you now than they would have before reading these articles—all of which means, you no longer have any excuse to surf the Web through insecure wireless hot spots without protection!■

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**

OpenVPN FAQ: **openvpn.net/index.php/open-source/faq.html**

OpenVPN HOWTO: **www.openvpn.net/index.php/open-source/documentation/howto.html**

Ubuntu Community OpenVPN Page: **https://help.ubuntu.com/community/OpenVPN**

The update-resolv-conf Script by Thomas Hood and Chris Hanson: **www.subvs.co.uk/openvpn_resolvconf**

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

NVIDIA PARTNERFORCE TESLA PREFERRED PROVIDER

intel Xeon inside
Powerful. Intelligent.

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 the latest Intel® Xeon® Processor 5600 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. Populate the twelve DDR3 DIMM slots, and you end up with 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

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.

Dynamic Config Files with Nmap

KYLE RANKIN

Port scans aren't just for script kiddies and network troubleshooting. You also can use them to scan your network for clients and build your server configs dynamically.

The great thing about tools is that you often can misuse them for a completely different purpose. The end of a screwdriver makes a passable hammer; a butter knife can be a screwdriver, and even a paper clip can substitute for a key in a pinch. Normally, you probably think of nmap as a security tool. After all, it's ideal when you want to test a machine for open, vulnerable ports. The other day though, I realized nmap had another use—a way to scan my network and build a dynamic configuration file based on what machines replied to my scan.

Munin Is Trendy

This whole project started when I decided to deploy Munin across my servers so I could graph trending data for each machine on my network. Munin is a great tool for trending, because once you install the agent, it often will discover what services and statistics to monitor and graph automatically. The downside for me though was that I already had a network full of servers. It was bad enough that I had to install an agent on each machine, but I also had to build a giant configuration file on my Munin server by hand that listed each server it should monitor. Plus, any time I added a machine to the network, I had yet another step in my build process as I had to add that new server to my Munin config.

Although this script worked great for Munin configs, you also could use the same procedure to scan for any number of services and build a configuration.

I'm a big fan of automation, and I figured there must be some easier way to add all my machines to this file. When you look at a Munin configuration file, it seems ripe for automation:

```
dbdir    /var/lib/munin
htmldir /var/www/munin
logdir   /var/log/munin
rundir   /var/run/munin
tmpldir /etc/munin/templates

[web1.example.net]
        address web1.example.net

[web2.example.net]
        address web2.example.net

[db1.example.net]
        address db1.example.net

[db2.example.net]
        address db2.example.net
```

The syntax for a generic munin.conf file is pretty straightforward. First, a few directories are defined, and then each server is defined within a pair of brackets. Inside those brackets, you can assign a name to the server or just use the hostname. After that, the following line lists the hostname or IP address for that server. In the above example, I've defined four servers.

If I wanted to generate this configuration file automatically, I had to figure out some way to detect what servers were running Munin on my network. Munin makes this simple though, because each server has a Munin agent listening on port 4949 by default. All I had to do was use nmap to scan the network and list all the machines that had port 4949 open. I figured I could parse that output and append it to my munin.conf file, and then maybe make a vim macro to go through each line and format it.

Nmap with Grepable Output

The first step was to find the right nmap syntax so that it would scan my network and list all machines that were listening to port 4949. First, I tried the standard command:

```
$ nmap -p 4949 10.1.1.0/24
```

```
Starting Nmap 4.11 ( http://www.insecure.org/nmap/ )
 ➥at 2010-03-01 20:18 PST

Interesting ports on 10.1.1.1:
PORT      STATE  SERVICE
4949/tcp closed unknown
MAC Address: 00:00:0C:01:CD:05 (Cisco Systems)

Interesting ports on purple1.example.net (10.1.1.50):
PORT      STATE  SERVICE
4949/tcp closed unknown
MAC Address: 08:00:20:CF:9D:D7 (SUN Microsystems)

Interesting ports on web1.example.net (10.1.1.53):
PORT      STATE SERVICE
4949/tcp open  unknown
MAC Address: 00:50:56:92:34:02 (VMWare)

Interesting ports on web2.example.net (10.1.1.67):
PORT      STATE SERVICE
4949/tcp open  unknown
MAC Address: 00:30:48:A0:12:98 (Supermicro Computer)
. . .
```

As you can see, for each machine that nmap finds, it lists the IP, whether the port is open, and even tries to identify the type of machine. Even though you could grep out the machines with open ports from this output, it would be quite a pain to parse everything with the multi-line output. Instead, I used the -oG argument to nmap, which tells it to output in "grepable format", along with the - argument, which tells it to send that output to STDOUT. The result was much simpler to parse:

```
$ nmap -oG - -p 4949 10.1.1.0/24
# Nmap 4.11 scan initiated Mon Mar  1 20:26:45 2010 as:
 ➥nmap -oG - -p 4949
# 10.1.1.0/24
Host: 10.1.1.1 ()       Ports: 4949/closed/tcp/////
Host: 10.1.1.50 (purple1.example.net)     Ports: 4949/closed/tcp/////
Host: 10.1.1.53 (web1.example.net)        Ports: 4949/open/tcp/////
Host: 10.1.1.67 (web2.example.net)        Ports: 4949/open/tcp/////
. . .
```

Now I could just grep for "open", and I'd get a

All geeked up and nowhere to go?

While there are many Linux-related conferences, few celebrate the vibrant community and active participation that fuel its tremendous growth.

Bring your inner geek to the center of the Carolinas and celebrate Linux in the GNU/South.

SouthEast Linux Fest

June 12th and 13th, 2010
Spartanburg, South Carolina
southeastlinuxfest.org

Listing 1. Script to Build Configuration File

```
#!/usr/bin/perl

my $munin_dir = '/etc/munin';
my $munin_config = 'munin.conf';
my $munin_config_temp = 'munin.conf.tmp';
my $node_port = '4949';
my $nmap = "nmap -oG - -p ";
my %subnets = (
                  "10.1.1.0/24" => "VLAN1",
                  "10.1.5.0/24" => "VLAN5",
                  "10.1.6.0/24" => "VLAN6",
              );

# iterate through each subnet and perform the nmap scan
foreach $subnet (keys %subnets){
   open NMAP, "$nmap $node_port $subnet | grep open |"
   ➥or die "Can't run nmap: $!\n";
   while (<NMAP>){
      $ip = $host = "";
# parse out the hostname and IP address
      /Host: (\d+\.\d+\.\d+\.\d+) \((.*?)\)/;
      $ip = $1; $host = $2;

      next if($ip eq "");

# sometimes nmap doesn't do rDNS properly,
# get it manually in that case
      if($host eq ""){
         $host = `dig -x $ip +short` or $host = $ip;
         chomp $host;
         $host =~ s/\.$//;
      }

      $munin_hosts{$host} = $ip;
   }
   close NMAP;
}

# output to a temp file in case munin
# runs while this script is open
open OUTFILE, "> $munin_dir/$munin_config_temp" or die "Can't open
$munin_dir/$munin_config_temp: $!\n";

# first print out the standard header for the munin file
print OUTFILE <<END_HEAD;
dbdir   /var/lib/munin
htmldir /var/www/munin
logdir  /var/log/munin
rundir  /var/run/munin
tmpldir /etc/munin/templates

END_HEAD

# then print out the config for each host
foreach $host (sort keys %munin_hosts){
   print OUTFILE "\[$host\]\n\taddress $host\n";
   # add any extra munin options for each host here
   print OUTFILE "\n";
}
close OUTFILE;

system("mv $munin_dir/$munin_config_temp $munin_dir/$munin_config");
```

list of all machines running Munin:

```
$ nmap -oG - -p 4949 10.1.1.0/24 | grep open
Host: 10.1.1.53 (web1.example.net)      Ports: 4949/open/tcp/////
Host: 10.1.1.67 (web2.example.net)      Ports: 4949/open/tcp/////
```

Perl to the Rescue

Once I started working on the regular expressions to parse through this output and generate the syntax I needed, I realized I should ditch vim and just write a script that built the entire configuration file for me and run that script with cron. That way, I'd never have to add a new server again. The only challenge was that I had multiple subnets I wanted to scan, and I discovered that sometimes nmap didn't resolve the IP addresses into hostnames for me. Listing 1 shows the resulting script.

Other than the hashes and a little fun with regular expressions, the bulk of this script is basic Perl. Once I tested it a few times by hand and was comfortable with it, I went ahead and copied the script into /etc/cron.daily. Of course, on my real network, I've added a few other fancy touches. For instance, every server on my network has a DNS TXT record that says what the server does. It is a useful practice for many reasons, but in this case, I found that because I used the same TXT record for similar servers, I could look it up and use that to group servers together under that heading.

Although this script worked great for Munin configs, you also could use the same procedure to scan for any number of services and build a configuration. I could see scripts that generate configuration files for Nagios, programs that poll SNMP or any other program that monitors multiple servers over a known port.■

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.

O'REILLY®
OSCON®
Open Source Convention

July 19–23, 2010
Portland, Oregon

MAKE IT HAPPEN

www.oscon.com

"Last week at the O'Reilly Open Source Convention was truly epic. The Community Leadership Summit + OSCON 2009 epitomized everything I've ever wanted to do in community as a Microsoft employee."

—Sarah Ford, Microsoft

REGISTER NOW AND SAVE 15%
use discount code os10ljr

DIRK ELMENDORF

Organizing a Library

"What is a library? It's like Google made out of a tree." —tevoul on Reddit.com

At first glance, you might assume that I'm going to discuss code libraries in this article, but instead, I'm talking about an actual library—one made of books, magazines and other dead-tree sources of wisdom. I have always collected books, and each new project or pastime becomes an excuse to expand my library. I don't always know what I have or, more important, where a certain book is. I try to keep my library organized in a physical sense, but I've always wanted a system that kept better track of my books.

Goals and Deliciousness

The goals for this project are pretty straightforward. I need something that can track all of the books I own. A big part of my collection is in my library at home, but I also have a large set of technical books at my office. I'd love to be able to see images of the covers (à la Delicious Monster—a Mac program that originally inspired me to sort this out). I also need something to show me where in the library the book is physically—the cabinet and shelf would be nice. One last thing is data entry. I have several thousand books, and I'd prefer not to have to type in a lot of information.

First Stop: Actual Library Software

The cool part about the Open Source world is that you can access software that is way beyond the scale of what you need. In the case of this project, I found Koha. According to the Web site, "Koha is the first open-source Integrated Library System (ILS). In use worldwide, its development is steered by a growing community of libraries collaborating to achieve their technology goals." The project is targeted at actual libraries, which sounded like overkill, but I could not resist downloading and taking it for a spin.

I decided to play with the development version (as the last release was June 2009). The first step was to check out the code repository:

```
git clone git://git.koha.org/pub/scm/koha.git kohaclone
```

The repository actually had install instructions for several distributions. Because I'm running Ubuntu, I followed those instructions. Based on the differences between the Web site instructions for installing on Hardy (8.04) and the instructions in the development version, it looks like a number of packages outside the standard package tree have been added. That is a good sign, because it means installation will get easier and easier. Be warned though, Koha is built using Perl, and a few Perl libraries are not currently packaged in Jaunty. The instructions show you how to use CPAN to install them properly (although that means you will have CPAN versions that are not controlled by the package system—a side effect of working with CPAN). After following all the instructions and getting everything installed, I ran through the Web install to set up the database.

Once everything was up and "running", I was ready to dive in to the heady world of running my own library. After spending an inordinate amount of time figuring out that I needed to provide some default values for the library and the type of content I was going to track, I was ready to add my first book. Pulling up Koha's add form presents a huge page of options, most of which meant very little to me (such as Leader, Control Number Identified and Fixed Length Data Elements). I forged ahead by trying to search for one of my test books by ISBN. I had to do something called a Z39.50 search. This is a protocol used for getting book information from other libraries. In the process, I learned that I had to

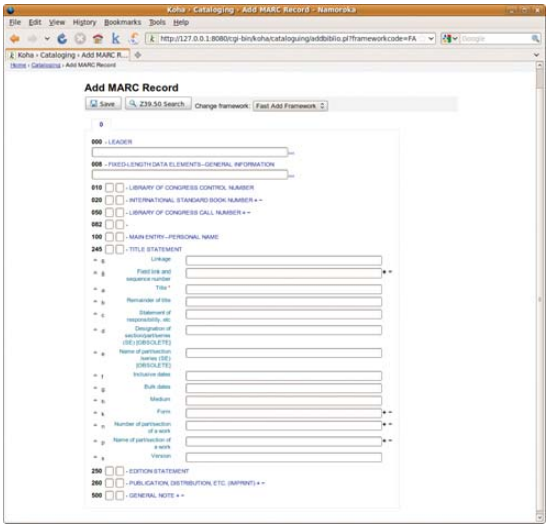


Figure 1. The Koha catalog screen—something tells me this is overkill.

add my own Z39.50 sources. I used the Library of Congress, because I figured it would have the most complete records (Settings are z390.loc.gov:7090 Database:Voyager - Syntax USMARC). Once all that was set up, I was able to add the book.

All of the above was a lot of work, and I added only a single book. As much as I would like to use an industrial-strength tool, the system was too confusing for me as a layperson (my wife was kind enough to point out that there is a reason it is called Library Science). If you want to see what a properly configured Koha system is capable of, go to the Plano ISD library system (see Resources), which is running a version of Koha. It shows the book covers and even has a shelf browser. So if you have your own public library, Koha is really neat, but I realized I needed something else.

"Personal" Is the Keyword

I eventually figured out that the problem with my previous searches for software was the omission of the word "personal". Adding that word narrows down the Google search a lot. As a result, I found two different options to consider: Alexandria and GCstar. Unlike Koha, both are available as Ubuntu packages. After dealing with the install guide for Koha, it was nice that all I had to do was `apt-get install`, and I could try them both (well, that was almost all I had to do). In the process of playing with these tools, I found another application called Tellico. It was nice to have several apps from which to choose.

Alexandria

Alexandria is a Ruby GNOME application for managing a book collection. The current official version is 0.6.5. Things got off to a very bumpy start with Alexandria. The default version in Jaunty is 0.6.3. It was not able to find either of the test books. Even worse than that, it crashed and exited when I tried to search by ISBN. Not one to give up easily, I ended up downloading a current beta version (0.6.6-beta1). There was a problem related to two Ruby libraries because I was installing it under Jaunty. To get everything to work, I had to install two gems (hpricot and htmlentities) and manually install the package:

```
sudo dpkg   --ignore-depends=libhpricot-ruby -i
➥Desktop/alexandria_0.6.6beta1.deb
```

The system relies on Amazon for some of the lookups. Due to a change in Amazon's policy, I had to sign up to get my own Amazon AWS access key. An explanation and link are available on the Alexandria Web site (see Resources). Technically, I could have removed Amazon as a provider and skipped this step.

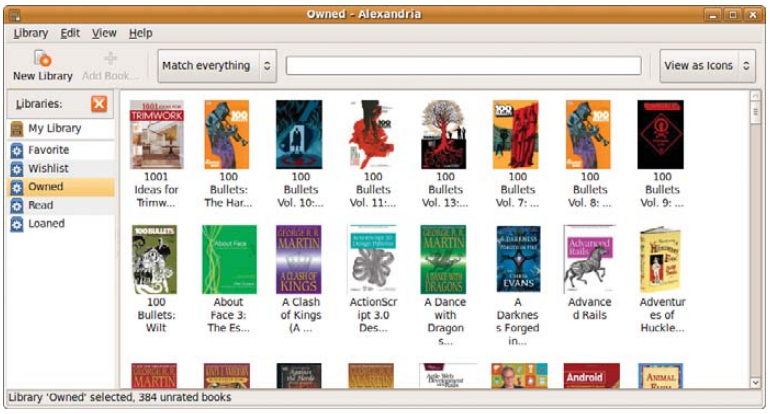


Figure 2. Alexandria Loaded with Some Books

The application itself is very simple, which was a nice change after wading through so many screens on Koha. You can search for books by title or ISBN. It lets you browse your library and search by details.

GCstar

GCstar collection management started out as GCfilms. As a result, it supports many different kinds of collections, including books, movies, music and board games, among others. It also allows you to define your own collection type, so you can track and collect anything you want.

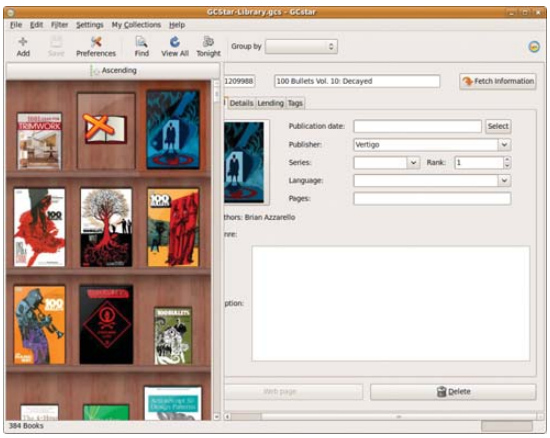


Figure 3. GCstar Loaded with Books

Installing GCstar was easy. I installed the package and then started the application. Obviously, I chose to start a collection of books. I clicked Add and started the process of looking up a book. I tried to use Amazon as an information source, but it never found anything. I assume this is related to the same policy change that affected Alexandria. I ended up using ISBNdb.com as my main source and was able to pull up information and book covers for all my test books. To make sure this wasn't fixed in a later

version, I upgraded the package to 1.5.0, and it still had the same problem.

Tellico

Tellico is a collection management application for KDE. It was available as a package, which installed with no problems. After creating a new collection of books, my first step was to add one of the test books. The process of adding a book was the most confusing out of the three applications. I clicked Create a new entry, which pulled up a dialog with a lot of options spread out over six different tabs. Title was on the first tab. ISBN was on the Publishing tab. I entered in a title for a book and clicked Save Entry. On the other applications, doing that triggered a lookup, but Tellico just sat there with no additional data. Eventually, I found an option to say Update Entry, which was able to pull down information and update it (though no book cover was provided). I tried a second time, and this time, I filled out only the ISBN field. I saved the entry and asked it to update, but nothing changed.

The version that shipped with Jaunty was 1.3.5. Version 2.1 was available as a Lucid package, so I decided to install that to see if any of these problems had been resolved. The good news is that the newer version fixed the problem with Amazon. The bad news is that it still was not able to look up the book with only the ISBN. The ISBN allows you to identify a book uniquely, which should simplify the process of confirming what book you are talking about. Searching by title provides a list of a lot of other books that are not the ones I want.

Adding Additional Lookup Sources

Now that I was able to add books to all three applications, I wanted to see how hard it was to add additional sources for lookup. GCstar ships with a number of sources. The application itself does not let you add or configure any of the sources, so your only option is to pick which one to use. The process of adding a book was straightforward. You just click Add and then put in the information. I am not sure if the problem was with authentication or something else, but the tool never found anything using Amazon as a source. I was able to pull up information about my books using ISBNdb.com. Once GCStar finds a book, the system pulls in a lot of details, including the book cover. There also is a field for storing the book's location. It is just a free-text field, so I would have to come up with my own way of encoding location. You can search by location, but there is no way to sort by it or store the search, so you can't browse the shelves based on where they are, which ends up being useful in my library, as I keep books on the same topic clumped together.

GCstar does not have any support for a scanner; however, it does have a number of different options for importing data. It even can import an Alexandria collection. One way to get the data into the system is to put the ISBN numbers into a CSV file, and GCstar then can import that CSV. Once the data is loaded, you have to go to each book to trigger the lookup in the remote repository.

Alexandria allowed me to add my Amazon credentials. It also supports adding in custom Z39.50 sources. Tellico had the most extensive list of options for adding additional sources. It included support for Z39.50 as well as GCstar plugins.

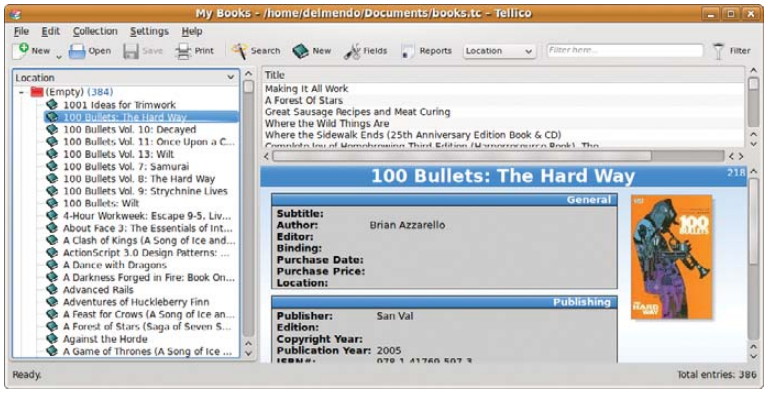


Figure 4. Default Tellico View

Where Is That Book?

One of the problems I run into with my library is that even if I remember I have a book, I don't remember where it is. Recently, I moved a large chunk of my technical books to my office, making the situation even worse, so I want to be able to track books' locations.

Alexandria does not have any concept of location baked in. It does support tags, which would allow me to enter a tag to give me a better idea about a book's location (for example, Home:Cabinet 1:Shelf 3). The search allows me to search by tags, so I could see other things on the same shelf, which would be useful because I tend to put books on the same subject next to each other.

GCstar 1.3.2 had a field for location. The newer 1.5 version has replaced that with support for tags. Once the books were tagged, I could browse the books by grouping them by tags. The search function did not support tags, so I couldn't limit my searches to books only at home or only at my office.

Tellico had the most advanced features for this part. I actually could add specific fields for library, cabinet and shelf. Then, I could use those fields for grouping and searching.

Entering Books into the System

All this searching and sorting is useful, but I saved the most important consideration for last. How do I get

.COM

FREE

FOR ONE YEAR*

NO STRINGS ATTACHED!

1&1® INSTANT DOMAIN PACKAGE:

- ✓ FREE Private Domain Registration
- ✓ 1&1 Starter WebsiteBuilder
- ✓ E-mail Account With 2 GB Mailbox
- ✓ 24/7 Toll-Free Customer Support

1&1®

MEMBER OF united internet

Get started today, call 1-877-GO-1AND1

www.1and1.com

*Offer valid as of May 1, 2010 and applies to the Instant Domain Package only. After first year, standard pricing applies. Limit 1 per customer. Visit www.1and1.com 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.

all my books into the system? The first option is simply to type in the ISBN of all my books. If your library is small enough that you are willing to do this, you probably don't need a system to track your books.

The next option is a barcode scanner. I happen to have a Flic Bluetooth barcode scanner from a previous project, and I was fortunate enough to find a great guide to getting it working under Linux (see Resources). Once everything was set up, I was able to scan the ISBNs from all the books quickly into a text file.

