AN SSC PUBLICATION

SPEED UP CRYPTO WITH HARDWARE AES

LINUX JOURNAL™

Since 1994: The Original Magazine of the Linux Community • **MAY 2005**

Large-scale data center management:
Cyclades AlterPath Manager

Linux for PROFESSIONAL PERFORMERS

Posters, ads, Web and more: publicize yourself with open-source tools

120 GIGABIT NETWORKING WITH INFINIBAND

Get fast Linux connections with zero-copy networking and remote DMA

PLUS:

Adding a front end to GCC 4.0

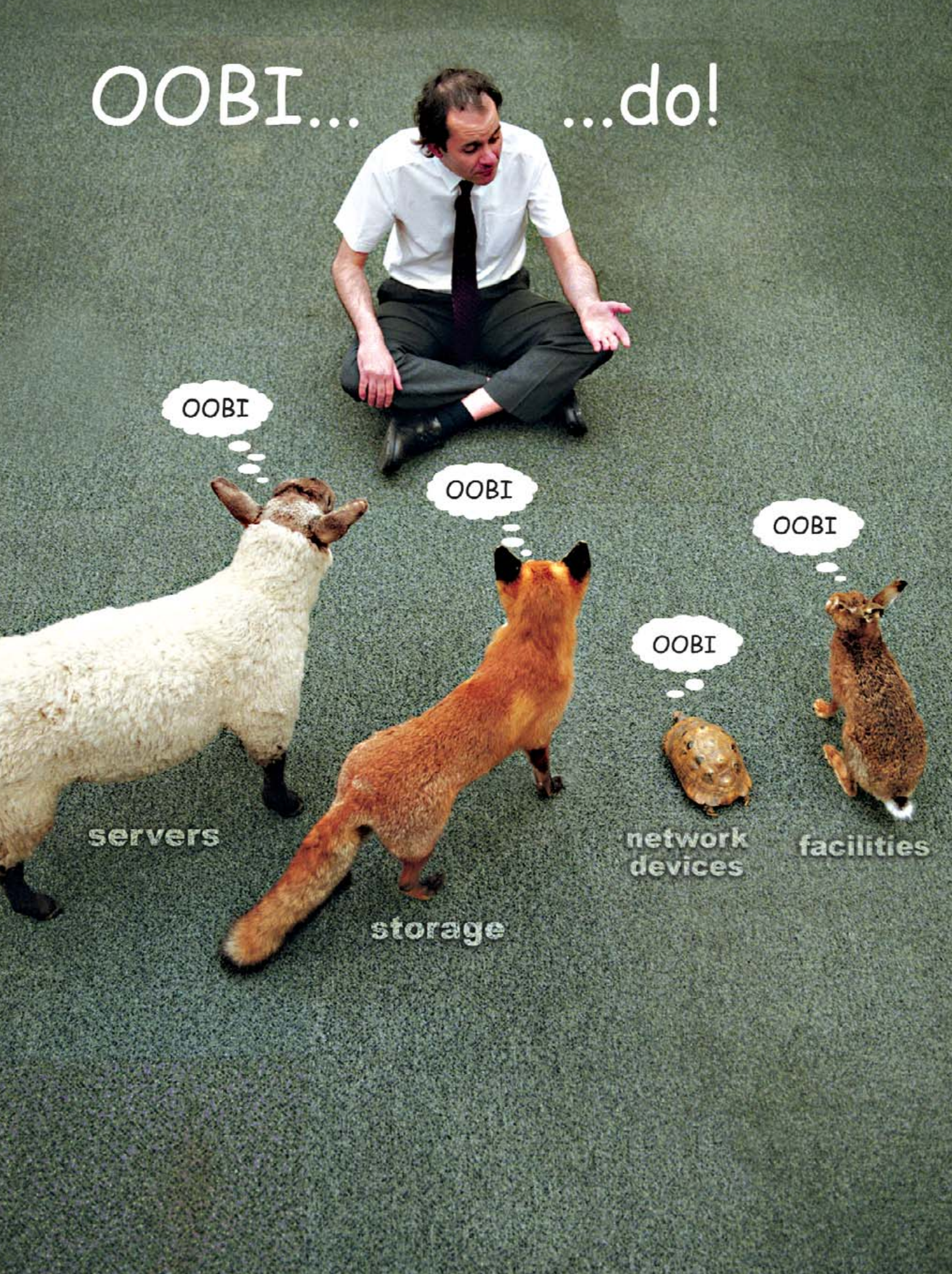
Kernel debugging with Kprobes

Managing calendars with Mozilla Sunbird

Securing wireless LANs with FreeRADIUS

USA $5.00 CAN $6.50
www.linuxjournal.com
05
0 71486 03102 4

LINUX JOURNAL · DEVELOPMENT · InfiniBand · Sunbird · Fd.o · VIA PadLock · FreeRADIUS · Kprobes · MAY 2005 ISSUE 133