I tried to import the ISBNs into Tellico, but each time, it crashed on the import. I wasn't able to confirm whether this was a problem with the program or the way I installed it.

GCstar was able to import the list of ISBNs with no problem. The annoying part of that process was that once the books were imported, it did not do any lookup on the ISBN. I had to go to each book individually to tell it to download the data. Once I did, I got the book cover and everything else.

Alexandria got it right. Not only did it do the import, but it also downloaded the information about the books.

I realize that not everyone has a barcode scanner lying around. Don't worry; you have other options. If you have a Webcam, you can install ZBar. This barcode-scanning software turns your Webcam into a barcode scanner. I was able to get the same list of barcodes from ZBar that I got from my barcode scanner. The only downside is that I had to bring each book to the camera. It's a lot cheaper, but not nearly as convenient if you are scanning in a lot of books.

Resources

Koha: **koha.org**

Plano ISD Library System: **pisd.kohalibrary.com**

Alexandria: **alexandria.rubyforge.org**

Changes to Amazon API Requires AWS Account: **alexandria.rubyforge.org/news/2009-08-15--amazon-support.html**

GCstar: **www.gcstar.org**

Tellico: **tellico-project.org/about-tellico**

Linux wedge (driver) for Microvision Flic barcode scanner: **www.gsmblog.net/linux-wedge-driver-for-microvision-flic-barcode-scanner**

ZBar: **sourceforge.net/projects/zbar**

Shelfari: **www.shelfari.com**

LibraryThing: **www.librarything.com**

"Amazon Acquires Shelfari: Moves to Corner Book-Centric Social Media": **techcrunch.com/2008/08/25/amazon-aquires-shelfari-moves-to-corner-social-book-space**

Remote Options

All the solutions I looked at are downloadable, but it seems a little silly to ignore some of the options available on the Web. I looked at two different on-line options: Shelfari and LibraryThing. It was very easy to add a book on both sites. I also was able to import my list of ISBNs into both sites. On Shelfari, the import happened very quickly. On LibraryThing, it was thrown into a queue, and I was told that it would take up to an hour for the ISBNs to process.

As for sources, LibraryThing supports more than 690 different sources for information. Shelfari did not offer any information source options. Amazon acquired Shelfari in August 2008, so I assume that is where it gets all its data.

Both sites support tags, so I can use that to encode the books' locations.

Because these are both Web sites, they offer advantages and disadvantages. You easily can access the library data from multiple computers. On the other hand, you may not want everyone in the world to know you have every book on Pokémon ever published. Originally, I was concerned any data I put into either site would be locked there, but after some surfing, I found that both sites will provide you with a complete download of your library data.

Conclusion

I had a Shelfari account before I wrote this article. I often use it to create virtual bookshelves to talk about what I'm reading or to recommend a reading list. I thought about moving my collection into it, but I would prefer to work locally before I deal with putting everything on the Internet. After looking at the various options, I decided to start with Alexandria. It was the easiest to use and was best for what I need it to do. Plus, it is built using Ruby (a language I know), so I might have a shot at adding any features I need. As a test, I exported the information I already had put into Shelfari into Alexandria. Then, I was able to export the Alexandria data to both Tellico and GCstar. That means once I collect all the data, I always can switch applications later, which may be essential, as I noticed Alexandria started to slow down with only 400 books in it. Now, I just need to carve out the time to get scanning!■

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

.COM
FREE FOREVER

Get up to 5 included domains at no additional cost with a 1&1 website plan!*

1&1® HOME PACKAGE
- **2 FREE** Domains (.com, .net, .org, .info or .biz)
- FREE Private Domain Registration
- 150 GB Web Space
- 1&1 WebsiteBuilder
- 1&1 Photo Gallery
- 1&1 Blog
- 24/7 Toll-Free Support

~~$6.99~~
SPECIAL OFFER 3 MONTHS FREE*

1&1® BUSINESS PACKAGE
- **3 FREE** Domains (.com, .net, .org, .info or .biz)
- FREE Private Domain Registration
- 250 GB Web Space
- 25 FTP Accounts
- 50 MySQL® Databases
- 1&1 WebStatistics
- 24/7 Toll-Free Support

~~$9.99~~
SPECIAL OFFER 3 MONTHS FREE*

1&1® DEVELOPER PACKAGE
- **5 FREE** Domains (.com, .net, .org, .info or .biz)
- FREE Private Domain Registration
- 300 GB Web Space
- 50 FTP Accounts
- 100 MySQL® Databases
- PHP 5/PHP 6 (beta) Supported With Zend® Framework
- 24/7 Toll-Free Support

~~$19.99~~
SPECIAL OFFER 3 MONTHS FREE*

1&1®

MEMBER OF united internet

Get started today, call 1-877-GO-1AND1

www.1and1.com

*Included domains are free as long as your 1&1 web hosting package is current and in good standing. 3 months free offer valid as of May 1, 2010, a 12 month minimum contract term and a setup fee of $4.99 for the Home Package, and $9.99 for the Business Package and Developer Package apply. Visit www.1and1.com 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.

Mandriva Enterprise Server

Mandriva bills its new Enterprise Server 5 (MES 5) as "simple and innovative", combining the best of Linux (reliability, security, low admin costs) with the best of Mandriva Linux (easy to use, exclusive Mandriva tools and professional services) for the administration and integration of servers. The most noteworthy new feature in the update is increased scalability in virtualization, driven by the added kernel-based Virtual Machine (KVM) technology. Other new features include the integrated Mandriva Directory Server, reduced bootup times, increased modularization to allow easier configuration of specialized servers and reduced energy consumption, among others.

www.mandriva.com

Terracotta's Ehcache

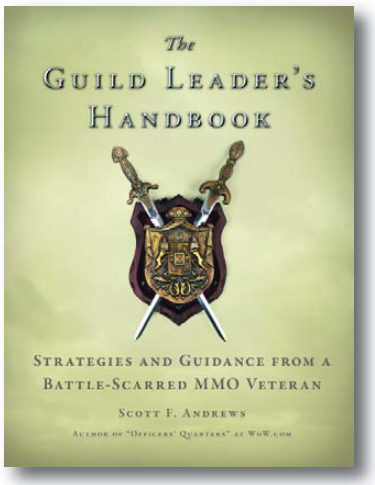
Terracotta's Ehcache, a popular open-source, enterprise-level caching solution, has added a new feature set wrapped in its latest 2.0 release. Terracotta says that Ehcache 2.0 adds enterprise-class capabilities of high-priced and proprietary technologies like Oracle Coherence without requiring application changes. Furthermore, this release brings instant scale to the majority of Java applications without code changes or costly database upgrades. Additional new features simplify the development effort, testing and scale-out, such as an express mode that easily clusters application data via configuration changes. Finally, a series of enterprise features better integrate Ehcache with the database, such as JTA for transactions and data write-behind to increase performance while avoiding database bottlenecks.

www.terracotta.org

6WIND's 6WINDGate

Embedded developer 6WIND's newly upgraded 6WINDGate, a packet processing application for networking and telecommunications systems, adds new features, such as ten-fold acceleration of packet processing functions and support for the multicore, embedded Intel Xeon processors EC5549 and E5645. The 10x speed-up compared to a standard Linux implementation allows customers to reuse their existing application software and accelerate their time to market. 6WIND further adds that the 6WINDGate SDS profile is optimized for platforms in which the networking Fast Path runs on dedicated cores without the overhead of a Linux-based Slow Path. 6WINDGate's architecture removes the complexity of integrating high-performance packet processing with the Linux environment, because it fully synchronizes the Fast Path and Linux, while preserving Linux APIs. It includes complete Layer 2 through Layer 4 embedded networking features (routing, IPsec, firewall, QoS, NAT, multicast and so on), reducing development time by as much as 70%.

www.6wind.com



Scott F. Andrews' *The Guild Leader's Handbook* (No Starch Press)

"Who said dragon slaying was easy?" is author Scott F. Andrews' Bilbo Baggins-esque invitation to try his new book *The Guild Leader's Handbook*, a guide to leading a guild in massively multiplayer on-line (MMO) games. Subtitled "Strategies and Guidance from a Battle-Scarred MMO Veteran", the book outlines not just how to create, build and maintain a successful guild but also how to lead it to glory. Andrews should know, because he leads one of the oldest-surviving guilds in *World of Warcraft*. He shares secrets from his long experience to illustrate how to plan successful raids, player vs. player battles, role-playing sessions and contests; deal with problem players and keep a lid on guild-fracturing drama; solve loot issues and choose the best loot system; boost morale, reputation and server presence; and promote and motivate an effective officer corps. *The Guild Leader's Handbook* is published by No Starch Press.

www.nostarch.com

Imsys' SNAP Classic

The Swedish firm Imsys proudly announced a new generation of its SNAP Classic module, a drop-in replacement for the DSTINI390 reference design—a Java-based networked controller. Using the same module size and interfaces, the SNAP Classic provides all interfaces supported on the TINI platform. The new SNAP Classic offers greatly improved performance while reducing cost and power consumption, retaining all the legacy and adding new compelling features. For heavy-duty, floating-point computations, SNAP Classic now offers a performance increase of more than 200 times that of TINI, says Imsys. A "crypto engine" consisting of 27 new opcodes supports ARC4, DES, AES, RSA, MD5, SHA1 and SHA256, accelerating them by a factor of up to 43 times compared to their C code implementations on the same processor. For development, the SNAP Classic user can use either free Java tools or purchase the optional Imsys Developer, a professional IDE that enables the developer to use a mix of Java, C and assembler programming.

www.imsystech.com

Mobile Edge ScanFast Laptop Cases

From the "Not Just for Linux Geeks Desk" comes Mobile Edge's new ScanFast line of laptop carrying cases and accessories, which the producer claims to be "the first TSA-Compliant Netbook case collection on the market". ScanFast is targeted at Netbook owners seeking a case more substantial than a simple sleeve and encompasses the Edge Netbook Briefcase, Messenger Bag and Backpack products. Each product has the additional advantage of being checkpoint-friendly at airports. To be checkpoint-friendly, TSA requires laptop compartments to be independent and clear of any other gadgets, cords, metal zippers and so on, and screeners must have a clear unobstructed view of the laptop itself. The cases support notebooks up to 13.3" (34cm) wide.

www.mobileedge.com/scanfast

DVEO's eYeCatcher ATSC-M/H

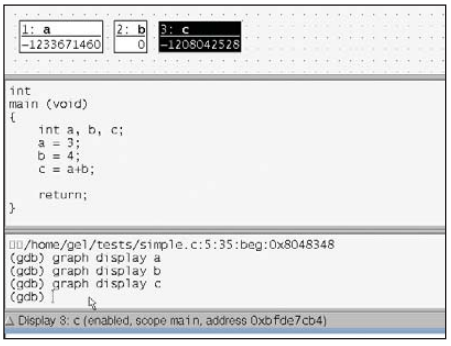
DVEO is now shipping the new eYeCatcher ATSC-M/H, a compact test modulator for emulating Mobile DTV (digital TV) signals. Designed for use in development labs and for technology demonstration purposes, the eYeCatcher ATSC M/H is a portable frequency agile modulator with IP, ASI or SMPTE 310M input and ATSC M/H output. It delivers real-time or stored video to cell phones, PDAs, handhelds and vehicles. The device is ideal for laboratory applications, testing set-top boxes and mobile devices, and in-store demonstrations of ATSC M/H devices.

www.dveo.com

Undo Software UndoDB

The Cambridge, UK-based Undo Software bills the new version 3.0 of its reversible debugger for Linux, UndoDB, as "a huge step backwards". UndoDB's reversible debugging capabilities (also known as replay or historical debugging) allows a developer to step or run an application backward and answer the real question when debugging: "How did that happen?" The Undo folks say that UndoDB 3.0 can debug nearly any Linux process, including those using multiple threads, asynchronous signal handlers and shared memory. The new edition also is reputed to be faster than ever, running applications with a slow-down of just 1.7x while still keeping full visibility of the program's entire execution history. Finally, UndoDB 3.0 supports reverse watchpoints, allowing programmers to find the root cause of elusive memory-corruption bugs easily.

undo-software.com



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

Fresh from the Labs

MilkyTracker—Open Music Tracker

www.milkytracker.org

Fans of musical tracking programs, such as Fasttracker, ProTracker, CheeseTracker and the like, will want to check out MilkyTracker, which has been quite popular on SourceForge and has had pretty widespread distro integration of late. To quote MilkyTracker's documentation:

> MilkyTracker is an open-source, multiplatform music application, more specifically, part of the tracker family. It attempts to re-create the module replay and user experience of the popular DOS application Fasttracker II, with special playback modes available for improved Amiga ProTracker 2.x/3.x compatibility.

Installation When it comes to installation methods, you are pretty spoiled for choice. Along with the usual source tarball, packages in various repositories are available for Ubuntu, Arch Linux, Debian, Enlisy, Gentoo and SUSE, as well packages for our FreeBSD and OpenBSD cousins. Ubuntu users are doubly spoiled with a binary tarball built upon Karmic.

For those who are running with source, according to the documentation, MilkyTracker can be compiled using the standard `./configure; make; make install` (note that `make install` requires the use of root or sudo). However, I ran into compilation problems during the "make". I hope you have more luck. I went with the binary tarball in the end, which ran with no problems.

As far as libraries go, there shouldn't be *too* much in the way of strange requirements, although I did need to install libzzip-dev and libsdl1.2-dev to get past the source code's configure script.

Once your installation has finished, run MilkyTracker with:

```
$ milkytracker
```

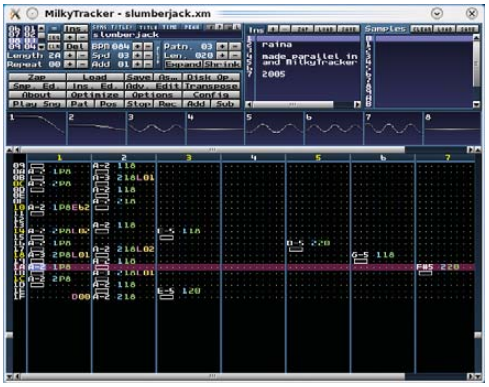
Usage The first thing I recommend doing is loading some of the provided songs, which instantly will show off MilkyTracker's capabilities. Click the Load button in the cluster of gray buttons near the top left of the screen, navigate to the directory in which MilkyTracker has been installed, and look at the songs directory. Choose one of the available tracks and click Play Song on the bottom-left corner of the main cluster of gray buttons. My personal favorite (or at least the most credible of these tracks—demonstration songs are always pretty dry) is "slumberjack", which is nice and progressive and shows off MilkyTracker's capabilities quite nicely.

As the track plays, you'll see a bar move rapidly down the main composition screen's page and move on to other pages of music as the song progresses into new movements. A welcome feature from classic tracker programs is the wave visualization inside those windows in the middle section. They give individual readouts for each channel. It's pretty cool to watch this multitasking in progress and see the music's very DNA scroll before your eyes.

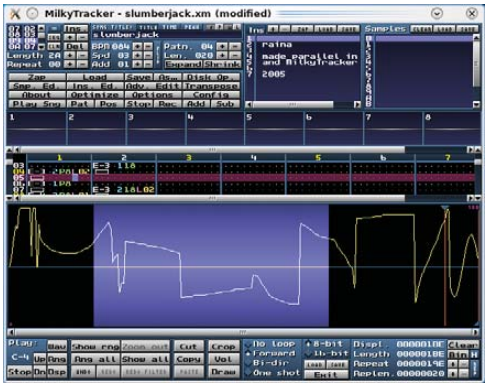
I also noticed a very willing use of the stereo spectrum in this program, which helped to add spice. That said, my favorite part of this project is the sample editor, which lets you manipulate waveforms by hand. It also lets you literally draw your own waveforms—effectively making something from nothing.

However, none of this stuff will come as a surprise to tracker veterans, who've grown up with such hard-core features since the days of DOS. Newbies who are used to soft-core programs like FruityLoops will freak out in this imposing retro environment. Veterans probably will rejoice in the imposing low-level interface and go back to skulking around in their basements listening to Kraftwerk and Wumpscut.

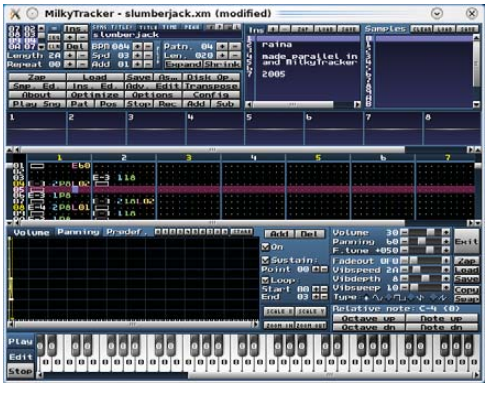
Ultimately, MilkyTracker provides an authentic environment for those who have grown up with these programs, while adding more modern capabilities and platform diversity. I personally find these programs way too daunting, but old-school Tracker fans are going to love it.



MilkyTracker provides all the old-school, low-level control from ye olde days of music tracking.



You don't get much more hard core than waveforms drawn by hand!



Thankfully, a keyboard-based instrument editor is included as well.

Paintown—2-D Arcade Fighting Engine

paintown.sourceforge.net and freshmeat.net/projects/paintown

Before I begin, there's been some recent controversy over this project, with the accusation that this project is ripping off someone else's work. Playdeb.net was sent the following message from the Senile Team:

> It may interest you to know that *Paintown* "borrows" original work from Senile Team without permission. To put it more bluntly, *Paintown* is a rip-off from *Beats of Rage* (see **www.senileteam.com/beatsofrage.html**).
>
> The source code and assets for *Beats of Rage* are freely available, and may be used by anyone—provided of course that they give proper credit. The author of *Paintown*, however, has openly refused to do so, and *Paintown* should, therefore, be considered in violation of copyrights.
>
> The author of *Paintown* has on several occasions been confronted with the impossible similarities between "his" game and *Beats of Rage*. However, rather than admitting to the obvious, he instead decided to alter some of "his" code and assets in order to hide their true origin. And yet even now, the screenshot seen on your site immediately betrays *Paintown* as a rip-off, containing several custom graphics that were made by Senile Team.

I had already written this month's piece when I received this information, and Playdeb.net took down the package (although things may change by the time this article is printed). However, I feel it's best to pass on this information and let you decide for yourself.

This is not my genre of gaming and I'm far from an expert, but this project instantly caught my attention and seems to have a great deal of potential. To quote the Web site:

> *Paintown* is a 2-D engine for fighting games. If you are looking for a side-scrolling, action-packed game like you used to play, or if you are looking for an extensible engine to write your own game, look no further. *Paintown* supports user-created content through a mod system and user-defined functionality through scripting.
>
> *Paintown* also supports an implementation of M.U.G.E.N. Our goal is to be 100% compatible with M.U.G.E.N 2002.04.14 beta as well as supporting any new updates in the 1.0 version.
>
> *Paintown* is completely open source, and we would love any contributions in the form of code, art or donations. Give *Paintown* a try!

Paintown has the following features, according to the Web site: low CPU and GPU requirements, network play, dynamic lighting, joystick support, mod/s3m/xm/it music modules, scripting with Python and the M.U.G.E.N engine.

Installation and Usage At the Web site, a source tarball was (supposedly) available, but I ran into some trouble with it. The Web site link wasn't working at the time of this writing, and it gave an error. However, the link from the project's Freshmeat page was working fine. I'll leave the rest of the installation details to you on this one, as things seem to be a bit up in the air with this particular project.

Inside the game, things are fairly intuitive, but documentation for some of the finer details is lacking, so please forgive me if I make some dumb errors. There are three main components: Adventure Mode, Adventure Mode with Computer and M.U.G.E.N mode. Adventure Mode puts you in a sideways-scrolling street-brawl game, with changing scenery and characters in the style of *Final Fight*, *Double Dragon* and

Paintown can be a bit intense, with lots of giblets, spinal cord and whatnot. Don't tell Mum though.

How many commercial games would let you team up Donatello with Wolverine, eh?

so on. Adventure Mode with Computer adds a computer-controlled Player 2, and you can assign a different character to each player.

The beauty of a noncommercial game such as this is that fans generally add their favorite characters from other games, and here you can choose from such characters as Ryu and Blanka from *Street Fighter*, Goku from *Dragon Ball* and even the time-honored Wolverine!

Each character has different strengths and weaknesses, and Attack buttons vary between them all. Nevertheless, there are similarities between most characters: Attack 1 usually is punch, Attack 2 is a kick, and Attack 3 generally grabs an opponent. A character may have only one Attack button in use, but that Attack will be particularly devastating. Or, all three Attack buttons will be used, but with less power in each, although a more even spread.

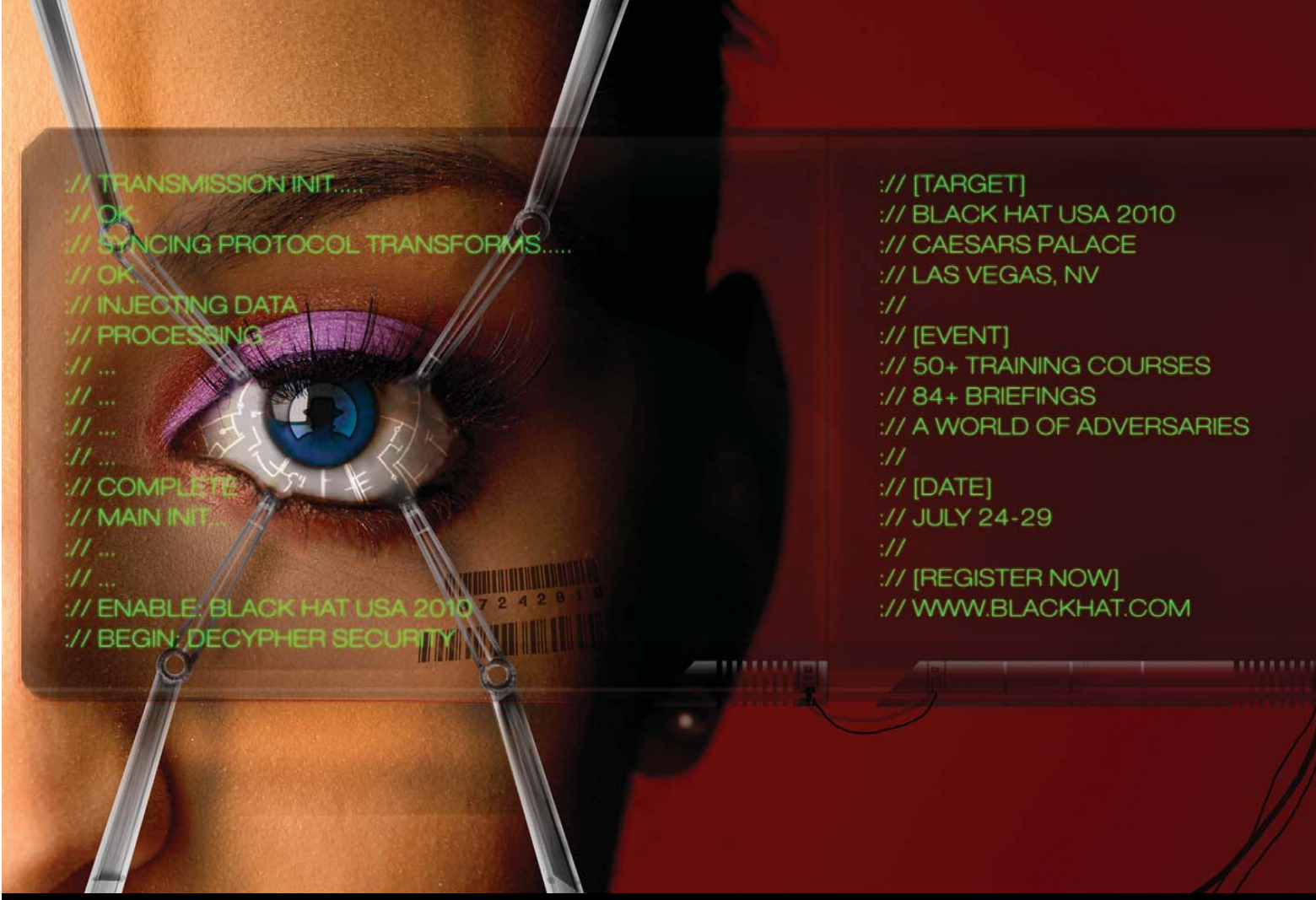
Now let's look at M.U.G.E.N mode. For those not in the know, *M.U.G.E.N* was a 2-D fighting game built around customization, creating characters, background stages and so on. It spawned a community all its own with versions for DOS, Windows and, thankfully, Linux. These communities still are running strongly today with extraordinarily dedicated projects, such as the Infinity M.U.G.E.N Team's highly ambitious *Marvel vs. Capcom*, which is an entire gig's download!

As for *Paintown*'s M.U.G.E.N game, it's very basic and rudimentary, with only one character from which to choose and fight against, with some joking cutscenes before and after playtime. Promisingly enough, it does have a training mode, as seen in the later years of this genre, so I look forward to seeing how things progress, given the attention to detail. Unfortunately, I'm out of space, so I can't really give it the coverage it deserves. I also didn't get a chance to look at the multiplayer networking side of things, but I'm sure it will make office lunch hours a good laugh!

Although this is not really my genre, and I'm not comfortable with advocating violent video games, I'd be remiss in my duties to not report on it due to my own biases. The open framework of this project is marvelous, and its integration of M.U.G.E.N is all the better, which should breathe life into a genre that's mostly been abandoned by mainstream commercial gaming. Plus, the noncommercial aspect allows fans to live out their gaming character fantasies that commercial licensing would simply not allow.■

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.



black hat usa 2010

DIGITAL SELF DEFENSE

LAS VEGAS, NEVADA
CAESARS PALACE HOTEL

Understanding the increasingly complex threats posed to an enterprise is a daunting task for today's security professional. The knowledge to secure an enterprise against those threats is invaluable. Come to Black Hat and learn from the industry's best. Register for Black Hat USA today at:
www.blackhat.com

july 24 - 29

DIAMOND SPONSOR

PLATINUM SPONSORS

GOLD SPONSORS

SILVER SPONSORS

HARDWARE

Pogoplug

Pogoplug, the easiest file server you'll never set up.

MIKE DIEHL

I'm sure I'm not the only one who is apprehensive about uploading family pictures and home movies to on-line services like Facebook or YouTube. As we all know, once something is on the Internet, it's out of our control, and who knows where it will turn up next. On the other hand, the Internet is a great way to share media with friends and family. The more technically savvy among us certainly can figure out how to host media on servers that we control, but most people need a box they can plug in that "just works". And, that's what Cloud Engine's Pogoplug does.

The Pogoplug is a box the size of a Wi-Fi router that has a Gigabit Ethernet port and four USB ports (one on the front and three on the back). Inside is an ARM processor running Linux kernel version 2.6.22—all this, and it's pink! Yes, I said pink. The case is white with a clear acrylic shell, trimmed in pink, with a cable management clip in the back of the device. Overall, it's rather Apple-esque (Figure 1).

Figure 1. Pogoplug (from Cloud Engine's Web site)

My review unit came with a bag of microwave popcorn and a simple note saying to go start the popcorn, then come back and follow the Pogoplug setup instructions. The note boasted that the Pogoplug would be ready before the popcorn, and it was. Once I connected power, networking and a USB hard drive, which I happened to have on hand, all I had to do was go to **my.pogoplug.com** and enter the device's ID, my e-mail address and a password. A couple seconds later, it was ready to go. From that point on, all I had to do was supply my e-mail address and password to access the device. This is the kind of simplicity most people expect from consumer electronics.

Once the device finished booting, it began scanning the attached storage for pictures, movies and music files. It wasn't long before I was uploading media to the device and making it available to various friends and family. At this point, it was all pretty intuitive, though by no means sophisticated. I was able to grant either read-write or read-only access on a directory basis—real simple. When I granted access to people, they received an e-mail message telling them how to access the shared media. When my friends clicked on the hyperlink, they were able to preview or download various files I had made available to them. From their points of view, it was just another Web page; from my point of view, it was a server that I controlled.

All interaction with the Pogoplug is through a single Web site, **my.pogoplug.com**. This Web site is able to access your device, or devices, and present you with an easy-to-use Web interface. You don't even need to know your device's IP address (Figure 2). In the center of the page, you see thumbnail previews of some movies and a music file that I uploaded. When you mouse over a thumbnail, a pop-up menu appears that lets you either preview (Figure 3), download or remove the file. Along the left side of the page, under My Media, are various media categories. These links simply show a filtered view of all of the files on the device. Under Show my files, there are a few other ways of accessing media files on the device. Maybe it's the nerd in me, but I prefer to access the drives directly, from the My library section. Here, I have two thumbdrives plugged in (middle left-hand side of Figure 2).

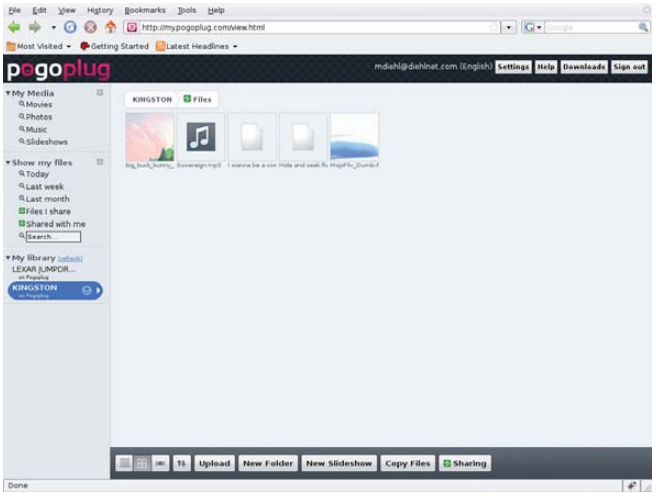


Figure 2. Web Interface to Pogoplug via my.pogoplug.com

Across the bottom of the page are buttons that allow you to change how the device displays your file lists, as well as buttons that let you upload new files, create folders and copy files from one folder to another. There's also a Sharing button. This is the button you press when you want to grant other people access to a directory.

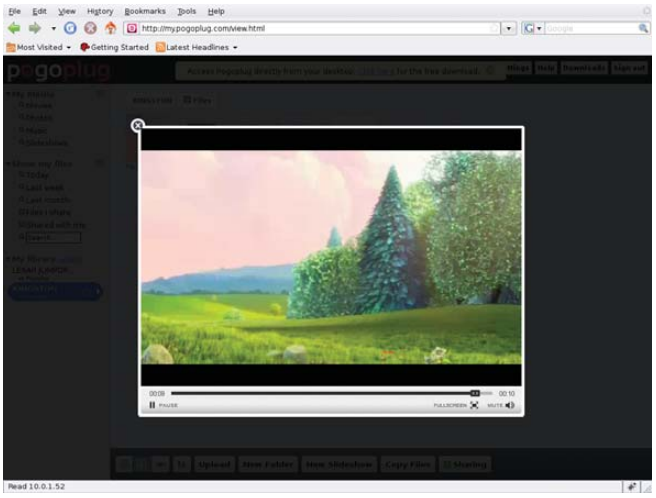


Figure 3. Previewing an Image

When you click on the Sharing button, you are presented with a screen that allows you to turn sharing on or off for the current folder. If sharing is on, you can invite people to access the share. Also, you can see who has access, and whether it's full access or read-only access. Additionally, this is the screen where you configure the social-networking features of the Pogoplug.

Most of my friends and family use Facebook to keep in touch, so I was intrigued to hear that the Pogoplug would integrate with Twitter, Facebook and MySpace. I decided to check out the Facebook feature. Once I supplied my Facebook user name and password, the device posted a news article to my "wall" any time I added files to the shared directory. Obviously, this would be a great way for the grandparents to keep up with all the grandkids.

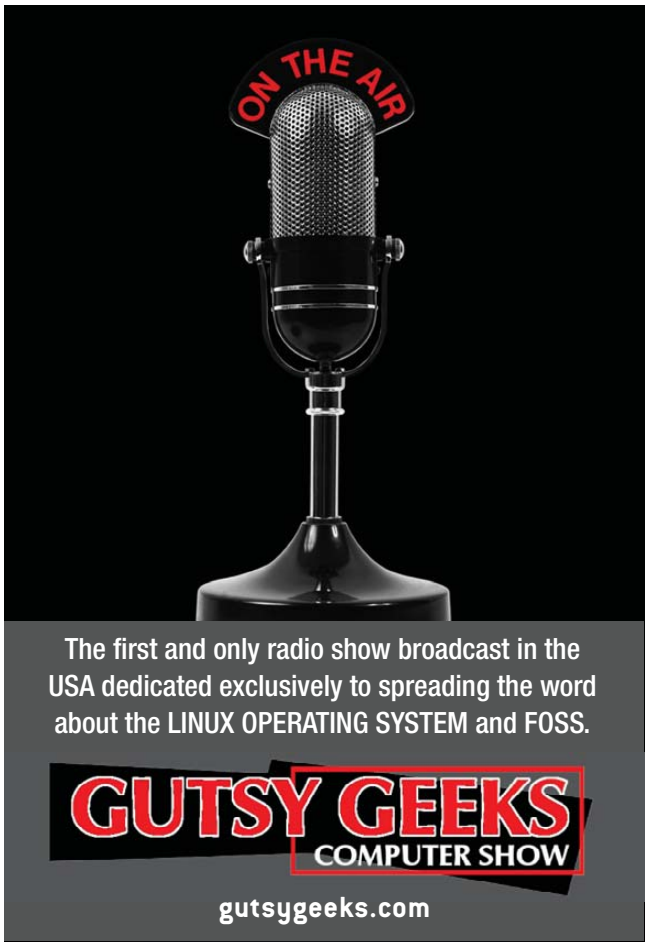
Okay, so how does it work, really? As soon as the device boots up, it calls home via what looks like a TCP-based VPN. From that point, the device is able to respond to commands, such as "Tell me what services you are providing" or "Delete a file." It's important to note that this is a free service, and that your files stay on your device and don't simply get uploaded to Cloud Engine's servers. But, one of the nice things about this architecture is that the Cloud Engine Web site is a central point of interaction with your Pogoplug device, or devices. This also enables you, the consumer, to write Web service requests from any Web page that can interact with your Pogoplug (I'll come back to this in a minute).

The dark side of this architecture is that cloud computing could become a potential single source of failure. If Cloud Engine went out of business, this infrastructure would go away also. However, to mitigate this threat and to dissuade fears, Cloud Engine has put its source tree in escrow to be released to the public if need be. In the meantime, Cloud Engine uses this infrastructure to distinguish its product from other similar devices. In fact, chatter on the Pogoplug user forums indicates that the infrastructure is being developed and improved actively. What I've gleaned from reading the forums is that Cloud Engine can push upgrades to the device—meaning that the Pogoplug is basically a managed device, but the owner maintains complete control.

Sadly, the device doesn't ship with a native Samba or NFS server. The device is targeted at people for whom this much configuration would be too much. But, there is a filesystem driver available for download that runs on Linux, Windows and Mac. I installed it on my Linux workstation and my wife's Windows laptop, and it worked quite well.

So although the device is designed simply to work right out of the box with little or no configuration, it still is quite hackable by more sophisticated users. However, you first have to determine what the device's IP address is, and this isn't to be found anywhere on the device's Web site! I had to ask my DHCP server. Armed with the IP address, I was able to `ssh` into the machine as root using the default password, ceadmin. This is a reasonably secure configuration, because presumably, the device will be behind your NAT router and not accessible from the outside world. However, I'm told that after March 16, 2010, SSH will be turned off by default, but users will be able to enable it through the Web interface.

Earlier, I alluded to the ability to send Web service requests to a Pogoplug. Cloud Engine's Web site documents how this is done and provides a fairly complete API for interacting with a Pogoplug. With a little bit of reading, I was able to perform some basic functions in only a few minutes.



First, you have to authenticate via the **service.pogoplug.com** Web site. This is done by sending an HTTP GET request with this structure:

```
http://service.pogoplug.com/svc/api/
➥loginUser?email=user@example.com&password=supersecret
```

You can see that the only parameters are your e-mail address and the password you use to access the device. The server will reply with a JavaScript object that contains various pieces of information, including a valtoken field, which contains an authentication token. You simply include that token in subsequent requests, such as:

```
http://service.pogoplug.com/svc/api/
➥listDevices?valtoken=5LfDTCKNZRoi86Ifdd
```

This request results in another JavaScript object detailing the hardware that's plugged in to my device:

```
{"devices":
[{"deviceid"   :"UZHTREWJF52ZMN",
  "type"       :"xce:unknown",
  "name"       :"Pogoplug",
  "version"    :"LINUX POGOPLUG - 2.0.3",
  "flags"      :"0",
  "ownerid"    :"228d65951f3fe1ff6c",
  "owner"      :
      {"userid"       :"228d6fe1ff6c",
       "screenname"   :"user@example.com",
       "email"        :"user@example.com",
       "flags"        :"sawoobe,gv=m,",
       "emails"       :[{"address"     :"user@example.com",
                         "validated"   :"1",
                         "default"     :"0"}],
        "options"      :[{"name"    :"disablehoverprev",
                          "value"  :"1"},
                          {"name"  :
                           "slideshare_4dDxYuUonfJQ"}]},
        "sku"          :{"skuid"   :"3",
                         "oem"    :"Cloud Engines",
                         "name"   :"POGO-TEST",
                         "user"   :"Pogoplug",
                         "terms"  :"0"},
        "terms"        :"0",
        "valstart"     :"0",
        "valend"       :"0",
        "services"     :[{"deviceid"     :"UZHTRETF52ZMN",
                          "serviceid"    :"Yxg9f32P-mcuVzIIZRKRAQ",
                          "sclass"       :"1",
                          "type"         :"xce:plugfs:vfat",
                          "name"         :"KINGSTON",
                          "version"      :"LINUX POGOPLUG - 2.0.3",
                          "online"       :"1",
                          "msgpending"   :"0",
                          "apiurl"       :"http://10.0.1.52:3333/...",
                          "space"        :"1358462976/1999044608",
                          "flags"        :"12288"}]}]}
```

You can see that it would be pretty usable inside a JavaScript program. I haven't looked at the API documentation with an eye toward writing a particular application, but it seems pretty complete and well documented.

As is, the Pogoplug is a simple file server, but thanks to OpenPogo (which is now known as PlugApps), you actually can extend the device's functionality. By default, the root filesystem is mounted read-only, but once the filesystem is remounted read-write, you can mount an attached USB hard drive manually. Then, you download the PlugApps tarball and extract it onto the mounted filesystem.

Once PlugApps is installed, you can install additional packages using the ubiquitous ipkg package manager. From here, it's pretty easy to imagine a Samba, Firefly or CUPS server running on the device. But, the neat thing about the way the installation is done is that it doesn't overwrite the existing filesystem. This means you don't lose any of the benefits of the Cloud Engine infrastructure.

Overall, I think the Pogoplug is a neat little device, but it does have some negatives. I was really hoping to be able to plug in my digital camera and have the Pogoplug recognize it as a regular USB hard drive and publish my pictures. This would have been a great way to offload photos and make them available to friends and family in one operation. Alas, that didn't work, but it probably has more to do with my camera than the Pogoplug. When I pulled the memory card out of the camera and put it in a USB card reader, it worked like a charm.

By itself, the device is very attractive-looking. However, all of the equipment in my LAN closet is black, as is most of the equipment on my desk in my office. Even my entertainment system is black. Against these black backgrounds, this pink Pogoplug kind of sticks out like a sore thumb. This leads me to wonder how hard it would be to offer different colored skins for the device. I'd snap up a black skin in a heartbeat, although I'm sure other designs, such as camouflage, might be popular. On the other hand, I have to admit, it is quite the conversation piece.

The configuration was designed to be simple, and I'd venture to say that it actually might be too simple. I'd like to be able to associate comments with the various files I upload. Also, the categories Movies, Photos, Music and Slideshows are a bit limiting, as I also have documents and spreadsheets that I share. Finally, the ability to organize my music collection better would be very much welcome.

That said, the Pogoplug is an extremely easy device to set up and use. This $130 device eliminates virtually all the hassle of sharing media, controlling access and telling my friends and family when I've uploaded new baby pictures, for example. My plan for the device is to tuck it away in the corner of my LAN closet and post some shares for family pictures. I'll also post a share for just our immediate family, where we'll store genealogy data that my wife can collaborate on with her father on the other side of the country. I'll probably also set up a share for my favorite YouTube videos. With the Pogoplug in place, my wife and kids may not have to interact directly with my main server anymore, but they'll still have convenient access to the stuff they want.■

Mike Diehl is a contract programmer and consultant in Albuquerque, New Mexico. Mike lives with his wife and three small boys and can be reached via e-mail at mdiehl@diehlnet.com.

2010

USENIX Federated Conferences Week

June 22–25, 2010 Boston, MA

www.usenix.org/confweek10

New for 2010, USENIX is combining established conferences and new workshops into one week full of research, trends, and community interaction.

USENIX ATC '10
2010 USENIX Annual Technical Conference
Wednesday–Friday, June 23–25 www.usenix.org/atc10

WebApps '10
USENIX Conference on Web Application Development
Wednesday–Thursday, June 23–24 www.usenix.org/webapps10

WOSN 2010
3rd Workshop on Online Social Networks
Tuesday, June 22 www.usenix.org/wosn10

HotCloud '10
2nd USENIX Workshop on Hot Topics in Cloud Computing
Tuesday, June 22 www.usenix.org/hotcloud10

HotStorage '10
2nd Workshop on Hot Topics in Storage and File Systems
Tuesday, June 22 www.usenix.org/hotstorage10

Google's Go Tutorial
Wednesday, June 23

FAST-OS Workshop
Wednesday, June 23

Cloud Virtualization Summit
Thursday, June 24

Configuration Management Summit
Thursday, June 24

Cfengine 3 Tutorial
Friday, June 25

EARLY BIRD DISCOUNT
REGISTER by JUNE 7 & SAVE

And more!

Why Attend
You'll leave this USENIX-style mashup with insights into the future of many hot topics. Your access to leading researchers and industry experts will not only provide answers to your most burning questions, but will also help you build lasting connections across multiple disciplines. Take full advantage of this opportunity and learn from the top researchers, practitioners, and authors all in one place, at one time.

USENIX
www.usenix.org

Stay Connected...

http://www.usenix.org/facebook
http://www.usenix.org/linkedin
http://twitter.com/usenix
http://blogs.usenix.org

DISTRIBUTIONS
A BRIEF HISTORY

Add one part GNU, one part Linux kernel, stir lightly, bake for 19 years, and you get 452 different meals.

JES FRASER

It seems as though there are as many Linux distributions as there are letters in the alphabet with which to name them. Certainly, there is a flavor to satisfy almost any palate. It wasn't always this way, however. How did it happen? Why hasn't the Linux world just standardized on a single distribution?

The beginning of the Linux distribution really started with Richard Stallman and his fledgling Free Software Foundation in the early 1980s. The GNU operating system was being developed, intending to re-implement a UNIX-like operating system as free software. Although many GNU tools enjoyed wide use, the project suffered various setbacks and delays in its hunt for a kernel. There was a lack of cooperation from some at Berkeley with using the BSD kernel, and there were licensing issues with Mach (Carnegie-Mellon University's microkernel). Before these issues were resolved and the GNU Project was able to make headway building its own kernel, Hurd (another free kernel) became available for use. Then in 1991, the following message appeared on a Usenet newsgroup **groups.google.com/group/comp.os.minix/msg/b813d52cbc5a044b?pli=1**:

> Hello everybody out there using minix-
>
> I'm doing a (free) operating system (just a hobby, won't be big and professional like gnu) for 386(486) AT clones. This has been brewing since april, and is starting to get ready. I'd like any feedback on things people like/dislike in minix, as my OS resembles it somewhat (same physical layout of the file-system (due to practical reasons) among other things).
>
> I've currently ported bash(1.08) and gcc(1.40), and things seem to work. This implies that I'll get something practical within a few months, and I'd like to know what features most people would want. Any suggestions are welcome, but I won't promise I'll implement them :-)
>
> Linus
>
> (PS. Yes - it's free of any minix code, and it has a multi-threaded fs. It is NOT protable [sic] (uses 386 task switching etc), and it probably never will support anything other than AT-harddisks, as that's all I have :-(

Linux provided a stopgap measure for the free kernel GNU needed to be a complete operating system. Because Linux was no more and no less than a kernel, it similarly needed tools like those provided by GNU and other projects to be usable by anyone other than a developer. Early enthusiasts put together bundles of software running on the Linux kernel, creating the first distributions. These early distributions were mostly created by universities for internal use, some releasing their handiwork to the wider community.

With Linux 0.12, Linus announced his intention to license the kernel under the GNU GPL (General Public License). Achieved by the release of Linux 0.99, this removed the prior restriction on commercial distribution, paving the way for commercially backed distributions that were soon to come along.