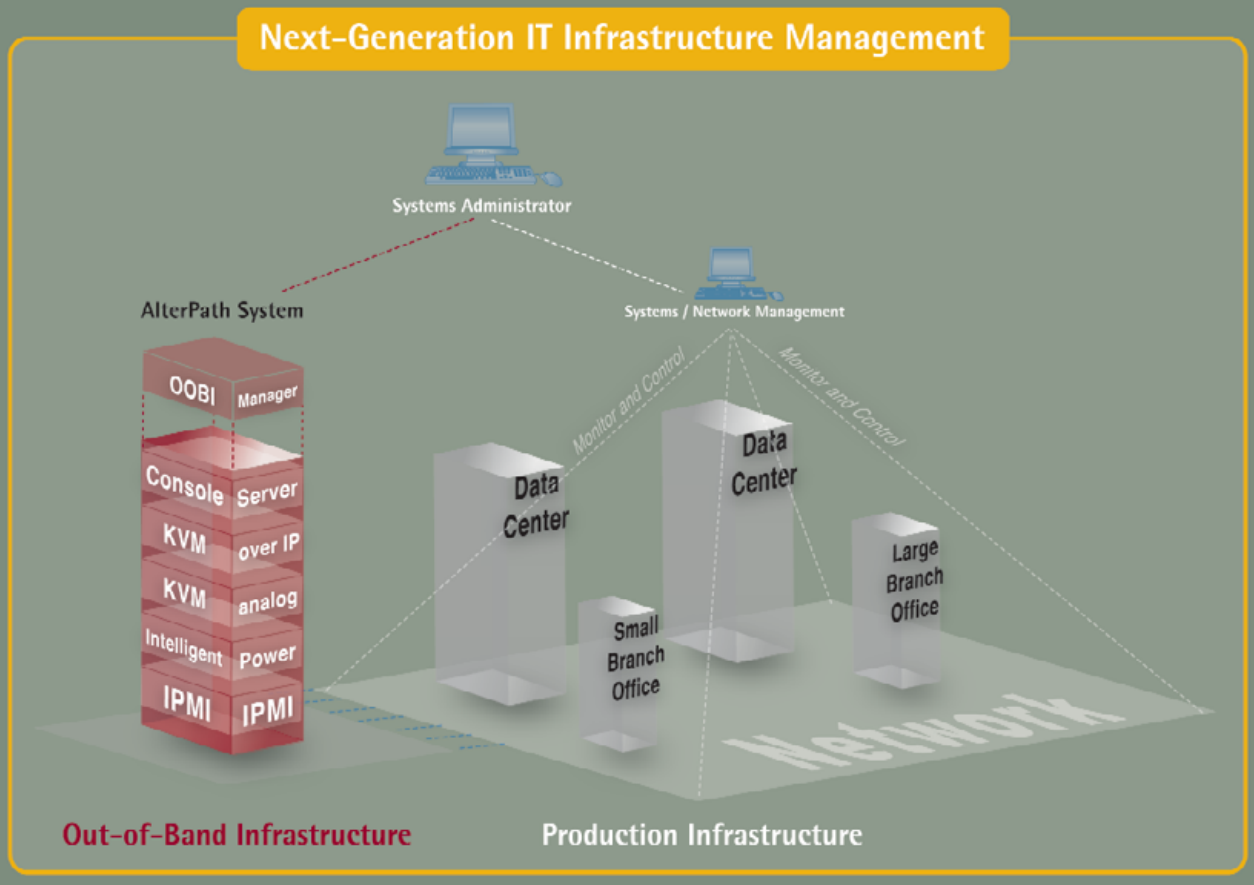

OOBI...
...do!
OOBI
OOBI
OOBI
OOBI
servers
storage
network devices
facilities

Let's talk OOBI

Next-Generation IT Infrastructure Management

OOBI, or Out-of-Band Infrastructure, is Cyclades' unique approach to enhance your IT infrastructure management capabilities. It integrates management of serial ports, KVM, KVM over IP, intelligent power distribution and IPMI devices in a secure, consolidated management solution for remote IT infrastructure administration. It complements system management tools like HP OpenView, IBM® Tivoli®, BMC PATROL® and CA Unicenter®.

An OOBI can cut costs and boost operational efficiency and productivity by minimizing the need for redundant equipment and personnel and, in most cases, eliminating the need for crash cart runs or remote visits to restore IT assets that have become disconnected from your production infrastructure.



To receive a **FREE** white paper on OOBI, Out-of-Band Infrastructure, visit us at **www.cyclades.com/oobiwp**

Over 85% of Fortune 100 choose Cyclades.

www.cyclades.com/lja

1.888.cyclades - sales@cyclades.com

cyclades®

Polywell First Class Linux Systems

The Most Reliable 64-bit Computing

2200A 1U 1TB Dual Opteron

- Dual AMD® Opteron™ 242 64-bit Processors
- 2GB 400MHz ECC DDR (Expandable to 16GB)
- 1TB (4x250G) SATA HD 7200RPM 8M