One of those was Soft Landing Systems' SLS Linux, one of the first widely used distributions. It was installable by floppy disk and included such cutting-edge features as TCP-IP networking support and the X Window System. SLS enjoyed popularity for the time, but it came under criticism for being buggy and unstable. When Soft Landing Systems announced it was going to change the default binary format from a.out to ELF, it met with a very negative response from the user base.

Among those upset by Soft Landing Systems' decision was Patrick Volkerding, who then created a modified version of SLS he named Slackware. The first release was July 16, 1993, and Slackware holds the honor of being the oldest currently maintained Linux distribution—by a few months.

Ian Murdock also became quite frustrated with the flaws he perceived in SLS, and he announced on the newsgroup comp.os.linux.development on August 17, 1993 (**groups.google.com/group/comp.os.linux.development/msg/a32d4e2ef3bcdcc6**):

> This is just to announce the imminent completion of a brand-new Linux release, which I'm calling the Debian Linux Release. This is a release that I have put together basically from scratch; in other words, I didn't simply make some changes to SLS and call it a new release. I was inspired to put together this release after running SLS and generally being dissatisfied with much of it, and after much altering of SLS I decided that it would be easier to start from scratch.

In the same year, Ian Murdock released the Debian Manifesto, detailing his vision for a free and open distribution that would be developed and maintained communally. He saw such a distribution as the way to avoid stagnation, crucial to Linux's success in the commercial market. The Free Software Foundation supported these efforts by funding Debian development for a year.

Both Debian and Slackware were born of the dissatisfaction of not just the problems with SLS but the closed nature of the distribution preventing any improvements being submitted by the user base. Ian Murdock's solution of having a distribution where many hands made light work of high standards persists today, although Debian's development process has come under criticism for its long release cycles.

Although Murdock built his solution on community involvement, Volkerding's response was almost the polar opposite. The Slackware team did grow to multiple developers; however, as late as 2000, all changes still were being signed off personally by Volkerding as the project's "czar". This, he was recorded to have said, ensured that there was "...a high level of quality and consistency" (from **http://slashdot.org/interviews/00/03/17/1120205.shtml**).

Red Hat Linux was another of the successful early distributions. Marc Ewing was developing software for the UNIX platform but turned to Linux as a more affordable option than a UNIX workstation. He found after a time that rather than work on his main project, he spent the majority of his time fixing issues with Linux. Marc eventually decided to put his original project aside and "...work on putting together a better Linux distribution" (**http://www.salon.com/tech/view/1999/10/04/marc_ewing**) as a business venture.

His first version took about a year and a half, and released in 1994. Bob Young purchased most of the produced copies of the distribution for ACC Corporation, a mail-order business that sold Linux- and UNIX-related products, including distribution media. Not long after that, in 1995, Young bought Ewing's business and merged it with ACC to become Red Hat Software. That year, Red Hat 2.0 was released, including the then-new RPM package management system. Four years later, Red Hat bought and merged with Cygnus, becoming the largest open-source company in the world at that time.

SuSE, standing for Software und System-Entwicklung in German (Software and Systems Development), originally started as a software development and UNIX consultancy company, consisting of its four founders Roland Dyroff, Thomas Fehr, Burchard Steinbild and Hubert Mantel. Their original business plan did not

succeed, so SuSE moved on to distributing Linux. To start with, SuSE provided software packages and services based on SLS. Once Patrick Volkerding had created Slackware, SuSE began translating it into German.

For some time, SuSE continued to translate and distribute Slackware, but for much the same reasons that Patrick Volkerding and Ian Murdock became frustrated with SLS, SuSE grew dissatisfied with Slackware. Slackware's closed development prevented SuSE from having its improvements and changes accepted. Disappointed by bugs it was unable to fix, SuSE decided to create its own Linux distribution.

The SuSE team identified a need for an easy-to-use installation and configuration tool and started work on YaST, the core of modern SuSE Linux. Over time, SuSE incorporated many features of Red Hat Linux, including the RPM packaging system and the Red Hat-style rc system.

Of course, both Red Hat and SuSE exist in somewhat different forms today. Red Hat eventually stopped supporting the free version of its operating system to focus on Red Hat Enterprise Linux, with the Fedora Project stepping in to maintain the community-focused distribution. SuSE Linux went through a similar change not long after being acquired by Novell, with OpenSUSE rising as a new free and open flavor. Slackware and Debian have navigated the years more or less intact, with Ian Murdock's vision of Linux's commercial success realized not in Debian itself but in the one who was yet to come—Ubuntu.

These are just the earliest distributions whose impact still clearly can be seen in the Linux world, with all of them featured on DistroWatch's top ten distributions list (albeit with the metamorphosized forms Fedora and OpenSUSE). Many others were not as successful but also played a part in shaping the early face of Linux—bonus points to any readers who have been Linux users long enough to remember MCC or Yggdrasil.

All four of the successful distributions had the common trait of providing a Linux distribution that was improved and easier to use, and all four of the distros discussed here were frustrated with their efforts at trying to contribute to the projects that had come before them. Ultimately, the story behind our myriad Linux distributions is like that of Linux itself: each was started by the most pragmatic kind of dreamer who could visualize a better way and had the talent and drive to make that vision real.■

Jes Fraser is a Linux specialist from New Zealand.

Resources

Release Note for Linux v0.12: **www.kernel.org/pub/linux/kernel/Historic/old-versions/RELNOTES-0.12**

The Debian Manifesto: **www.debian.org/doc/manuals/project-history/ap-manifesto.en.html**

Red Hat History: **www.redhat.com/about/companyprofile/history**

DistroWatch: **distrowatch.com/dwres.php?resource=major**

Linux Timeline: **www.linux.org/info/linux_timeline.html**

ASA COMPUTERS

Want your business to be productive?

The ASA servers powered by the Intel® Xeon® Processor provide the quality and dependability to keep up with your growing business.

" Since 1989 Integration and service with pride "

ASA 1U Series

Intel® Atom™ processors D510 and D410 based platform and are designed for embedded industrial PC (IPC) applications. These quiet, energy saving solutions make ideal network appliances, print and email servers.

ASA 2U Series

QPI, Intel® Xeon® Processor 5500 Series in a high-density 2U form-factor are ideal for network infrastructure, front-end enterprise, and minimal downtime cluster server systems maximum upto 4 nodes in 2U.

ASA 3U Series

Server series excel under ISCSI/NAS/JBOD environments, the 3U servers support high availability storage and mission critical business applications.

ASA 4U Series

Optimized for enterprise-level high-capacity storage applications, features 36x (24 front + 12 rear) 3.5" Hot-swap HDD bays, reliable and hassle-free maintenance storage system.

ASA Blade Series

Blade server with, Highest computing density (20 DP nodes and 2.56TB mempry in 7U) Fastest and Most Cost-Effective Networking Solution (Infiniband DDR/QDR support)

E-mail - sales@asacomputers.com
Call - 1800-REAL-PCS

ASA Computers, Inc
645 National Ave,
Mountain View. CA 94043
www.asacomputers.com

intel Xeon inside™

Powerful. Intelligent.

Intel, the Intel logo, Xeon, and Xeon Inside are trademarks or registered trademarks of Intel Corporation in the U.S. and other countries.

Philosophy and Fancy

There are no rules about what you can and can't do when you create your own Linux distro, but if you step back from the trees, the forest starts to emerge.

DAN SAWYER

Free and open-source software isn't just a way of computing, it's a way of life, right? Or, is it just a way to get cool free stuff? Or, is it the ultimate insurance against totalitarian government, corporatist/mercantilist rule? Or, maybe it's a toolkit that forms a common cultural substrate from which enterprises are built? Maybe not—I'm confused.

Actually, this kind of confusion isn't unusual for newcomers to the open-source space. Why in the hell would people release their software for free in the first place? Why would anyone want to *use* open-source software? What bearing do any of these questions have on choosing a distro? And, by the way, why do some distros strip the logo from Firefox or not include the ability to play DVDs or Internet videos?

Sometimes in life, philosophy dictates reality rather than vice versa, and one of the areas where this is at least partly the case is in the packaging, design and distribution of free and/or open-source operating systems.

What became the Open Source movement was born out of sixties and seventies radicalism, articulated by far-left quasi-Marxist anarchists like Richard Stallman, objectivist idealists like Eric Raymond, and colorful characters elsewhere along the political/philosophical spectrum. The idealism and the tensions that arose in that early hacking culture have gone on to infect the community at large and to affect the business models of some of the largest tech companies in the world. Even when the philosophy hasn't, the amount of free software that's simply lying around has turned into the DNA of much of the modern computing universe (Microsoft, for example, relies heavily on BSD code for its Windows Networking stack). Linux—perhaps partly because of the stated ideological neutrality of inventor Linus Torvalds and partly because of its broader popularity—can look like little more than a battleground for the different philosophical factions that gave birth to a number of major distributions. Some people even argue very loudly about whether the operating system we all love is Linux or GNU/Linux (speaking of which, those of you in the Stallman camp might want to start writing that hate mail now).

It's hard to believe in this era of supposedly mindless consumerism that something as arcane as a philosophy can impact how your computer functions, but there you go. There are actually a few different arenas in which philosophy plays a major role in the design of Linux distributions—some of them coming out of the tensions outlined above, and some of them out of more practical kinds of paradigmatic considerations. To understand why Linux distributions are put together the way they are, it helps to understand these axes.

MAJOR DIFFERENCES IN PHILOSOPHY

These differences in philosophy directly impact distribution design decisions, leading to three major species of Linux distribution.

The first species is the free distribution, and it follows the philosophy laid out by Stallman and his camp. This distribution accepts no patent-encumbered or commercial software, and it might not include hybrid software released under licenses like the LGPL or the Mozilla Public License. The Free Software movement initially arose in the 1970s and 1980s in the hacker culture, when people who owned computers were almost exclusively programmers and engineers. As commercial software came in with the dawn of personal computing, Stallman and his cohorts became concerned with their rights as users (a term which used to mean a lot more than it does now) and as programmers, so they founded the GNU Project, which eventually aimed to put together an entirely free (as in speech and as in beer) operating system. The GNU philosophy, articulated by the nonprofit Free Software Foundation and eventually embodied in the GNU General Public License, very cleverly uses copyright to subvert centralized control of information.

Free distributions are sometimes so devoted to the free software model that they will strip the trademarked artwork from, for example, Firefox, and substitute it for something more generic, in order to avoid having to deal with trademark legalities or other issues that might complicate the licensing of the distribution as a whole. Fedora Core and Slackware are prime examples of distros that follow this philosophy.

The second species is the commercial distribution, much maligned by the first camp. This sort of distribution takes advantage of the fact that the GNU GPL allows for commercial packaging (so long as the source code also is made available) and leveraging of GNU software together with various commercial and quasi-commercial programs to create a distribution that attempts to perform at the level of competing commercial operating systems and tends to aim for the same space. These often will have bundled proprietary tools, or a proprietary overlay with the system management tools, and generally will be licensed under a catchall proprietary license with a note that much of the included software is GPL, LGPL or another free license, but that the proprietary tools and software are not covered by conventional open-source thinking. Thus, caution is warranted when modifying or redistributing the operating system. SUSE is a currently viable commercial distro.

The third major species is the hybrid distribution, and its approach tends to be very pragmatic rather than ideologically slanted either toward the Free Software movement or toward commercial operating systems. These are the distributions that slant heavily in favor of free/open-source software, but still include (either bundled or via easily downloaded add-ons) proprietary fonts, codecs and other goodies of a legally murky nature (such as DVD ripping and decryption software) that often are not included in or available for commercial distributions short of compiling them oneself. This type of distribution is a recent arrival on the scene. Throughout the late 1990s and early 2000s, distros tended to be packaged either as commercial software or as free software + support, and sold in software boxes in computer shops. The burst of the dot-com bubble and the rise of widespread broadband, coupled with the legal problems caused by the DMCA, created a demand for distributions that delivered the benefits of free software with the functionality of commercial software, all while insulating distributors from the legal problems associated with patent infringement or enabling copyright circumvention. As a result, this last variety has become the most popular, although the other two species definitely are still around. Mandriva and Ubuntu both fall solidly in this category.

Sometimes in life, philosophy dictates reality rather than vice versa, and one of the areas where this is at least partly the case is in the packaging, design and distribution of free and/or open-source operating systems.

MAJOR DIFFERENCES IN APPROACH

Fights between licensing philosophies are not the only issues that help shape distributions. There's also the question of computing models. Sun Microsystems (which started life as Stanford University Networks) used to maintain that "The network is the computer." Because Linux is descended from System V UNIX—via its dependence on innovations made by the Berkeley Software Distribution (BSD)—and UNIX always has been a network-centric operating system family, it makes sense that some, or most, Linux distributions would follow this network-centric philosophy. Time-share computing and time-share-with-a-fresh-coat-of-paint (that is, Cloud Computing) are the major paradigms of the network-centric distribution. Two others are cluster-based computing and dumb terminals running remotely off a central server (both useful in scientific and commercial environments). Some flavors of Red Hat are specifically tailored to this computing model.

On the flip side, we have the desktop distribution. This is the operating system for the personal computing revolution. It stores the operating system and all the user's data locally (while the network-centric system prefers the opposite). These distributions are usually general-purpose, including a selection of software that can meet almost every need, from setting up a home Web server to running a small business, from playing games to Web browsing and word processing or producing a podcast. The desktop distribution is the Swiss Army knife of Linux distros. Ubuntu, SUSE and Mandriva show this approach in action.

You can see a vestige of the early heritage of your particular distribution by looking at the filesystem structure. Does your package manager install software to /usr/local/* or to /usr/*? If the former, your distro probably started life as a network-centric operating system for an office environment. If the latter, your distro has probably been designed (or, in some cases, redesigned) with the desktop in mind.

Alas, there are some things for which the Swiss Army knife just isn't suited, and in the last four years, several custom-purpose distributions have come on the scene to solve the shortcomings of the desktop distribution for different specific purposes. The most obvious of these are the studio distributions, customized for real-time audio and video production in high-demand environments, but there also are customized distributions for firewalls, Web servers and laptops as well as market-specific distros targeting churches, activist

groups, hackers and crackers, and grandparents (that is, users who are incapable of interacting with their machines as anything other than appliances).

Moving beyond the customized distro space, there's an entire field of customized Linux distributions that deserves special mention: the live CD. Knoppix was the first mover here, and since then, the space has exploded. With a live CD, you can run Linux on almost any hardware, including the programs you use most often (if one live CD doesn't have it, chances are another probably will), without touching the machine's hard drive. Live CDs are very useful for diagnostics, for testing whether a distribution will play nice with your hardware or for taking a familiar environment into hostile territory (for example, when visiting relatives whom you don't want to find out that you like visiting dolphinsex.com purely for research while writing your latest romantic epic about trans-species love among aquatic mammals).

No discussion of the different approaches would be complete without mentioning embedded distributions—versions of Linux and derivative operating systems (such as RockBox and Android) designed to run on handheld devices, in networking appliances, NAS Servers and dozens of other gadgets, toys, tools and machines that consumers love to use and hackers love to repurpose. Some of these you can find for download on the Web, but a greater number are created and used in-house at different companies that manufacture devices of different sorts and often include a goodly amount of proprietary code to interact with the device's firmware.

MAJOR DIFFERENCES IN ADMINISTRATION

There's a third axis along which distributions sort themselves out, and that has to do with how you answer the question "Whose job is it to administrate the system?"

Linux's architecture segregates system functions from user access—a major reason that Linux has proved remarkably insusceptible to viruses and worms. In a classical setup, what I'll call office administration, this means that only the root account can install and remove software, monkey with system settings, load and unload kernel modules, and change the hardware. A user account may merely use the software and access the data files generated by that particular user or shared with it by another user. This is still the most common setup, and it's useful in small-office, home-office and family environments where more than one user will be accessing a given system regularly.

However, laptops and Netbooks often don't need this kind of strict segregation, because the user almost always also is the system administrator. Those distributions aimed at this market and at the single-user desktop operate according to a home administration model—that is, to remove the encumbrance of having to log in to root separately, a number of modern distros do not enable the root account by default. Instead, the primary user is also the sysadmin and must furnish only a password to perform administrative functions. Ubuntu and its derivatives use this scheme by default, although they easily can be converted to the more classical administration method.

The final major administrative paradigm is most commonly encountered in embedded systems and appliances. These gadgets, such as your trusty blue-box Linksys router, are generally headless and are administered remotely from a master system over SSH or (more commonly) through an easy-to-understand Web interface.

MAJOR TYPES OF USERS

The last axis that distributions tend to fall along isn't exactly philosophical, but it still is a real consideration. If you're putting together a distribution, who is your end user?

A distro put together for a hacker will, by default, install a raft of security-auditing and coding tools, and perhaps a handful of black hat tools as well, just for fun. These utilities almost always are available on other distros, but rarely do they install by default. Slackware is aimed squarely at this demographic.

On the other hand, a distro designed with the end user in mind might install some games, an office suite, a media player or two and a handful of toys, but generally its goal is to not overwhelm users with options. Give users one tool that works reasonably well and a good package manager, and let people fill up the corners as they need to. A small footprint and an idiot-proof setup is the name of the game here. Xandros and Linux Mint are prime examples of this type of targeting successfully taken to its logical extreme.

Power users are another animal entirely. These users want to be able to tinker with the system right up to the end of their expertise, and one tool for one job is never enough. After all, when was the last time you saw a perfect tool? Power users want it all and know how to milk the last bit of functionality out of their systems. They want an OS that is heavily optimized, that crashes rarely (if ever), and that they can control down to the process level without necessarily having to get their hands dirty. Systems designed for power users tend to be both slick and decadent, armed to the teeth with the kind of tools that would make most end users crawl into a straitjacket and whine for their mommies. The studio distributions, such as PlanetCCRMA and 64 Studio, tend to be this variety.

The administrator is the last major species of user, and distros designed for these users tend to be thin on flash and heavy on function. They also tend to run in environments where no end user would ever dare to tread, like server farms, render clusters and data centers.

WRAPPING IT UP

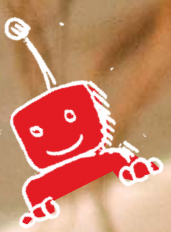
The plethora of Linux distributions can be enough to make anyone's eyes glaze over, and it's tempting (particularly for a newcomer to the F/OSS world) to chalk up the diversity in distro design as just another symptom of a fractious, political hacker culture with more time than sense. The reality is a little different. The axes outlined above demarcate the major design considerations that distro packagers have to weigh, and where you put your pin on each axis also generally will have knock-on effects in terms of what kind of hardware you want to support, whether and when you will ship kernel source, whether you ship with a modular kernel, what kind of package manager you'll use, and how you'll manage the community that springs up around your product.

Likewise, understanding these considerations can help make the F/OSS space look a lot less like a battleground or a junkyard and lot more like what it really is: a rich ecosystem, more than a bit jungle-like, of different designs evolving to serve different economic and ecological niches, always filled with interesting experiments—some of which succeed handsomely.■

Dan Sawyer is the founder of ArtisticWhispers Productions (www.artisticwhispers.com), a small audio/video studio in the San Francisco Bay Area. He has been an enthusiastic advocate for free and open-source software since the late 1990s. He currently is podcasting his science-fiction thriller *Antithesis* and his short story anthology *Sculpting God*. He also hosts "The Polyschizmatic Reprobates Hour", a cultural commentary podcast. Author contact information is available at www.jdsawyer.net.

Lullabot-Powered

The most super powered sites in the world are created in Drupal, by you and Lullabot.



Suzi Arnold
Director of New Media
Sony Music

Lullabot™ New Lullabot Learning Series training DVDs at Lullabot.com

SMALL SYSTEMS and BIG IRON:

Linux on Non-x86 Computers

Discover the options for running Linux on PowerPC, ARM and Itanium.

KIRA SCARLETT

Thousands of Linux distributions run on Intel-compatible x86 PCs, but these systems actually make up only a minority of the computers produced. Other architectures, such as MIPS, SPARC, ARM, Power and Intel's Itanium chip are used heavily in certain applications. These computers historically have been the domain of high-end UNIX variants or specialized embedded operating systems, but Linux has taken both embedded systems and big-iron server systems by storm. In this article, I explain the important Linux distributions for ARM, IBM Power and Intel Itanium and their features.

ARM

ARM is the world's most-popular 32-bit processor. TVs, set-top boxes, portable devices and large numbers of other device categories contain ARM processors. They also are starting to be used in Netbooks due to the poor performance and high power consumption of the Intel Atom processor. ARM chips are popular due to their simple design, decent performance, low power consumption and for the massive amount of code that already runs on the architecture.

Most ARM processors, especially lower-end ones, run tiny real-time operating systems like QNX and VxWorks. On higher-end devices, however, Linux is becoming a major player. Google's successful Android cell-phone OS is a Linux system, although a heavily customized one. Nokia also supports Linux on its high-end smartphones and Internet tablets with its Maemo operating system, a Debian derivative with hundreds of Linux programs ported to it.

For hobbyists, the choice of distribution for ARM largely is defined by the available hardware. Most hobbyists do not have the time or resources to design their own Linux ARM board and develop a board support package for it, but a few hobbyist-friendly Linux hardware platforms have been developed.

Starting with version 9.04, Ubuntu officially supports the ARM platform. Canonical releases images only for a fairly obscure Freescale development board, but the Linux community has created ports to numerous other ARM computers, including the BeagleBoard and the N800 and N810 Internet tablets. Sharp even released an Ubuntu/ARM Netbook in Japan, the PC-Z1 NetWalker; however, it received a poor reception due to substandard build quality and a miniscule optical trackpad.

As Canonical is investing considerable resources into the Ubuntu ARM port, it is expected to become significant as an OS for Netbooks and similar devices in the future. If you have compatible hardware, the Ubuntu ARM port is of very high quality. If you're interested in experimenting but don't have compatible hardware, Ubuntu also can be run in QEMU. Ubuntu's wiki has tutorials for doing so, although the procedure is somewhat complicated.

Quite a few mobile Internet devices and other small computer systems are using Ubuntu or Ubuntu variants as their default operating system, and you can expect Ubuntu for ARM to become even more important as Canonical invests more money into its continued development.

Nokia's Maemo platform has received quite a bit of attention lately. It originally was released on the company's N770 tablet and was used by its successors: the N800/N810 Internet tablets and the N900 smartphone. Heavy interest developed in the platform after its use in the N900. Maemo is a full-featured Debian system with a custom user interface. It uses a stripped-down X server, KDrive and a touchscreen-friendly GTK-based desktop environment called Hildon. Nokia also provides Scratchbox, a package containing an SDK and emulator. Unlike most consumer embedded platforms, Maemo automatically can update itself with its APT-based package management system.

Although Nokia supports Maemo only on its Internet tablet systems, its components are being ported to other operating systems and devices. A community project, Mer, exists to develop a new distribution based on code from both Maemo and Ubuntu and to port the new system to a wide variety of mobile devices with both ARM and x86 processors. Maemo probably still is a superior operating system on the Internet tablets due to its high level of completeness, but Mer is under heavy development and becomes more polished with every release.

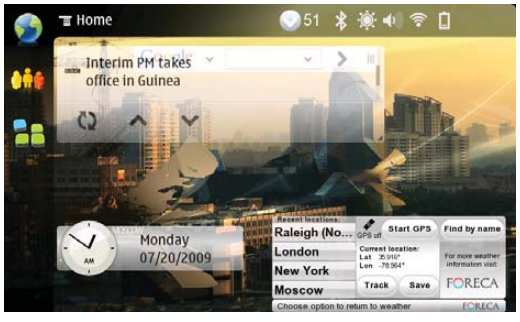


Figure 1. Maemo Screenshot

The Android operating system, developed by Google and its partners, is probably the fastest-growing operating system for smartphones and is becoming increasingly popular on tablet computers, such as the Dell Mini 5. Analysts predict that hundreds of millions of devices will ship running Android in the next few years, making it one of the most important embedded operating systems. Android runs a Linux kernel and has a basic shell, but in other ways, it's very different from most Linux distributions. It uses a custom window system that's very different from X11, which is what most users and developers are used to. It has a high-level application framework, WindowManager, that is backed by a low-level library called SurfaceManager. Android applications are written in Java and executed by the Dalvik Virtual Machine, which is designed to have a low memory footprint and lacks Just-in-Time Compilation, which is an optimization feature provided by most mainstream Java virtual machines. Clever developers and hobbyists have found that Android is easily tweakable, and have figured out how to run OpenSSH and X11 on the platform for those who prefer a more traditional Linux environment.

Ångström

The most popular ARM Linux distribution for hobbyists is Ångström, built by a merger of several older embedded Linux projects. Ångström uses a low-overhead package manager called ipkg, similar to the dpkg system used in Debian. An interesting feature of this distribution is the ability to generate a custom system image on the Web site, with user-defined packages and capabilities. During the past few months, developers have released plenty of software as Ångström packages, including GNOME, Firefox, GIMP, Quake, Mono and dozens of other pieces of software. Ångström also is used as the base for other embedded Linux distributions.

PowerPC/Power

The PowerPC architecture was jointly developed by IBM and Motorola around 1990 and is used on a very wide range of devices. All three current major game

consoles use PowerPC-based processors, as do many routers, onboard computers in cars and aircraft, and high-end servers from IBM. Although PowerPC largely has faded from desktop computers since Apple switched to Intel x86 in 2006, use of the processor for embedded and server purposes remains a multibillion-dollar industry.

During the peak of PowerPC desktop usage, most Linux distributions offered builds for the architecture. Sadly, this is no longer the case. Novell and Red Hat both offer Enterprise Server distributions for Power, but they aren't cheap and really are designed only for current IBM servers. Although their distributions are mature and well supported, most consumer PPC/Power hardware actually consists of older Macs and the EFIKA line of computers from Genesi. For these systems, the Enterprise Server distributions of SUSE and Red Hat Linux aren't really optimal. Thankfully, the Linux community still provides quite a bit of support for this architecture, largely due to the significant amount of inexpensive PPC hardware from before Apple's switch to Intel. The Sony PlayStation 3 also has provided the Linux community with inexpensive, but powerful hardware. Its main limitation is the fact that it has only 256MB of RAM.

Historically, one of the primary Linux distributions for PowerPC was Yellow Dog Linux. YDL is based on Red Hat Linux and uses RPM to manage packages. Its hardware support is generally quite good. The distribution runs on PowerPC Macs, IBM servers and workstations, and the PlayStation 3. YDL has some fairly significant differences from other distributions, such as using Enlightenment 17 as the default desktop environment. E17 offers quite a few advantages over both Enlightenment 16 and conventional desktop environments. It's far lighter on resource consumption than KDE, GNOME or Xfce, while offering a much larger feature set than most low-footprint window managers. Almost every element of the user interface is customizable with graphical tools and plugins. E17 also includes a built-in file manager, unlike previous versions. However, E17 still is under heavy development and may not be as stable as mature versions of GNOME or KDE. For users who prefer them, Yellow Dog also has packages for other desktop environments and window managers.

Fedora also offers up-to-date PowerPC ISOs of every version, including the latest, Fedora 12. Fedora offers a more complete default installation and more application packages, but also is considerably more resource-intensive. It attempts to give a GNOME or KDE environment comparable to the x86 build of the same version. This means if you're used to Linux systems on x86, you'll be more at home with Fedora than with Yellow Dog, but it also means it's noticeably sluggish on older systems. Generally, I'd recommend Fedora for systems with a 1GHz or faster G4 or G5, and at least 512MB of RAM. It is important to note that support for Fedora on the PlayStation 3 is experimental and in a very early development stage, although there have been efforts to finish the port. Fedora 12 will be the last officially supported version for PowerPC, but there are efforts to provide community-supported PowerPC builds of Fedora 13 and later versions.

Figure 2. Enlightenment Desktop (E17)

Ubuntu ended official support for PowerPC in late 2006 with Ubuntu 6.10. Since then, there have been PPC builds of every Ubuntu release developed by the community. These generally are very high quality and have excellent stability. Ubuntu has similar system requirements to Fedora, with most older hardware probably being too slow to handle it smoothly. Ubuntu also offers an ISO for the PlayStation 3, but it's still experimental and somewhat limited in features due to the PS3's insufficient amount of RAM, so using the live CD is likely to be unpleasant.

Itanium

Itanium, also called IA64, was the last attempt by Intel to replace the aging x86 architecture, following the disappointing iAPX432 and i860 processors in the 1980s and 1990s. At one time, industry analysts considered it to be the future of the high-end servers, and Intel planned to use the architecture in personal computers eventually as well. Itanium uses a unique architecture, neither RISC nor CISC, that can execute several instructions per cycle in parallel. SGI and HP replaced their own high-end processors with Itanium out of the expectation that it would bring a revolution in performance. However, the first-generation Itanium core, code-named Merced, delivered disappointing performance results while consuming massive amounts of power and prevented the market-share breakthrough that Intel had hoped for.

With the release of the Itanium 2 core in late 2002, performance increased significantly, but the platform's reputation had been hurt by the low quality of the first-generation processors, and Itanium remains a high-end product with low market share. Intel still claims to be committed to the architecture. A new quad-core Itanium chip, code-named Tukwila, was released in February 2010, and according to Intel, at least two more generations are under development.

HP is the last major manufacturer of Itanium-based workstations, and it still makes more Itanium servers than all other companies combined. HP's workstations, the ZX2000 and ZX6000, are easily available secondhand and have excellent Linux support.

The only remaining commercial distribution with any focus on Itanium is SUSE Linux Enterprise Server. Red Hat offers a version of Red Hat Enterprise Linux for Itanium, but support for the platform will be ended in version 6.

Red Hat Enterprise Linux 5 will be maintained and receive bug fixes until 2014, so if you don't mind missing out on feature updates, you can purchase a license from Red Hat. Red Hat also offers a free trial version, which blocks access to update repositories after 30 days.

If you have a large Itanium system or cluster, SUSE Linux makes a lot of sense due to its active development status and excellent support for virtualization, but for smaller installations and hobbyists, several other systems are available, including Gentoo and Debian. Fedora and Ubuntu both define Itanium as a secondary platform and occasionally provide new releases or updates, but both largely have ignored the architecture in recent releases. The last build of Fedora available for Itanium was Fedora 9, and Ubuntu releases for the platform have been broken or seriously buggy since 8.04. HP-UX, OpenVMS and Microsoft Windows also are available for Itanium.

Debian probably is the most stable and modern IA64 Linux distribution available for free. It still is an officially supported Debian platform, and still is under active development. It offers all the features of x86 Debian, including a full GNOME desktop environment. Debian for Itanium has access to the full Debian package set and generally is fairly stable. In recent months, development has declined somewhat due to declining overall interest in the Itanium platform, but it appears that IA64 will continue to be a supported platform for the foreseeable future. An occasionally active mailing list exists for users and developers of the Itanium port of Debian.

Red Hat and its derivatives occasionally have run on Itanium. Red Hat is preparing to release Red Hat Enterprise Linux 5.5, which will support the platform natively, but it might not be a good idea to run RHEL on a new Itanium system due to the fact that version 5 will be the last release available for IA64. CentOS, an open-source clone of RHEL, no longer supports Itanium, although the developers have suggested that future releases may run on it, as well as other less-common architectures including SPARC and Alpha. Fedora was available for Itanium through version 9, and some RPMs from version 10 are available. Overall, the Fedora experience on Itanium isn't bad with all updates installed, but users who want more up-to-date packages may end up needing to use source RPMs or compiling software themselves.

Conclusion

Of the platforms discussed in this article, Power and especially Itanium both have a questionable future. Although IBM and Intel are committed to developing future generations of their products, the market for high-end proprietary processors has been somewhat eroded by increasingly fast and cheap x86 processors. Many analysts felt that the Tukwila Itanium was underwhelming in comparison to both Power7 and high-end x86 server processors, such as recent Xeon and Opteron chips. IBM expects Power to be a viable platform for a long time, because it still is substantially faster than the x86 alternatives, but even for them, competition is closer than it once was. As a result, Linux support for these platforms probably is going to decline over time, although as long as there is hardware using these architectures, people will be using and developing Linux on them, as has happened with "dead" architectures, such as the DEC Alpha and the HP PA-RISC. ARM, on the other hand, has a bright future, having been dominant on low-power systems for decades and starting to become popular on consumer computer hardware, with constantly improving Linux support.■

Kira Scarlett has been using Linux for eight years. She frequently ends up owning strange and unusual computer hardware, and she has used Linux on almost every major processor of the last 20 years. She also is interested in graphic design and is an avid hockey fan. Kira can be reached at kira_arc4@ovi.com.

Resources

Scratchbox: **scratchbox.org**

Ångström: **angstrom-distribution.org**

Gelato: **gelato.org**

Ubuntu PowerPC Wiki: **wiki.ubuntu.com/PowerPC**

The 1994–2009 Archive CD, back issues, and more!

www.LinuxJournalStore.com

LINUX DISTRIBUTION CHART

For this *LJ* distro chart, we selected distributions and categories based on suggestions from *Linux Journal* editors and readers, and gathered the information from each distro's Web site and DistroWatch.com. *Linux Journal* readers shared their comments, favorite distributions and thoughts about each distro's best use in our readers' poll on LinuxJournal.com. We include a few readers' comments here with the chart, but be sure to visit **www.linuxjournal.com/content/what-each-distribution-best** and **www.linuxjournal.com/content/which-linux-distribution-do-you-use-most-frequently-0** for many, many more comments and to add your own feedback—we're sure we left out at least a few people's favorites! Note that under the "Best for" category on the chart, all distributions were voted as favorites on both desktops and servers, so in the interest of avoiding repetition, we left those out. Also note that in the on-line readers' poll for "Most Frequently Used Distro", 2% voted for "other".

JUSTIN RYAN

"I use **Debian** for its stability, support and availability of third-party packages and programs."

"Debian combines great sysadmin friendliness with a terrible release policy; **Ubuntu** takes its great design and adds sanity."

"I now use **Fedora** because each successive version of Ubuntu caused different problems with my 3.5-year-old laptop (camera, sound, wireless, graphics). Each version would fix some problems and cause others. Fedora has been stable, fast and less trouble to set up than Ubuntu."

DISTRIBUTION	LATEST STABLE RELEASE (DATE)	FIRST RELEASE	RELEASE CYCLE	SUPPORT LIFECYCLE	BASED ON	DEVELOPED BY	SPONSORED BY	PACKAGE FORMAT	PACKAGE MANAGEMENT
Arch Linux	2009.08 (08/10/2009)	03/11/2002	3–4 months (follows kernel releases)	None (rolling releases)	None	Aaron Griffin & Community	None	tar.gz	Arch Build System, Packman
CentOS	5.4 (10/21/2009)	12/2003	2 years (follows Red Hat Enterprise Linux)	7 years	Red Hat Enterprise Linux (open-source SRPMs)	CentOS Project	None	rpm	RPM, YUM, up2date
Debian	5.0 "Lenny" (02/14/2009)	08/16/2003	2 years (beginning with 6.0 "Squeeze")	3 years	None	Debian Project	None	deb	dpkg, APT, Synaptic
Fedora	12 "Constantine" (11/17/2009)	11/05/2003	6 months (approximate)	13 months (approximate)	Historically: Red Hat Linux	Fedora Project	Red Hat	rpm	RPM, YUM, PackageKit
Gentoo	None (versionless)	03/31/2002	Rolling releases	None (rolling releases)	None	Gentoo Foundation	None	ebuild	Portage
Linux Mint	8 "Helena" (11/28/2009)	08/27/2006	6 months (follows Ubuntu)	18 months (follows Ubuntu)	Ubuntu	Linux Mint Team	None	deb	dpkg, APT, MintInstall/ MintUpdate
Mandriva	2010 (11/03/2009)	07/23/1998	6 months	18 months (base updates); 12 months (desktop updates); 24 months (server updates)	Historically: Red Hat Linux	Mandriva S.A.	Mandriva S.A.	rpm	urpmi/rpmdrake
Mepis	8.0.15 (01/12/2010)	05/10/2003	Unspecified (6 months to 1 year)	Unspecified	Debian/Ubuntu	MEPIS LLC	MEPIS LLC & Community	deb	dpkg, APT
openSUSE	11.2 (11/12/2009)	03/1994	8 months	2 releases + 2 months	Historically: SUSE Linux	openSUSE Project	Novell	rpm	RPM, YaST, Zypper
PCLinuxOS	2009.2 (06/30/2009)	11/2003	Unspecified	Unspecified	Historically: Mandriva	PCLinuxOS Development Team	None	rpm	APT-RPM, RPM, Synaptic
Puppy Linux	4.3.1 (10/17/2009)	06/18/2003	Unspecified	Unspecified	None	Puppy Community	Puppy Foundation	.pup, .pet	PetGet
Red Hat Enterprise Linux	5.4 (09/02/2009)	03/26/2002	18–24 months	7 years	Fedora	Red Hat	Red Hat	rpm	RPM, YUM
Slackware	13.0 (08/26/2009)	07/16/1993	Unspecified	N/A	Historically: Softlanding Linux System	Patrick Volkerding & Community	Slackware Linux, Inc.	txz/tgz (tarball)	installpkg/ upgradepkg (pkgtool)
SUSE Linux Enterprise	11 (03/24/2009)	03/1994	Major: 24–36 months; Service Packs: 9–12 months	5–7 years	openSUSE	Novell	Novell	rpm	YaST, Zypper
Ubuntu	9.10 "Karmic Koala" (10/29/09); long-term support "Hardy Heron" (04/24/2008)	10/20/2004	Biannually (April/October)	18 months; long-term support: 3 years for desktop, 5 years for server	Debian	Ubuntu Community	Canonical	deb	dpkg, APT, Synaptic, Ubuntu Software Center
Yellow Dog	6.2 (06/29/2009)	1999	Unspecified	Whichever is longer—1 year from launch or 3 months from new version	RHEL, CentOS	Fixstars Solutions	Fixstars Solutions	rpm	YUM

"I think most distros are a lot more polished and user-friendly than they were a few years ago, but I'm going to go with **Ubuntu**. I used to use Kubuntu from 6.06 to 8.04, but the transition from KDE 3 to KDE 4 hasn't been the smoothest. I gave regular Ubuntu 9.10 a spin and have been really impressed, since it's probably the first GNOME-based distro I've actually enjoyed. There are practical reasons for going with Ubuntu as well. Canonical has done a great job getting it out there and making it known, as well as presenting it as an OS for everyday users and not just networks and servers. And, the fact that it's such a popular distro means there are lots of users posting how-tos and solving common problems."

"I wouldn't say that '**Ubuntu** is the most easy Linux for everyone'. I definitely would agree that it is the one with the catchy-hard-to-forget name, in the HD-DVD vs. Blue-ray vein. When you put that aspect with the fact that it is free, then you get the 'World's Most Popular Linux Distro', whether it's the easiest one or not. Lots of people who don't particularly care about 'free' don't care about Ubuntu—especially the learning curve required to 'fix it'. These people, willing to pay for quality software and OS, are an admitted minority in the Linux camp, but they *do* exist. I don't, however, think that these people give a fig about Linux 'touching the masses' and all the underlying tones of cloying that phrase implies. They just want an easy-to-use and efficient/intuitive OS that works without hassle. Ubuntu is the most popular Linux one, but not the easiest Linux one."

Arch Linux

"Reasons I use Arch:

1. Rolling upgrade.
2. Up to date packages.
3. Awesome community/ documentation.
4. Great performance.
5. Minimalist design.
6. Simple from top to bottom.
7. Teaches me as I go."

"I prefer **Slackware** because it's very simple and stable. It gives me the power I need to get things done very efficiently."

"I use **CentOS** simply because of its reliability. It's also flexible, and very light—with it being light leaves more resources to actually do what you want. Hence, that's why I use it for all my servers."

"I've been using Windows for a long time, since Windows 95, and I've been an IT professional for about 9 years. Through it all, I've always been turned off to Linux. I didn't have time to try anything new. I was just trying to keep up with the changes in Windows. Just a month ago, a new coworker gave me a **Linux Mint** CD. I took it home and ran the live CD on one of my IBM laptops. I've been hooked ever since. I even changed my wife's laptop from XP to Mint. The bottom line is, Linux just works....I'm sold."

DEFAULT DESKTOP ENVIRONMENT(S) (VERSION)	LINUX KERNEL	DEFAULT FILESYSTEM	OFFICIAL PORTS	DERIVATIVE DISTRIBUTIONS	MOST FREQUENTLY USED (READERS' POLL)	BEST FOR (READERS' POLL)
None (user selected	2.6.32.3	None (user selected)	x86, x86-64	None	7%	Ease of upgrade, education, older hardware
GNOME (2.16)	2.6.18	ext3	x86, x86-64	None	2%	Ease of installation, proprietary hardware support, security
GNOME (2.22), alternate CDs: KDE, Xfce, LXDE	2.6.26	ext3	x86, Alpha, SPARC, PowerPC, ARM, MIPS, Itanium, HP PA-RISC S/390, AMD64, ARM EABI	Ubuntu, Knoppix, Damn Small Linux, Linspire, Maemo	9%	Ease of upgrade, getting support, security
GNOME (2.28), Fedora Spins: KDE, LXDE, Xfce	2.6.31.5	ext4	x86, x86-64, PowerPC	Red Hat Enterprise Linux, Yellow Dog Linux, Moblin	9%	Ease of installation, new users, security
None (user selected)	2.6.32	None (user selected)	Stable: x86, x86-64, PA-RISC, PowerPC, SPARC 64-bit, DEC Alpha; Development: MIPS, PS3, System Z/s390, ARM, SuperH	Sabayon	4%	Education, older hardware, real-time apps
GNOME (2.28); Community: KDE, Xfce, Fluxbox	2.6.31	ext3	x86, x86-64	None	7%	Ease of installation, multimedia, new users
KDE (4.3.2), GNOME (2.28.1), Xfce & twm	2.6.31.12	ext4	i586, i386, x86-64, PowerPC, MIPS, ARM	PCLinuxOS	6%	Ease of installation, education, new users
KDE (3.5)	2.6.22.14	ReiserFS, ext3	x86, x86-64	SimplyMEPIS, antiX	2%	Ease of installation, new users, older hardware
GNOME (2.28), KDE (4.3.1)	2.6.31	ext4	x86, x86-64	SUSE Linux Enterprise	11% (with SUSE Linux Enterprise)	Ease of installation, new users, proprietary hardware support
KDE (3.5.10)	2.6.16	None	x86	None	4%	Ease of installation, multimedia, new users
JWM/IceWM	2.6.30.5	SquashFS (ext2)	None	None	1%	Ease of installation, new users, older hardware
GNOME (2.16)	2.6.18	ext3	IA-32, x86-64, PowerPC, i386, ia64, s390, s390x	CentOS	1%	Getting support, proprietary hardware support, security
Blackbox, Fluxbox, FVWM, KDE (4.2.14), WMaker, Xfce; Community: GNOME	2.6.29.6	ext4	x86, x86-64, IBM S/39	Slam64, SLAX, VectorLinux	4%	Education, older hardware, security
KDE (4.1), GNOME (2.24)	2.6.27.19	ext3, JFS, ReiserFS	XFSIA-32, x86-64, PowerPC, Itanium	None	11% (with openSUSE)	Getting support, proprietary hardware support, security
GNOME (2.28)	2.6.31; long-term support: 2.6.24	ext4; long-term support: ext3	x86, x86-64, ARM, SPARC	Kubuntu (KDE), Edubuntu, Xubuntu (Xfce), Ubuntu Studio, Linux Mint, Crunchbang, Ubuntu Netbook Edition	31% (any flavor)	Ease of installation, getting support, new users
Enlightenment, GNOME (2.16.0), KDE (3.5.4)	2.6.29	ext3, JFS, ReiserFS	XFSPower	None	0%	Gaming, older hardware, proprietary hardware support

"I've had nothing but utterly awful experiences over ten years with **RHEL**, despite its high cost. I can see the point of CentOS if you need RH without the cost, but it's just revolting to work with and the documentation is terrible too, so I'd never run either by choice."

"In the lightweight division, we have used **Puppy Linux** a *lot* of the time, installing it to HDD on a half-dozen of the same GoBook P3 laptops and giving them to kids as gifts. At around $50 each (well used), this was affordable."

"Don't forget **SliTaz** though. I have it on my old 433MHz Celeron machine, and it is fantastic. If you need a lightweight Linux distro for old hardware, I would take this over Puppy Linux any day."

"I love live CDs, but liked PCLOS Big Daddy so much, I felt the need to install it with a dual-boot of Windows at the time. By the time PCLOS 2007 came out, I'd gotten a newer computer and erased the Windows partition to put the exclusive Linux desktop on it. I haven't looked back since. I no longer dual-booted. The other people I know who have **PCLinuxOS** tend *not* to be techie types that you see at work, but more like teenagers and housewives and early-adopter-gadgety folk around here—not the Computer Crowd, as much as the people with *lots* of cool toys. They don't dual-boot either. When VirtualBox came to Synaptic repos years ago, I put my Windows XP install disk in there to test it out and made a video of Linux running Windows better than Windows. I ended up taking the virtual Windows off though, because I never used it. PCLinuxOS rules."