- 1 x PCI-X 133MHz Solt
- On-board Dual Gigabit Ethernet
- On-board ATI Graphics
- Slim CD-ROM Drive, Optional DVD-RW or CDRW
- Slim Floppy Drive
- 1U 24" Depth Rack Chassis with 460W P/S
- 4 x Swappable Drive Bays (SATA or SCSI)
- Supports Linux, FreeBSD or Windows
- Custom Configuration Available
- Please call for other Options

Order# 2200A242LJ05A

$2,550

Linux Appliance PCs

- Custom Made Odd Size Chassis
- AC or DC Power Supply
- Low Power Voltage Processor
- or High Performance AMD® Opteron™ Processor
- Diskless or Flash OS Boot Drive
- Swapable Hard Drive, CD-ROM, FDD
- Integrated Graphics, Ethernet, USB
- Optional LCD LED Control Module
- IS2 Audio, MPEG2/4 Hardware Video
- Upto 4 Ethernet Ports or 4 Serial Ports
- We have over 18 years OEM Experience in Set top Box, Digital Media Player, POS Kiros, Thin Client, Networking Appliance,

OEM Appliance starts at

$299

8422B 2U 4-Way Opteron

- Quad AMD® Opteron™ 846 64-bit Processors
- 4GB 400MHz ECC DDR (Expandable to 32GB)
- 2 x 133, 2 x 66MHz PCI-X, 1x32bit PCI Slots
- Dual Gigabit Ethernet, U320 SCSI Controller
- Dual Hot-Swap 36GB SCSI Drive for Mirror
- Slim CD-ROM Drive, Optional DVD-RW or CDRW
- 2U 28" Depth Rack Chassis with 700W PFC P/S
- On-board ATI Graphics
- Supports Linux, FreeBSD or Windows
- Custom Configuration Available
- Please call for other Options