"I like **Gentoo** for its extremely useful control over the system and love the flexibility. It appeals to the tweaker in me! All my systems, including laptops, run Gentoo! That's five systems in total! I have tried other distros, but nothing comes close to Gentoo. I loved portage so much, at some point in time, I ported it to Solaris. Now, with prefix support, anybody can use portage on Solaris, BSD or Mac OS. The Gentoo community is exemplary!"

An Introduction to MINIX

It's not Linux, but MINIX can introduce you to the basic concepts without all the baggage.

BRUCE BYFIELD

Remember MINIX? Short for Minimal UNIX, MINIX is a close cousin of GNU/Linux. To GNU/Linux users, it is simultaneously familiar and foreign, and it challenges orthodox assumptions about how an operating system should be designed.

MINIX originally was developed in 1987 by Andrew S. Tanenbaum as a teaching tool for his textbook *Operating Systems Design and Implementation*. Today, it is a text-oriented operating system with a kernel of less than 6,000 lines of code. MINIX's largest claim to fame is as an example of a microkernel, in which each device driver runs as an isolated user-mode process—a structure that not only increases security but also reliability, because it means a bug in a driver cannot bring down the entire system.

In its heyday during the early 1990s, MINIX was popular among hobbyists and developers because of its inexpensive proprietary license. However, by the time it was licensed under a BSD-style license in 2000, MINIX had been overshadowed by other free-licensed operating systems.

Today, MINIX is best known as a footnote in GNU/Linux history. It inspired Linus Torvalds to develop Linux, and some of his early work was written on MINIX. Probably too, Torvalds' early decision to support the MINIX filesystem is responsible for the Linux kernel's support of almost every filesystem imaginable.

MINIX remains a shell-based operating system, and its concessions to the desktop are minimal.

Later, Torvalds and Tanenbaum had a frank e-mail debate about the relative merits of macrokernels and microkernels. This early history resurfaced in 2004 when Kenneth Brown of the Alexis de Tocqueville Institution prepared a book alleging that Torvalds borrowed code from MINIX—a charge that Tanenbaum, among others, so comprehensively debunked, and the book was never actually published (see Resources).

Now at version 3.1.6, MINIX has taken a turn in its development. While versions 1 and 2 focused primarily on the operating system as a learning tool, with version 3, MINIX began targeting low-cost laptops and embedded devices as well. More generally, the project's Web page recommends MINIX for "applications where very high reliability is required" and for projects where the GNU General Public License would be too restrictive.

However, these new targets seem more ideal than real. I can find little evidence of MINIX being used in embedded devices or for its high reliability or licensing. Similarly, MINIX still lacks the user-friendliness that would make it a candidate for a project like One Laptop Per Child. As with previous releases, MINIX's greatest value continues to be as an educational aid to give users experience of another UNIX-like system.

Still, for those familiar with GNU/Linux, MINIX does take some acclimatization. Accordingly, what I present here is not a comprehensive review, but an introduction to help those who are interested in orienting themselves to MINIX, its structure and resources.

Installing MINIX

MINIX's hardware requirements (see Resources) should not be a major concern for most users. Requiring 16MB of RAM and a gigabyte of hard drive space, MINIX should install on most computers made in the last decade, even if all peripheral devices are not supported.

If you are interested mainly in studying MINIX, you might consider installing it in a virtual machine. MINIX is installable with a variety of virtualization solutions, including Bosch, QEMU, VMware and VirtualBox. Instructions for each solution are available on the project Web site. Installing MINIX as a guest operating system has the advantage of allowing you to make easy comparisons with a GNU/Linux host.

No matter how you decide to install MINIX, have some paper ready to take notes. Some on-line instructions are available, but, at the time of this writing, they differ so significantly from those provided by the installer that they are not reliable. The first set of instructions (Figure 1) is especially important because it explains the following:

- That you install with the setup command.
- How to shut down the system.
- That you use xdm to start the X Window System.
- That you use packman to install additional packages.

What the instructions do not mention is that you can log in after installation as the root user with no password.

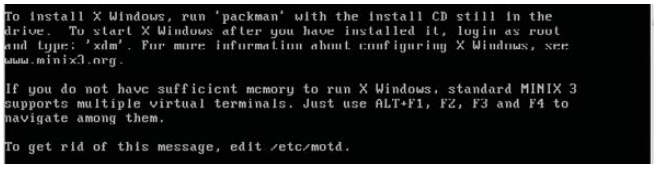


Figure 1. Bootup Instructions

Although text-based, the MINIX installer should provide

few obstacles for anyone who has installed operating systems in the past. Probably the biggest standard challenge is to do expert partitioning, because MINIX has its own system. However, the default partition scheme, which includes separate partitions for /, /home and /usr, should be satisfactory for most users.

A more serious problem for some people will be the fact that MINIX supports only eight common Ethernet cards; the installer does, however, auto-detect cards.

True to its name, MINIX installs a minimal system. One of the installer's final warnings is that the first time you start the new installation, you should add users and passwords.

If you are installing on a multiboot system, you also need to add MINIX to the bootloader. For example, if you are using Legacy GRUB and MINIX is installed on the second partition of the first hard drive, the stanza in /boot/grub/menu.list would be:

```
title MINIX
rootnoverify (hd0,1)
chainloader +1
```

As with Windows, GRUB does not support MINIX natively and has to pass off its booting to MINIX's own bootloader.

Navigating MINIX

MINIX remains a shell-based operating system, and its concessions to the desktop are minimal. It starts with a boot menu of different system states, including (assuming you followed the install instructions) a pristine version of the operating system that you can use for recovery. When you are finished, the command `shutdown` halts the system, and `shutdown -r` reboots it.

For anyone who has used a UNIX-like system, the MINIX directory hierarchy should be broadly familiar (Figure 2). However, you will notice a few missing top-level directories, such as the ever-contentious /opt, and directories added to operating systems like GNU/Linux for user-friendliness, such as /cdrom and /media. Also missing is /proc, which tells you that the pseudo-filesystem procfs does not exist to access process information from the kernel. Because MINIX runs drivers in userspace, it does not have the need for /proc that GNU/Linux does.

Figure 2. File Hierarchy

Descend a directory level, and you find that the logic of the directory hierarchy is differently applied. For instance, GNU/Linux's /var/spool directory, which contains queues for cron jobs, printing and mail as well as locks, is located in /usr/spool instead. But, such examples are exceptions, and previous experience with UNIX-like systems can only benefit those exploring MINIX for the first time.

What may require more acclimatization is MINIX's naming system for devices. Open /etc/fstab, and, if you accepted the default partitioning scheme during installation, you will see something like:

Advertiser Index

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, 3, 37, 39	Lullabot www.lullabot.com	57
Aberdeen, LLC www.aberdeeninc.com	23	Microway, Inc. www.microway.com	C2, C4
Archie McPhee www.mcphee.com	78	O'Reilly OSCON www.oscon.com	33
ASA Computers, Inc. www.asacomputers.com	53	O'Reilly Velocity conferences.oreilly.com/velocity	77
Cari.net www.cari.net	43	Polywell Computers, Inc. www.polywell.com	78, 79
Codero www.codero.com	11	Saint Arnold Brewing Company www.saintarnold.com	78
Digi-Key Corporation www.digi-key.com	79	Servers Direct www.serversdirect.com	9
Emac, Inc. www.emacinc.com	25	Silicon Mechanics www.siliconmechanics.com	19, 29
Gecad Technologies/Axigen www.axigen.com	79	SouthEast LinuxFest southeastlinuxfest.org	31
Genstor Systems, Inc. www.genstor.com	75	Technologic Systems www.embeddedx86.com	17
Gutsy Geeks www.gutsygeeks.com	47	Techweb Live Events (BlackHat) www.blackhat.com	45
Intel www.intel.com	13	Trusted Computer Solutions www.trustedcs.com/SecurityBlanket	7, 79
iXsystems, Inc. www.ixsystems.com	C3	USENIX Association www.usenix.org/confweek10	49
Logic Supply, Inc. www.logicsupply.com	21	Utilikilts www.utilikilts.com	78

ATTENTION ADVERTISERS

September 2010 Issue #197 Deadlines
Space Close: June 28; **Material Close:** July 6

Theme: Web Development

BONUS DISTRIBUTIONS:
USENIX Security, Utah Open Source Convention

Print: contact **Joseph Krack, +1-713-344-1956 ext. 118, joseph@linuxjournal.com**

On-line: contact **Michael Beasley, +1-713-344-1956 ext. 119, michael@linuxjournal.com**

```
root=/dev/c0d0p0s0
usr=/dev/c0d0p0s2
home=/dev/c0d0p0s1
```

Although this naming system may seem intimidating at first, in practice, it is very simple. It lists the physical controller and disk, followed by the partition and sub-partition, with the first of each item numbered 0.

Naturally, other distinguishing characteristics of MINIX will become clearer as you explore it in more detail. But if you do need help, MINIX supports man pages, just like most UNIX-like systems, and it includes an interesting application called whichman that attempts to find approximate matches to a query. However, you will not find any info pages, despite the fact that MINIX uses utilities provided by the GNU Project. You also can find help on the MINIX Wiki, although it is not always up to date and often suffers from a lack of detail.

Software Selection and Installation

When you install MINIX, the result is a minimal system (a setup that is in keeping with basic security principles). If you want more, you have to install it yourself. Beyond the basic system, MINIX has a small but well-rounded collection of 135 packages, tailored to the needs of the command line. By default, it uses the ash shell, but BASH and zsh are also available. It includes support for several programming languages, including Tcl, Perl, Python and FLTK, and users can choose between vile, vim and nano for text editors.

Some of MINIX's applications, such as Kermit, might seem old-fashioned from a modern GNU/Linux user's perspective. Others will seem thoroughly contemporary, such as SQLite, OpenSSL and wget. Then, there are the usual suspects, such as ImageMagick, tar and zip. You even can unwind with a game of *Nethack* on MINIX. In keeping with MINIX's status as an educational operating system, typing a command without any parameters displays a brief summary of usage.

In MINIX, you won't find desktop applications, such as Firefox or OpenOffice.org. Such programs are many times

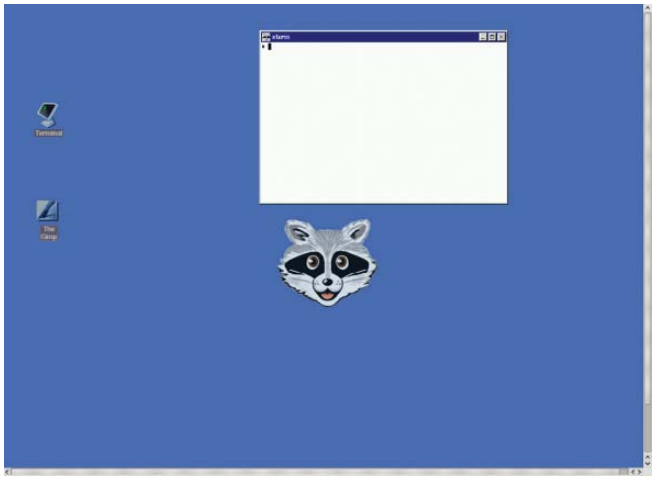


Figure 3. Equinox Desktop Environment (EDE)

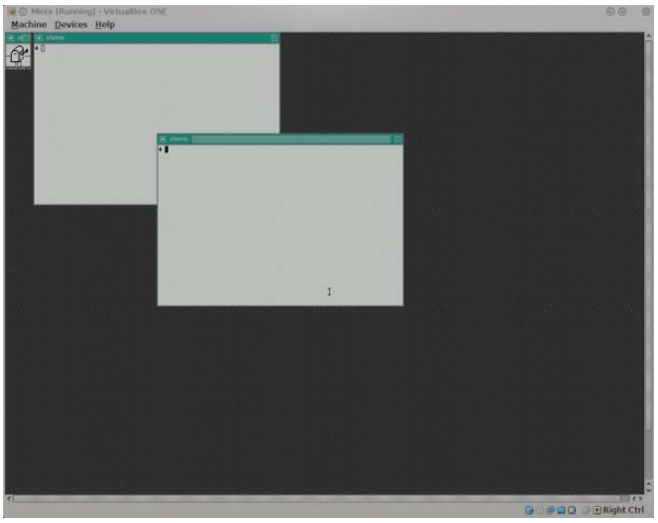


Figure 4. Timeless Windows Manager (TWM)

```
 1 apache-1.3.37   The Apache HTTP Server v1.3.37  (8112 kB)
 2 apr-0.9.12      apache portable runtime library (gcc format)  (336 kB)
 3 apr-util-0.9.12 apache portable runtime utils  (256 kB)
 4 ascii 3.8       Interactive ASCII name and synonym chart  (18 kB)
 5 atk-1.9.0       The ATK library.  (368 kB)
 6 autoconf-2.60   Autoconf set of automatic configuration tools  (1008 kB)
 7 automake-1.10.3 GNU automake  (976 kB)
 8 avra-0.7        Assembler for Atmel AVR microcontrollers  (98 kB)
 9 bash-3.0        GNU bourne-again shell  (752 kB)
10 bc 1.06         Arbitrary precision calculator  (110 kB)
11 bchunk-1.2.0    CD image format conversion from bin/cue to iso/cdr  (38 kB)
12 bcrypt-1.1      Cross platform file encryption utility  (134 kB)
13 binutils-2.16.1 A collection of GNU binary tools  (15024 kB)
14 bison-2.1       Parser generator (requires gnu m4 in m4 package)  (2144 kB)
15 bsd-utils       Port of various base system BSD utilities from OpenBSD.  (09
6 kB)
16 catdoc-0.94.2   view various file types such as ms word in text  (400 kB)
17 CSSC-1.0.1      SCCS Version Control Software Clone  (6224 kB)
18 cvs-1.11.21     Concurrent versioning system  (640 kB)
19 db-4.4.20       Berkeley DB - Open source developer database  (27168 kB)
20 diffutils-2.8.1 GNU diff and friends  (272 kB)
21 dungeon 2.7.1   Text adventure dungeon exploration game  (336 kB)
22 edc 1.1         Equinox Desktop Environment 1.1 (needs EFLTK)  (38701 kB)
23 efltk-2.0.6     EFLTK - Extended Fast Light Toolkit 2.0.6  (7072 kB)
standard-input, 2-24
```

Figure 5. Available Packages in Packman

```
132 zip 2.31        make .zip archives (compressed files into one)  (166 kB)
133 zlib 1.2.3      zlib compression library  (156 kB)
134 zsh 4.2.6       The Z shell  (1140 kB)
Format examples: '3', '3,6', '3-9', '3-9,11-15', 'all'
Package(s) to install (RETURN or q to exit)? 129
Get source(s) too? (y/N) n

Retrieving 129 (X11R6.8.2) from primary location into /usr/tmp/packages ..
Retrieved ok. Installing ..
Setting size of /usr/X11R6/bin/X to 8000000
/usr/X11R6/bin/X: Stack+malloc area changed from 15728640 to 8000000 bytes.
Installed ok.

Showing you a list of packages using more. Press q when
you want to leave the list.
Press RETURN to continue..
```

Figure 6. Installing X with Packman

larger than the whole of MINIX, and including them would go against the project's goals of being suitable for embedded systems. Strangely enough, you will find a package for The GIMP. But the closest you will find to Firefox is Lynx, and the closest to OpenOffice.org is TeX.

For that matter, you will find little attention paid to graphical interfaces in general. The X Window System is available, but the interfaces are few. You can run TWM (Figure 4) for an extremely basic desktop, but with the unaliased text, you are better off at the command prompt. The Equinox Desktop

Environment (Figure 3) is considerably more sophisticated, but unless you're doing something like viewing graphics, running any sort of graphical interface in MINIX is mostly beside the point. Although you could study the X Window System in MINIX, the overwhelming majority of the work you might do in MINIX works just as well from the command line, if not better.

Whatever your choice of extras, they are installed with the command `packman`. Packman opens with a list of the available packages (Figure 5). Dependencies are not resolved automatically in packman, but the list informs you when a package requires another one.

When you know what packages you want to install, press the q key, and enter their numbers at the prompt (Figure 6). When you are finished installing, press the q key twice to exit.

Conclusion

Assuming you have experience with UNIX-like operating systems, you shouldn't need more than a session or two to familiarize yourself with MINIX and its resources.

If your interest is not just casual, your next step might be to read the MINIX Developer's Guide. This concise guide goes all the way from the resources you might need to learn more about programming in MINIX to explaining the MINIX API and packaging format in detail. Then, you might want to see what MINIX projects are already organized on the project's Who Is Working On What page and joining the Google Group for MINIX. Conversely, if you are considering using MINIX for teaching, look at the Teaching with MINIX Web page for resources.

But is MINIX worth this effort? Is it, perhaps, a historical relic whose best days are past? How you answer that is very much a matter of perspective.

On the one hand, MINIX development and teaching are both relatively small worlds, so individuals might expect (all else being equal) to contribute to them more easily and meaningfully than to larger free and open-source projects. MINIX development in particular seems to be at a significant stage as the project tries to redefine its relevance. And, there is something to be said about learning and teaching about UNIX-like systems in a smaller, less-cluttered environment, especially one that is not desktop-oriented.

On the other hand, some might consider MINIX (to be frank) a dead end. Why, they might argue, should anyone put effort into such a small project when working with GNU/Linux or FreeBSD is more relevant today and promises to teach more marketable job skills?

Assuming you have experience with UNIX-like operating systems, you shouldn't need more than a session or two to familiarize yourself with MINIX and its resources.

Either way, MINIX is worth some attention. You may decide not to invest a lot of time in MINIX, but after examining it in any detail, you will probably return to your own free operating system of choice with a better understanding of how it is structured. For all the efforts to refocus MINIX, teaching is very much what it continues to do best.■

Bruce Byfield is a computer journalist who covers free and open-source software. He has been a contributing editor at *Maximum Linux* and Linux.com, and he currently is doing a column and a blog for *Linux Pro Magazine*. His articles appear regularly on such sites as Datamation, LinuxJournal.com and Linux Planet. His article, "11 Tips for Moving to OpenOffice.org" was the cover story for the March 2004 issue of *Linux Journal*.

Resources

The Tanenbaum-Torvalds Debate: **oreilly.com/catalog/opensources/book/appa.html**

Some Notes on the "Who wrote Linux" Kerfluffle, Release 1.5 (Kenneth Brown book proposal): **www.cs.vu.nl/~ast/brown**

MINIX Hardware Requirements: **wiki.minix3.org/en/UsersGuide/HardwareRequirements**

MINIX man pages: **www.minix3.org/manpages**

MINIX Wiki: **wiki.minix3.org/en/FrontPage**

MINIX Software Packages: **www.minix3.org/software**

MINIX Developer's Guide: **wiki.minix3.org/en/DevelopersGuide**

Who Is Working On What: **wiki.minix3.org/en/WhoIsWorkingOnWhat**

Google Group for MINIX: **groups.google.com/group/minix3**

Teaching with MINIX: **minix1.woodhull.com/teaching**

Automating Remote Backups

Most home users don't think about backups till their disks crash. With a little bit of upfront work, you can automate your home backups and sleep easy at night.

MICHAEL J. HAMMEL

Linux users are a diverse group because of the wide swath of choices they have at their fingertips. But, whether they choose Ubuntu, Fedora or Debian, or KDE, GNOME or Xfce, they all have one thing in common: a lot of data. Losing data through hard disk failure or simply by overwriting is something all users must face at some point. Yet, these are not the only reasons to do backups. With a little planning, backups are not nearly as hard as they might seem.

Hard disk prices have dropped to the point where USB storage easily replaces the need for off-line tape storage for the average user. Pushing your data nightly to external USBs, either local or remote, is a fairly inexpensive and simple process that should be part of every user's personal system administration.

In this article, I describe a process for selecting files to back up, introduce the tools you'll need to perform your backups and provide simple scripts for customizing and automating the process. I have used these processes and scripts both at home and at work for a number of years. No special administrative skills are required, although knowledge of SSH will be useful.

Why We Backup

Before proceeding, you should ask yourself the purpose of the backup. There are two reasons to perform a backup. The first is to recover a recent copy of a file due to some catastrophic event. This type of recovery makes use of *full backups*, where only a single copy of each file is maintained in the backup archive. Each file that is copied to the archive replaces the previous version in the archive.

This form of backup is especially useful if you partition your system with a root partition for the distribution of choice (Fedora, Ubuntu and so forth) and a user partition for user data (/home). With this configuration, distribution updates are done with re-installs instead of upgrades. Installing major distributions has become fairly easy and nearly unattended. Re-installing using a separate root partition allows you to wipe clean the old installation without touching user data. All that is required is to merge your administrative file backups—a process made easier with tools like meld (a visual diff tool).

The second reason to perform a backup is to recover a previous version of a file. This type of recovery requires the backup archive to maintain an initial full backup and subsequent incremental changes. Recovery of a particular version of a file requires knowing the time between when the full backup was performed and the date of the version of the file that is desired in order to rebuild the file at that point. Figure 1 shows the full/incremental backup concepts graphically.

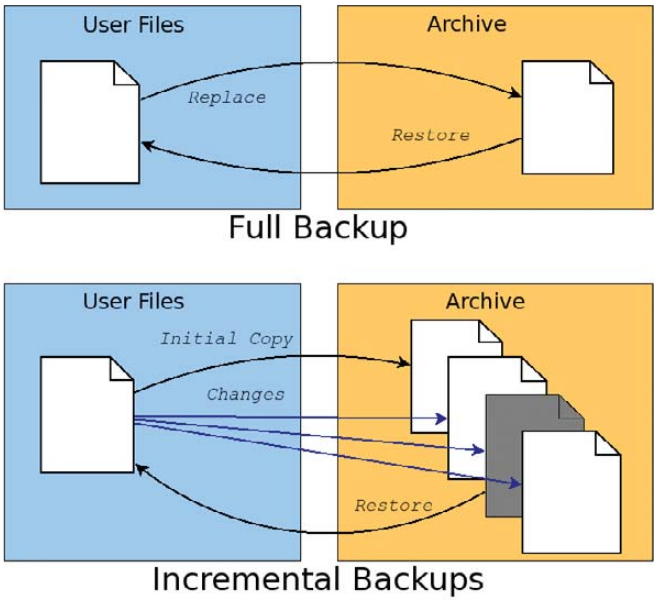


Figure 1. Full backups replace archive contents. Incremental backups extend archives with time-based file changes.

Incremental backups will use up disk space on the archive faster than full backups. Most home users will be more concerned with dealing with catastrophic failure than retrieving previous versions of a file. Because of this, home users will prefer full backups without incremental updates, so this article focuses on handling only full backups. Fortunately, adding support for incremental backups to the provided scripts is not difficult using advanced features of the tools described here.

In either case, commercial environments often keep backups in three locations: locally and two remote sites separated by great distance. This practice avoids the possibility of complete loss of data should catastrophe be widespread. Home users might not go to such lengths, but keeping backups on separate systems, even within your home, is highly recommended.

Tool Primer

The primary tool for performing backups on Linux systems is rsync.

This tool is designed specifically for handling copying of large numbers of files between two systems. It originally was designed as a replacement for rcp and scp, the latter being the file copy tool provided with OpenSSH for doing secure file transfers.

As a replacement for scp, rsync is able to utilize the features provided by OpenSSH to provide secure file transfers. This means a properly installed SSH configuration can be utilized when using rsync. In fact, SSH transfers are used by default using standard URI formats for source or destination files (such as user@host:/path). Alternatively, rsync provides a standalone server that rsync clients can connect to for file transfers. To use the rsync server, use a double colon in the URI instead of a single colon.

SSH (secure shell), is a client/server system for performing operations across a network using encrypted data. This means what you're transferring can't be identified easily. SSH is used to log in securely to remote Linux systems, for example. It also can be used to open a secure channel, called a tunnel, through which remote desktop applications can be run and displayed on the local system.

SSH configuration can be fairly complex, but fortunately, it doesn't have to be. For use with rsync, configure the local and remote machines for the local machine to log in to the remote machine without a password. To do this, on the local machine, change to $HOME/.ssh and generate a public key file:

```
$ cd $HOME/.ssh
$ ssh-keygen -t dsa
```

ssh-keygen will prompt you for various information. For simplicity's sake, press Enter to take the default for each prompt. For higher security, read the ssh-keygen and ssh man pages to learn what those prompts represent.

ssh-keygen generates two files, id_dsa and id_dsa.pub. The latter file must be copied to the remote system under $HOME/.ssh and appended to the file $HOME/.ssh/authorized_keys. In this code, *remoteHost* is the name of the remote computer and *localHost* is the name of the local computer:

```
$ scp id_dsa.pub \
      remoteHost:$HOME/.ssh/id_dsa.pub.localHost
$ ssh remoteHost
$ cd $HOME/.ssh
$ cat id_dsa.pub.localHost >> authorized_keys
```

In this article, I assume a proper SSH configuration with no password required in order to perform the rsync-based backups. These automated backup scripts are intended to be run from cron and require a proper SSH configuration.

Backup UI: Grsync

For users who prefer to use a desktop tool instead of scripts for setting up and performing backups, there is the Grsync tool. This is a GTK+-based tool that provides a nearly complete front end to rsync. It can be used to select a single source and destination and is likely available from Linux distribution repositories.

Although previous versions appear to have had an integrated cron configuration, the current version available with Fedora does not. Also, Grsync does not allow selection of multiple source files or directories nor does it allow setting exclusion lists. Both of these are supported by the rsync command line. Grsync can create a session file that can be called from cron, but it does not include information on how to notify the user of the results of the backup.

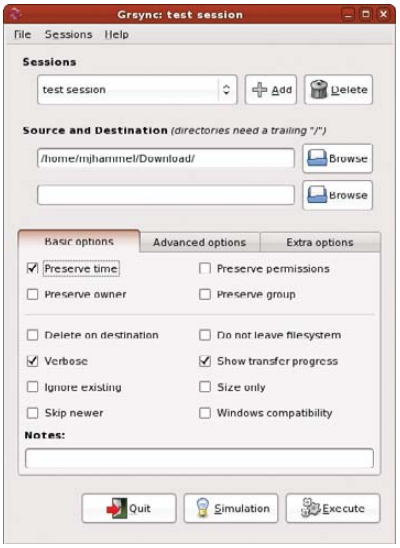


Figure 2. Grsync is a desktop tool for scheduling backups. Although generally useful, it lacks include/exclude options and direct cron management.

Due to the lack of cron integration, missing include and exclude options and no integration of user notification, Grsync is not an ideal backup solution. The scripts described here, along with the addition of ssmtp for simplified SMTP-based notification, are a better solution.

File Selection

With SSH set up and the choice to script backups instead of using a desktop application out of the way, it is time to consider what files to back up. Four sets of files should be considered: system configuration files, database files, users' home directories and Web files.

System configuration files include files such as the password and group files, hosts, exports and resolver files, MySQL and PHP configurations, SSH server configuration and so forth. Backup of various system configuration files is important even if it's not desirable to reuse them directly during a system re-install. The password and group files, for example, shouldn't be copied verbatim to /etc/passwd and /etc/group but rather used as reference to re-create user logins matched to their home directories and existing groups. The entire /etc directory can be backed up, although in practice, only a few of these files need to be re-installed or merged after a distribution re-installation.

Some applications built from source, such as ssmtp, which will be used for notification in the backup scripts, may install to /usr/local or /opt. Those directories can be backed up too, or the applications can be rebuilt after a distribution upgrade.

MySQL database files can be backed up verbatim, but it may be easier to dump the databases to a text file and then reload them after an upgrade. This method should allow for the database to handle version changes cleanly.

User home directories typically contain all user data.

Generally, all files under /home except the /home/lost+found directory should be backed up. This assumes that all user logins are kept on /home. Check your distribution documentation to verify the location of user home directories.

Home users may not use Web servers internally, but there is no reason they shouldn't be. Wikis, blogs, media archives and the like are easy to set up and offer a family a variety of easy-to-use communication systems within the home. Setting up document root directories (using Apache configuration files) under /home makes backing up these files identical to any other user files.

There are also files and directories to avoid when performing backups. The lost+found directory always should be excluded, as should $HOME/.gvfs, which is created for GNOME users after they log in.

Scripting and Notification

All of the backups can be handled by a single script, but because backup needs change often, I find it easier to keep with UNIX tradition and created a set of four small scripts for managing different backup requirements.

The first script is used to run the other scripts and send e-mail notifications of the reports on the backup process. This script is run by root via cron each night:

```
#!/bin/bash

HOST=`hostname`
date=`date`
mailfile="/tmp/$$.bulog"

# Mail Header
echo "To: userid@yourdomain.org"           > $mailfile
echo "From: userid@yourdomain.org"        >> $mailfile
echo "Subject: $HOST: Report for $date"   >> $mailfile
echo " "                                  >> $mailfile
echo "$HOST backup report:"               >> $mailfile
echo "------------------------------"     >> $mailfile

# Run the backup.
$1 >> $mailfile 2>&1

# Send the report.
cat $mailfile | \
    /usr/local/ssmtp/sbin/ssmtp -t \
    -auuserid@yourdomain.org -apyourpassword \
    -amCRAM-MD5
rm $mailfile
```

The first argument to the script is the backup script to run. An enhanced version would verify the command-line option before attempting to run it.

This script uses an external program (ssmtp) for sending backup reports. If you have an alternative tool for sending e-mail from the command line, you can replace ssmtp usage with that tool. Alternatively, you can skip using this front end completely and run the backup scripts directly from cron and/or the command line.

ssmtp

ssmtp is a replacement for Sendmail that is considerably less complex to configure and use. It is not intended to retrieve mail, however. It is intended only for outbound e-mail. It has a small and simple configuration file, and when used as a replacement for Sendmail, it will be used by command-line programs like mail for sending e-mail.

ssmtp is not typically provided by Linux distributions, but the source can be found with a Google search on the Internet. Follow the package directions to build and install under /usr/local. Then, replace sendmail with ssmtp by setting a symbolic link from /usr/sbin/sendmail to the installation location of ssmtp.

```
$ mv /usr/sbin/sendmail /usr/sbin/sendmail.orig
$ ln -s /usr/local/sbin/ssmtp /usr/sbin/sendmail
```

If your distribution supports the alternatives tool, you may prefer to use it instead of the symbolic link to let the system use ssmtp instead of Sendmail. Note that, as a bonus, when the author replaced Sendmail with ssmtp, LogWatch suddenly began sending nightly reports via e-mail, allowing me a view on system activity I never had seen before and which many Linux users probably never have seen before either.

System Configuration File Backups

Backing up system configuration files is handled by a Perl script that verbosely lists the files to be copied to a location on the /home partition. The script is run by root via cron every night to copy the configuration files to a directory in user data space (under /home):

```
#!/usr/bin/perl
$filelist = <<EOF;
/etc/passwd
/etc/group
...  # other config files to backup
EOF

@configfiles = split('\n', $filelist);
for (@configfiles)
{
    if (-e $_) { $files = join(" ", $files, $_); }
    elsif (index($_, "*") >= 0) {
        $files = join(" ", $files, $_);
    }
}
print "Creating archive...\n";
`tar Pczf $ARGV[0]/systemfiles.tar.gz $files`;
```

This brute-force method contains a list of the files to back up, joins them into a single tar command and builds a tar archive of those files on the local system. The script is maintained easily by modifying the list of files and directories. Because the configuration files are copied locally to user data space, and user data space is backed up separately, there is no need for rsync commands here. Instead, the system configuration tar archive is kept with user data and easily referenced when doing restores or

system upgrades. The backup script functions as a full backup, replacing the tar archive with each execution unless a different destination is specified as a command-line argument.

What this script lacks in Perl excellence it makes up for in simplicity of maintenance. Note that the "retail" version of this script ought to include additional error checking for the command-line argument required to specify the location to save the archive file.

Database Backups

Like system configuration files, databases are backed up to user data directories to be included in the user data backups. Databases are of slightly higher importance in day-to-day use, so this script uses a seven-day rotating cycle for database file dumps. This allows restoring backups from up to a week ago without overuse of disk space for the backups. This method is not incremental, however. It is a set of seven full backups of each database.

Like the system configuration file backup script, this script lists the items to back up. The mysqldump command assumes no password for the root user to access the databases. This is highly insecure, but for users behind a firewall, it is likely the easiest way to handle database management:

```
#!/usr/bin/perl -w
use File::Path qw(make_path remove_tree);
my $BUDIR1="/home/httpd/db";
my ($sec,$min,$hour,$mday,$mon,$year,
    $wday,$yday,$isdst) = localtime time;
$year += 1900;
$mon += 1;
if ($mon < 10 )  { $mon  = "0".$mon; }
if ($mday < 10 ) { $mday = "0".$mday; }
$TODAY = $wday;

@dbname = (
    "mysql",
    "wordpress",
);

make_path ("$BUDIR1/$year");
foreach $db (@dbname) {
    $cmd = "mysqldump -B -u root $db " .
           "-r $BUDIR1/$year/$TODAY-$db.sql";
    system("$cmd");
}

print ("Database Backups for " .
       $year . "/" . $mon . "/" .
       $mday . "\n");
print ("--------------------------------\n");
open(PD, "ls -l $BUDIR1/$year/$TODAY-*.sql |" );
@lines = <PD>;
close(PD);
$output = join("\n", @lines);
print ($output);
```

Unlike the configuration file backup script, this script prints out the list of files that have been created. This provides a quick, visual feedback in the e-mailed report that the backups produced something meaningful.

User Data Backups

The system configuration backup script and the database backup script are run first to generate backups to user data space. Once complete, all data is ready to be backed up to the remote system with an rsync-based script:

```
#!/bin/bash
function checkRC
{
    rc=$1
    name=$2
    if [ $rc != 0 ]
    then
        echo "== $name failed with rsync rc=$rc =="
    fi
}
LOGIN=root@feynman
BRAHE=$LOGIN:/media/BackupDrive/feynman
if [ "$1" != "" ]
then
    BRAHE=$1
fi
```

The script includes a shell function to test rsync's return code and print an error message on failure. The front-end script redirects output from this script to a file, so error messages show up in the e-mailed backup report.

The default destination for the backup is configured at the start of the script. The first command-line argument can be used to override the default:

```
DIR1="/home/httpd"
DIR2="/home/mjhammel"
EXCL2=--exclude-from=/home/mjhammel/.rsync/local
```

The user data backup script is focused on directories. Unlike the other backup scripts, the list of items to back up are hard-coded in separate variables. Again, this is a brute-force method used for simplicity, because each directory to back up may have one or more sets of include and exclude arguments. Associative arrays could be used instead of the set of variables in a more generalized version of this script.

Notice that this configuration calls out individual directories under /home instead of backing up all of /home. The script from which this was pulled is used on a machine with development directories under /home that do not need to be backed up. Specifying /home and using an exclusion file is an alternative way of doing the same thing:

```
DATE=`date`
echo "== Backing up `uname -n` to $BRAHE."
echo "== Started @ $DATE "
echo "== Directory: $DIR1"
```

```
rsync -aq --safe-links $DIR1 $BRAHE
checkRC $? "$DIR1"
```

The first directory is backed up to the remote system. The -a option tells rsync to operate in archive mode, where rsync will do the following:

- Recursively traverse the specified directory tree.
- Copy symlinks as symlinks and not the files they point to.
- Preserve owner, groups, permissions and modification times.
- Preserve special files, such as device files.

The safe-links option tells rsync to ignore symbolic links that point to files outside the current directory tree. This way, restoration from the archive won't include symbolic links that may point to locations that no longer exist. The -q option tells rsync to run with as few non-error messages as possible:

```
echo "== Directory: $DIR2"
rsync -aq --safe-links $EXCL2 $DIR2 $BRAHE
checkRC $? "$DIR2"
DATE=`date`
echo "Backups complete @ $DATE"
```

The second directory tree is backed up using an exclusion list. This list is a file that specifies the files and directories within the current directory tree to be ignored by rsync. Entries in this file prefixed with a dash are excluded from the set of files and directories rsync will process. The three asterisks match anything below the specified directories:

```
- /mjhammel/.gvfs/***
- /mjhammel/Videos/***
- /mjhammel/brahe/***
- /mjhammel/iso/***
```

This example shows that no files under the Videos and iso directories will be included in the backup. It would be a poor use of disk space to back up files that exist in your home directory but that also can be retrieved from the Internet.

The brahe reference is a mountpoint for the home directory of an identical user ID on a remote system. This allows access to files under a login on another system simply by changing into the remote system's local mountpoint. But, there is no reason to back up those remote files on the local system, as that remote system has its own backup scripts configured.

The full version of this script includes an SSH-based verification that the remote system has the required external USB drive mounted and it is available for use. This allows the script to recognize that the remote system is misbehaving before wasting time trying to run a backup that would fail anyway.

Automation via Cron

The order in which these scripts is run is important. The system configuration file backup script and the database backup script can run in parallel but must complete before the user data backup script is run:

```
30 0 * * * /path/to/backup-db.pl
30 1 * * * /path/to/backup-configfiles.sh \
           /path/to/save/dir 2>&1 > /dev/null
30 2 * * * /path/to/backup-frontend.sh \
           /path/to/backup-data.sh
```

To pass arguments to backup-data.sh, enclose the entire command in double quotes:

```
30 2 * * * /path/to/backup-frontend.sh \
          "/path/to/backup-data.sh
root@copernicus:/backups"
```

Each morning, the backup report is available for each machine that runs these scripts and can be reviewed to make sure the backups completed successfully. In practice, the most common problems encountered are related to unmounted or non-functioning drives, or to network outages that occur before or during the backup process.

Summary

In preparing a personal backup strategy, it is important to identify the purpose of the backup, establish a set of processes that prepares files for backup and performs backups to remote systems. It is also important that automation of these processes provide feedback, so users can have at least a starting point of understanding why backups are failing and when that may have occurred.

The methods shown here are somewhat simple and certainly not ideal for every user. The scripts probably are not bug-free and also have room for improvement. They are intended only as a starting point for building personal backup plans. I welcome feedback on any improvements you make to these scripts.■

Michael J. Hammel is a Principal Software Engineer for Colorado Engineering, Inc. (CEI), in Colorado Springs, Colorado, with more than 20 years of software development and management experience. He has written more than 100 articles for numerous on-line and print magazines and is the author of three books on The GIMP, the premier open-source graphics editing package.

Resources

Backup Scripts for This Article: **ftp.linuxjournal.com/pub/lj/listings/issue194/10679.tgz**

ssmtp: **www.graphics-muse.org/source/ssmtp_2.61.orig.tar.gz**

rsync: **samba.anu.edu.au/rsync**

OpenSSH: **www.openssh.com**

meld: **meld.sourceforge.net**

TECH TIPS

Formatting Information about RPMs

RPM packages contain a number of potentially useful information fields, and you can see many of them by using the -q and -i options with a package name:

```
$ rpm -q -i kernel-desktop
Name        : kernel-desktop          Relocations: (not relocatable)
Version     : 2.6.31.12               Vendor: openSUSE
Release     : 0.1.1                   Build Date: Thu 28 Jan 2010 ...
Install Date: Sat 20 Feb 2010 ...     Build Host: build35
Group       : System/Kernel           Source RPM: kernel-desktop-...
Size        : 122840714               License: GPLv2
Signature   : RSA/8, Thu 28 Jan 2010 09:16:41 AM MST, ...
Packager    : http://bugs.opensuse.org
URL         : http://www.kernel.org/
Summary     : Kernel optimized for the desktop
Description :
This kernel is optimized for the desktop. ...

Source Timestamp: 2010-01-27 08:20:11 +0100
GIT Revision: bb438b6d99aaffb7aade40764bab1810cc21b01b
GIT Branch: openSUSE-11.2
Distribution: openSUSE 11.2
```

These fields and others can be output individually by using the `--qf` or `--queryformat` options for rpm. Not everything above is directly an option, but many are, such as `NAME` and `VERSION`, so that you can do something like:

```
$ rpm -q --qf "Installed kernel is v%{VERSION}\n" kernel-desktop
Installed kernel is v2.6.31.12
```

The format string is similar to a printf format string, except the type specifier is replaced by the tag to output inside braces. To see a table of all installed packages and their versions, you could do this:

```
$ rpm -q -a --qf '%-30{NAME}  Version %{VERSION}\n'|head
gpg-pubkey                    Version a1912208
yast2-trans-stats             Version 2.15.0
yast2-country-data            Version 2.18.20
libpciaccess0                 Version 7.4
libpth20                      Version 2.0.7
libpcre0                      Version 7.9.0
netcat                        Version 1.10
libusb-1_0-0                  Version 1.0.2
libbz2-1                      Version 1.0.5
libgmp3                       Version 4.3.1
...
```

You also can change the formatting for an item by appending a qualifier. For example, in the info listing above, there's a field called `Install Date`. The actual rpm tag is `INSTALLTIME`, and if you use that directly you get a less-than-useful value:

```
$ rpm -q --qf "Kernel installed %{INSTALLTIME}\n" kernel-desktop
Kernel installed 1266703208
```

For something more useful, append `:date` to the tag:

```
$ rpm -q --qf "Kernel installed %{INSTALLTIME:date}\n" kernel-desktop
Kernel installed Sat 20 Feb 2010 03:00:08 PM MST
```

—MANTRA UNIX

Connect to Your Google Calendar from the Command Line with gcalcli

gcalcli is an easy-to-use command-line tool to interact with your Google Calendar. gcalcli probably is not installed by default on your system, but it may be available via your package manager. If it's not available, get the source at **code.google.com/p/gcalcli**.

Once installed, you can view your current calendar week by typing:

```
$ gcalcli --user USERNAME --pw PASSWORD calw
```

To view the current month of calendar, the command is very similar:

```
$ gcalcli --user USERNAME --pw PASSWORD calm
```

Adding an item to your calendar is easy as well with the `quick` option:

```
$ gcalcli --user USERNAME --pw PASSWORD quick \
              'Write another Tech Tip for LJ tomorrow 5pm'
```

gcalcli has many other options, such as choosing your own color scheme and agenda view. See the man page for more options.

—KRISTOFER OCCHIPINTI (AKA METALX1000)

Get Rid of the Annoying "x is an executable text file" Message Box

When using GNOME, if you open a text file that has the execute bit set, you get an annoying message box each time saying "Do you want to run x.txt or display its content? x.txt is an executable text file."

To get rid of this annoying message box, in Nautilus, go to Edit→Preferences→Behaviour, and check the radio button next to view executable text files when they are opened. The next time, the file will open directly, and no such message box will irk you.

—MANOJ GUMBER

Send a tech tip to techtips@linuxjournal.com, and if we publish it in the magazine, we'll send you a free T-shirt.

KYLE RANKIN

BILL CHILDERS

Mobile Phones

Kyle and Bill have argued about everything from cloud computing to AJAX. This month, join the guys as they argue about something much more personal: mobile phone preferences. Where do you stand?

BILL: So Kyle, I hear you've got a shiny new wireless phone. What'd you get?

KYLE: Well, after reviewing a Nokia N900 for a few months, when it was time to give it back, I decided to buy one of my own [see Kyle's review of the Nokia N900 in the May 2010 issue of *LJ*].

BILL: Wow, you picked that over a BlackBerry, Droid or iPhone?

KYLE: Yeah. Honestly, I was looking for more of a portable Linux computer and less of a smartphone. Because I wanted Linux, the iPhone and Blackberry were out of the equation, so that left me with a choice between the Droid and the N900. How many phones are you carrying around these days?

BILL: Just two—one personal and one work-assigned. I try to keep a good work/home separation these days. You're running two phones as well, from what I remember.

KYLE: Between the Droid and the N900, the hardware was mostly the same (same processor, hardware keyboard and so on), so it came down to the OS. In my opinion, Maemo was just more open and hackable out of the box than Android. Plus, all the apps for Android are written in a custom version of Java. Also, on Android, if you really want to own the device, you have to run unauthorized firmware that relies on exploits just to get root. On Maemo, root is easy to get out of the box without voiding any warranties or getting any C&D letters in the mail.

BILL: Yeah, well, those pesky cellular carriers don't like modified devices mucking about on their networks.

In my experience, Android isn't quite like the Linux we use on our laptops. Android is far more integrated and streamlined. For instance, there's no X server. And thanks to your—I'll use the word "distaste"—of all things Java, you decided to go with the Maemo-powered N900. How do you like it so far?

KYLE: I have to admit, I've been pretty pleased with it so far. I think Maemo is about as close to a regular Linux distribution that you'll get on a portable device that still has lots of the features of some of the shinier smartphones. Speaking of shiny, last I heard you were still on the iPhone bandwagon.

BILL: Oooh, shiny....

KYLE: So, why no Android or Maemo device in your pocket?

BILL: Practicality. I got my iPhone 3GS last year, and the only Android device then was the T-Mobile G1, which is on the wrong network, and there was no Maemo device at the time. Like all things Apple, the experience hasn't been all that bad.

It's like driving a BMW. You can't open the hood and change the oil because only the dealer can do that, but you can cruise down the road at 80 MPH, snubbing your nose at the folks who don't have the nice ride and air conditioning you have.

KYLE: Thanks for the car analogy, by the way, you know how I love those. Since you did bring up the car analogy though, I thought we Linux users didn't want our hoods welded shut? Since you got it, there have been a few different devices to come out with similar hardware but with either Android or Maemo, so why not switch?

BILL: Well, for one, I'm not made of money. $300 for the iPhone last year, and then another $500 for an unlocked Maemo device is just a little too much for my CFO at home to handle. And, you're welcome on the analogy. I know how much you love those. Although the hood may be welded shut, you *can* pop the hood if you know how (via jailbreak). And yes, my phone is jailbroken.

KYLE: See, that's the deal-breaker for me both on the iPhone and on Android devices. I don't think I should have to jailbreak anything to run what I want on it. If you truly own the device, you should be able to install your own software.

BILL: Another reason why I've stuck with the iPhone to date is that we have a stable of iDevices around here. I swear, the things have procreated overnight. Kelly's also running an iPhone, my daughter's got an iPod Touch, and my son has a jailbroken first-generation iPhone without a SIM. All apps we buy can

land on all the devices because they are associated with the same iTunes account. I know, the next thing you're going to bring up is "walled garden"! Although I get the whole freedom argument, sometimes it's nice to have things that "just work".

KYLE: And, every app is now tied to those devices, so even if you did want to switch to a different OS, you'd have to face throwing that money you spent away.

BILL: That's true, but things just work. That happens with any installed base of commercial software, by the way.

KYLE: I mean, it's a nice way to make sure you stick with a particular vendor, but again, it seems to fly in the face of what we stand for as longtime Linux users. It sounds like if it just worked, you wouldn't have to jailbreak your device, now would you?

BILL: Hey, you're making a moral argument out of this. This is a practical thing. After a day of hacking on servers and toiling in the data center, the last thing I want to do is mess around on my daughter's handheld device.

KYLE: Every geek I see with an iPhone seems to have a jailbroken one, so basically to get to this magical "just works" state, people have to take their expensive portable computer out of warranty. Anyway, nothing says I have to hack my N900. You certainly can get plenty of use out of everything it offers by default. It just has a whole other world of options open to you if you do want to tinker, just like any other regular Linux install.

BILL: To bring up another car analogy, my dad has been a mechanic forever. Yet my mom's car, and his truck, are relatively new and covered by warranty. Why? Because although he can build a kick-butt vehicle out of junk, he simply doesn't *want* to. He can just hand the keys to someone and say "fix it". He has the option to open the hood, just like I do with my jailbreak, but he doesn't have to.

KYLE: Not another car analogy.

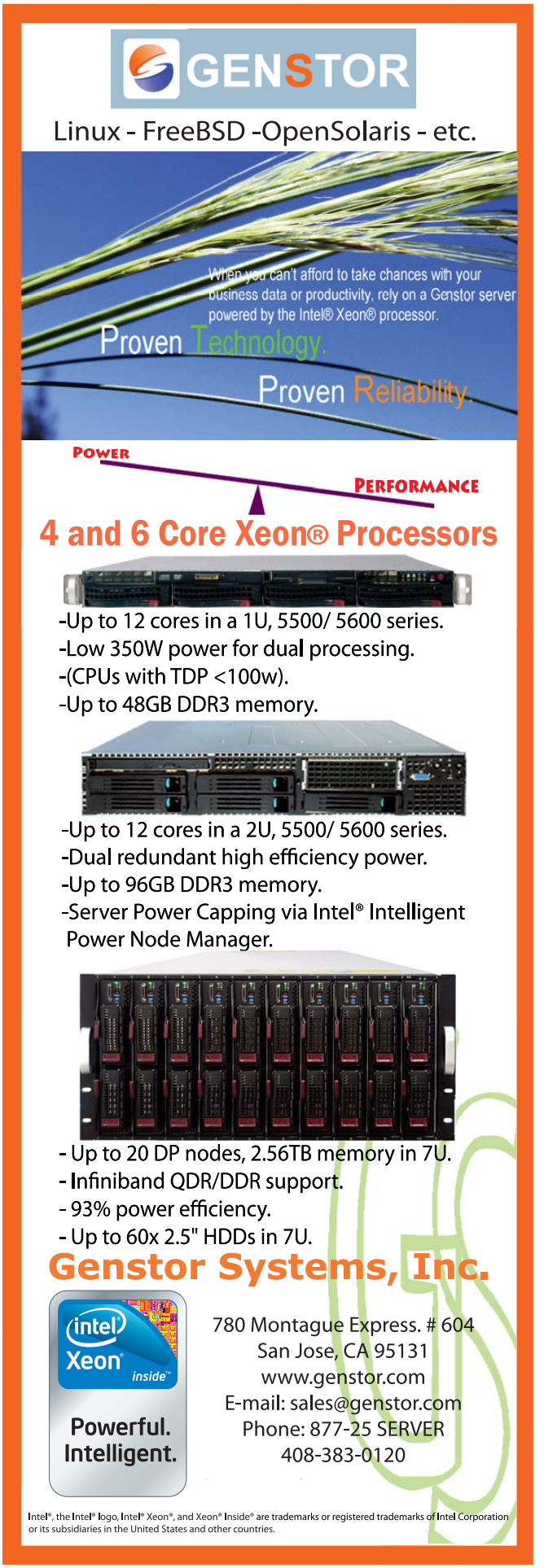
BILL: Yes, another car analogy—cause you know I'm all about that.

KYLE: You don't have to open the hood with the N900 either. There's a whole set of default applications, plus many more are available with the standard application manager.

BILL: I have an N800 Maemo device, and I had to tinker a lot with it to get it where I wanted.

Do you get root right out of the box? I don't think you do. You have to install gainroot, I believe.

KYLE: All you have to do is install one extra program called rootsh, and root is yours, and honestly, I think that is just so you can say you accept the responsibilities of root. See, with your N800, I think you tinkered with it because you knew you could tweak it. With an iPhone, basically it either does something you want or you are out of luck.



BILL: Yeah, I wanted to tweak with the N800 more, I'll grant you that. I even wrote an article on it. Although the N800 and N900 share an ancestry, the use case is different. One's a tablet PC companion, the other is a phone. But jailbreaking isn't much harder than installing your rootsh program. I ran one executable on the computer, and it was done. Regarding the iPhone, I've gotten it to do the tasks I've needed. I've clearly not been out of luck with it.

KYLE: Yet I bet you didn't get that jailbreaking app from the app store did you?

BILL: No, of course not. It was merely a google away.

KYLE: It's like saying, "Sure, I can get any cable channel I want now that I got this descrambler."

BILL: Well, yeah, so what? I can get any cable channel I want. But that's not the point of this column, is it? We're not here to discuss the morality of a device, but its practicality.

KYLE: Really the argument between the iPhone and environments like Maemo is no different from the argument about OS X versus Linux.

BILL: I'll agree there. Actually, they are precisely the same, as iPhoneOS uses the OS X kernel, and Maemo uses Linux. That's the first analogy you've used in this article that has legs.

KYLE: It comes down to whether you are willing to sacrifice the freedoms you are used to in Linux to have something that allegedly "just works", or if it doesn't, you can buy apps until it does.

BILL: I'll bet you I've spent less on my iPhone + apps than you did on your N900.

KYLE: Possibly. After all, I bought the N900 shortly after it came out at an unsubsidized and *unlocked* price. Just like on OS X, you can generally buy your way out of your problems.

BILL: Sometimes, that's the most efficient use of resources. My time is worth something.

KYLE: But then, you seem to be fine with any and all locks, so I'm sure you are fine being locked into a phone contract as well.

BILL: I don't like most locks. I typically break them. I don't mind a phone contract though.

KYLE: See, my time is worth something, but so is my freedom. The beauty of open-source software is that most of the time, I'm not the only one who wants a feature. Because the platform is open, most of the time someone else adds a cool program or feature for me. Even if others don't, at least I have the option if I want it. Plus, on the N900, I can code in C, C++, Python or even bash and use either GTK or Qt if I want, and I don't have to get Apple's blessing for others to use and help improve my program.

BILL: You're trying to pitch me on running Linux, man. I am on the editorial staff here. I get it. I choose, *today*, for my wireless device not to run Linux. I'll admit, that irks me a bit. But it's not enough to cause me to dump the installed base and change my work paradigm...again.

KYLE: I'm just saying the same principles apply whether your computer is desktop-sized or fits in your pocket. In fact, as more and more people use pocket-sized computers, these issues are going to become more important, not less.

BILL: Remember, I moved to the iPhone last year. I really don't feel up to changing things drastically again. It's about practicality for me.

KYLE: The bottom line for me is that I want my portable computer to give me the same freedoms I'm used to on my desktop or laptop. Right now, for me, the only platform that seems to get close is Maemo. I think the open-source model works, and I want it enjoy those advantages no matter what device I use.

BILL: That's cool, and that's your priority. Mine is to have the device's tech just get the heck out of my way so I can do what I need to do. At the end of the day, I'm tired, and I want to get stuff done so I can get home to my kids.

I won't lie and say I've not thought about getting an N900. I have. They look way cool, but playing with a device isn't my top priority right now.

KYLE: Like always, I think we'll have to agree to disagree on this one.

BILL: You think we'll have to? I know we will. But that's okay, we've done that before too.■

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.

O'REILLY

Velocity

Web Performance and Operations Conference

June 22–24, 2010
Santa Clara, CA

Urs Hölzle | Google

Tim O'Reilly | O'Reilly Media

Phillip Dixon | Shopzilla

John Adams | Twitter

Training for the heroes of your IT infrastructure.

The best engineers and developers understand web performance and operations in relation to the whole. Only Velocity, the Web Operations and Performance conference from O'Reilly Media, gives attendees access to the speakers, in-depth technical content, and industry contacts most critical to your company's IT infrastructure and bottom line.

O'Reilly Velocity Conference. Real training for real web ops and performance professionals.

Velocity Tracks and Topics

Three dedicated content tracks featuring topics including:

- Cloud Computing
- Metrics Metrics
- Mobile Web Performance
- Multiple Data Centers
- Configuration Management
- Scalable Video & Social Gaming
- Web Ops & Performance Culture and Management

©2010 O'Reilly Media, Inc. O'Reilly logo is a registered trademark of O'Reilly Media, Inc.
All other trademarks are the property of their respective owners. 10253.1

Save 15% with discount code vel10ljr at **conferences.oreilly.com/velocity**

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

Silent Eco Green PC

The Best Terminal PC

Intel® / AMD® x86 Processor
Energy efficient, Quiet and Low Voltage Platform. starts at $199

LD-001

5048A

5U-48Bay 96TB Storage Server

4U24A

4U-24Bay 48TB
RAID-6, NAS/iSCSI/SAN Storage
Mix SAS / SATA, 4 Giga / 10Gbit LAN

2012A

2U-12Bay 24TB
RAID-6, NAS/iSCSI/SAN Storage
Mix SAS / SATA, 4 GigaLAN

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 Computers, Inc 1461 San Mateo Ave. South San Francisco, CA 94080 650.583.7222 Fax: 650.583.1974

NVIDIA, ION, GeForce and combinations thereof are trademarks of NVIDIA Corporation. Other names are for informational purposes only and may be trademarks of their respective owners.

POLYWELL

INNOVATION ON THE GO
ORDER YOUR BEAGLE BOARD FROM DIGIKEY.COM

AVAILABLE EXCLUSIVELY AT DIGI-KEY
beagleboard

only $149.00

LOW-COST, NO FAN, SINGLE-BOARD COMPUTER

Digi-Key CORPORATION

www.digikey.com

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.

SECURITY BLANKET BY TCS ™

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

NVIDIA® ION™

The World's Small, Greenest, Fanless PC with Blu-Ray Ready

ITX-40A with NVDIA® ION™ Graphics $199 Barebone system
4GB RAM, 1.6GHz Intel® 4W Processor $599 with Blu-ray, 500G HD

ITX-10A 1.4" ThinPC

ITX-20A with SlimDVD

ITX-30A with PCI Riser

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

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 ■ 5-Year Warranty, Industry's Longest ■ First Class Customer Service

Polywell Computers, Inc 1461 San Mateo Ave. South San Francisco, CA 94080 650.583.7222 Fax: 650.583.1974

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.

POLYWELL

EOF

Commons Interests

Were the Winter Olympics a win for Creative Commons?

DOC SEARLS

On a cold February morning in 2009, I noticed beautiful ice patterns had formed overnight inside our apartment's storm windows. Some looked like corners of snowflakes. Some looked like trees. Others looked like feathers. Naturally, I shot pictures of them. Later I put the photo set up on Flickr, tagged the images generously and gave them all a Creative Commons license meant to encourage their enjoyment and re-use. That license happened to be Attribution-Share Alike 2.0 Generic.

The decision to use that license was not a highly considered one. It's a default I chose back when I started uploading photos in 2005. On the whole, this has worked very well. For example, as of today, I have more than 34,000 photos on Flickr, more than 130 of which appear

in Wikimedia Commons, and most of those also show up in Wikipedia. If you go to the Wikipedia pages for Boreray Island, San Gorgonio Mountain, Sarah Lacy or dozens of other topics, you'll find them illustrated by photos of mine, through no additional effort of my own.

I see this as nature taking its course. I am as generous with my photos as trees are with leaves in autumn, and I wish to exercise the same level of control over how they are used. I create them for the commons. Alas, while Creative Commons does offer public domain tools (**http://creativecommons.org/publicdomain**), Flickr doesn't leverage those. So instead I go with one that I hope will encourage re-use going forward, as well as credit back to myself—and to other creators, if any are involved.

Not that I mind making money. Over the years since I started posting on Flickr, about $400 has flowed my way, all in the form of voluntary payments for one use or another. Still, that's not my purpose in putting the pictures up there. My purpose is making them useful. To anybody.

In November 2009 one of those anybodies turned out to be Mark Levy, VP and Creative Director for NBC Sports, writing to say the network would like to use some of my winter ice images in graphic backgrounds for the upcoming Winter Olympics in Vancouver. Since text already would be running over those backgrounds, he asked permission to waive the license details and handle attribution by listing me in the credits as a member of NBC's design team. I said that was fine and didn't give it more thought until the Olympics started running. To my surprise, my ice photos served as framing elements for all kinds of stuff: talking heads in studios, features about athletes, titles of events and settings, and text running below the action on ski slopes and toboggan runs. It was not only fun to watch, but also to feel a sense of participation in a good cause that transcended the commercial interests involved. In other words, I felt honored, not exploited.

Some concern was raised, however, close to home—at Harvard's Berkman Center, where I have been a fellow for the last several years. Creative Commons was born at Berkman, when Lawrence Lessig was there around the turn of the millennium. One of the other fellows at Berkman, Herkko Hietanen (who wrote his doctoral thesis on Creative Commons) saw a potentially interesting problem with NBC's use of the photos. In his blog at MIT's Communications Futures Program, Herkko wrote, "...there is a legal side to the story that could have wreaked havoc. What NBC's designers may have missed was that the license Doc Searls used did not only require credit but also that the adaptations made from Searls' photos share the same license terms." Later he added, "The exact amount of the material that would be affected with the ShareAlike term is unclear....To make the matter even more complicated, NBC does not own and can't license out many of the copyrightable elements that are shown on the screen next to the background graphics. There is no doubt that NBC never wanted its crown jewels, the Olympics, to fall to any royalty-free licensing scheme." Herkko also noted that NBC and I were both satisfied with our agreement and added this response from Creative Commons VP Mike Linksvayer: "NBC's extensive use of Searls' photos, and Searls' happiness for that use, demonstrates the power of Creative Commons licenses as a means to signal openness to collaboration, even if the resulting collaboration does not occur under the terms of the license originally offered."

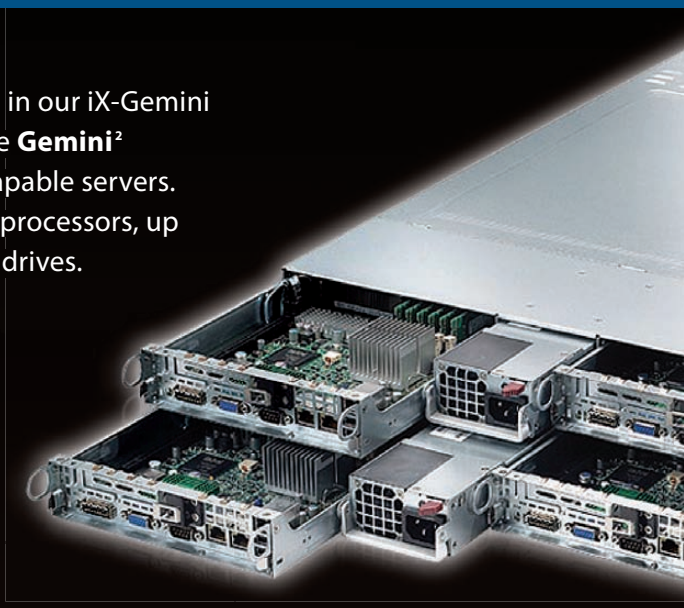
But that didn't sit well with everybody either. On my blog, one commenter wrote, "Wow. Nice of you to give a corporation something worth a few thousand dollars for free, without even having them abide by the share-alike clause of copyleft!"

I still don't know what I should have done differently here. I believe in cultivating a culture of sharing, mixing and remixing. I also want to help Creative Commons push forward its pioneering work in copyright and reform. To be safe for now, I'm moving my photos on Flickr to a simple attribution license. But I'm also open to suggestions—for all of us.■

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.

Gemini²: The Fantastic Four

IXsystems is proud to introduce the latest offering in our iX-Gemini line, the **Gemini²**. Cleverly disguised as any other 2U server, the **Gemini²** secretly houses 4 highly efficient, extremely powerful RAID 5 capable servers. Each node supports the latest Intel® Xeon® 5600 or 5500 series processors, up to 192GB of DDR3 memory, and three 3.5" hot-swappable hard drives.



This system architecture achieves breakthrough x86 server performance-per-watt (375 GFLOPS/kW) to further satisfy the ever-increasing demands for efficiency, density and low-TCO of today's high performance computing (HPC) clusters and data centers. For more information and pricing, please visit our website at http://www.iXsystems.com/gemini2.

Features

Four hot-pluggable systems (nodes) in a 2U form factor

Each node supports the following:

- Dual 64-Bit Socket 1366 Six-Core, Quad-Core, or Dual-Core, Intel® Xeon® Processor 5600/5500 Series
- 3 x 3.5" SAS/SATA Hot-swappable Drive Bays
- Intel® 5520 Chipset with QuickPath Interconnect (QPI)
- Up to 192GB DDR3 1333/1066/800 SDRAM ECC Registered Memory
- 1 (x16) PCI-E (Low Profile)
- Matrox G200eW 8 MB DDR2 Memory Video
- Integrated Remote Management - IPMI 2.0 + IP-KVM with dedicated LAN
- All four nodes share a Redundant 1200W High-efficiency Power Supply (Gold Level 92%+ power efficiency)

800-820-BSDi
http://www.iXsystems.com
Enterprise Servers for Open Source

Intel, the Intel logo, and Xeon Inside are trademarks or registered trademarks of Intel Corporation in the U.S. and other countries.

Powerful.
Intelligent.

More TFLOPS, Fewer WATTS

Microway delivers the fastest and greenest floating point throughput in history

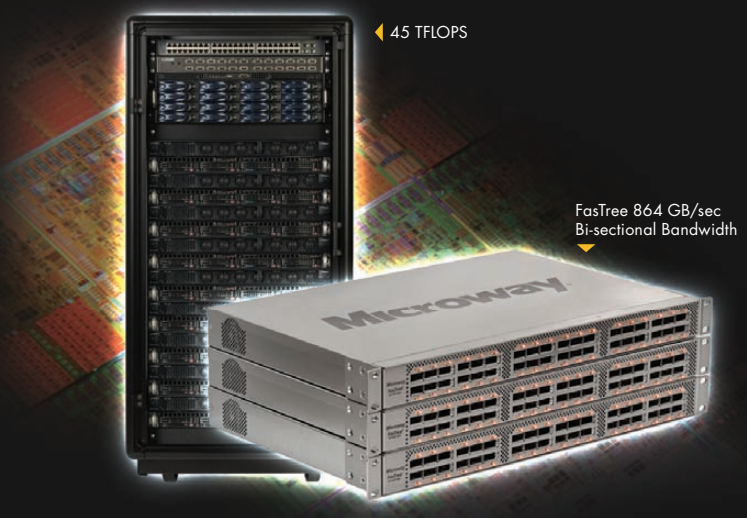
Enhanced GPU Computing with Tesla Fermi

- 480 Core NVIDIA® Tesla™ Fermi GPUs deliver 1.2 TFLOP single precision & 600 GFLOP double precision performance!
- New Tesla C2050 adds 3GB ECC protected memory
- New Tesla C2070 adds 6GB ECC protected memory
- Tesla Pre-Configured Clusters with S2070 4 GPU servers
- WhisperStation - PSC with up to 4 Fermi GPUs
- OctoPuter™ with up to 8 Fermi GPUs and 144GB memory

New Processors

- 12 Core AMD Opterons with quad channel DDR3 memory
- 8 Core Intel Xeons with quad channel DDR3 memory
- Superior bandwidth with faster, wider CPU memory busses
- Increased efficiency for memory-bound floating point algorithms

Configure your next Cluster today!
www.microway.com/quickquote
508-746-7341



FasTree™ QDR InfiniBand Switches and HCAs

- 36 Port, 40 Gb/s, Low Cost Fabrics
- Compact, Scalable, Modular Architecture
- Ideal for Building Expandable Clusters and Fabrics
- MPI Link-Checker™ and InfiniScope™ Network Diagnostics

Achieve the Optimal Fabric Design for your Specific MPI Application with ProSim™ Fabric Simulator

Now you can observe the real time communication coherency of your algorithms. Use this information to evaluate whether your codes have the potential to suffer from congestion. Feeding observed data into our IB fabric queuing-theory simulator lets you examine latency and bi-sectional bandwidth tradeoffs in fabric topologies.

NVIDIA Tesla CUDA | Intel Channel Partner Premier 2010 | AMD Fusion Partner Premier | GSA

GSA Schedule Contract Number: GS-35F-0431N

Microway®
Technology you can count on℠