Order# 8422B846LJ05B

$7,499

1U Power Saving ISP Server

- AMD® Sempron™ or Opteron™ Processor
- 512M Memory
- 80GB Hard Drive
- 10/100Mbit Ethernet
- We provide Drive Image Service
- 1U 14" Short Rack, allow 2 x 1U per Rack
- Low Power Usage to save your Data Center Cost
- Perfect Entry Level ISP Server or Appliance System
- IDE Flash Drive is available
- Supports Linux, FreeBSD or Windows
- Custom Configuration is Available
- Please call us to discuss your specification

Order# 6231U800ISP02E

starts at **$399**

8400B 4U 2TB 4Way Opteron

- Quad AMD® Opteron™ 846 64-bit Processors with Hyper Transport Technology
- 4GB 400MHz ECC DDR (Expandable to 32GB)
- 2 x 133, 2 x 66MHz PCI-X, 1x32bit PCI Slots
- Dual Gigabit Ethernet, On-board ATI Graphics
- 8-Channel SATA RAID-5 PCI-X Controller
- 2TB 8 x 250GB SATA 7200RPM Hard Drive 8M
- CDRW-DVD Combo Drive, Floppy
- Optional Tape Back Up, Media Reader
- 4U 26" Depth Rack Chassis
- 950Watt 3+1 Redundant Hot Swap Power Supply
- Supports Linux, FreeBSD or Windows
- Custom Configuration Available
- Please call for other Options

Order# 8400B46LJ05C

$8,888

SANAS 4U 9.6TB Storage

- SAN Ready NAS Storage Server with upto 24 x 400GB Hard Drive
- Dual Processor FreeBSD/Linux NAS Appliance
- upto 4 x 2G Fibre Channel Ports for SAN
- upto 8 Gigabit Ethernet Ports for NAS
- Supports UNIX, Linux, FreeBSD, Windows
- 24 x Hot Swap Drive Bays for SATA or SCSI Drives
- 950W 3+1 Redundant Hot Swap Power Supply
- Custom Configuration is available
- Remote Support Available

Order# SANAS6TBLJ05F

6TB NAS starts at **$7,999**

18 Years of Customer Satisfaction
5-Year Warranty, Industry's Longest
First Class Customer Service

AMD 64 Opteron™

888.765.9686

www.Polywell.com/us/LJ

POLY SYSTEMS

AMD64 architecture reduces I/O bottlenecks, increases bandwidth, and reduces memory latency. Critical information gets to those who need it quickly and efficiently.

Polywell Computers, Inc 1461 San Mateo Ave. South San Francisco, CA 94080 650.583.7222 Fax: 650.583.1974

COVER STORY

FEATURES

INDEPTH

EMBEDDED

TOOLBOX

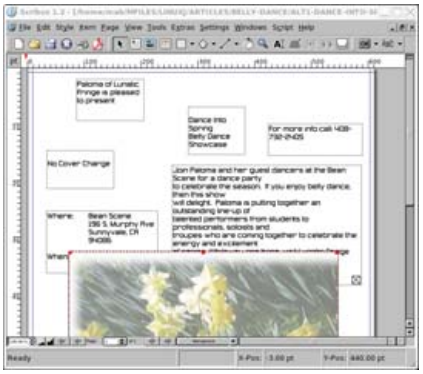
COLUMNS

REVIEWS

COVER PHOTO: MICHAEL BAXTER

LINUX JOURNAL
MAY 2005 ISSUE 133

DEPARTMENTS



Need to do a poster or flyer? Bring your layout ideas to print with Scribus (page 50).

NEXT MONTH

INTRANET

If you've been following the kernel changelogs, you'll notice new support for ATA over Ethernet. It really is what it sounds like—an inexpensive SAN using standard Ethernet hardware. Ed Cashin covers how to set up a fast storage array on a budget and grow it while mounted when you need more space.

Mick Bauer is all set to finish up the series on setting up WPA and RADIUS. The last step even includes adding in some non-Linux clients to protect them too. If your wireless network is your security weak point, you'll need this.

Back while the 2.6 kernel was in development, Red Hat borrowed 2.6 features and nonstandard patches to build a unique, customized 2.4 kernel. Now the 2.6-based Red Hat Enterprise Linux 4 is out, and Tim Burke is back with a look at how a distribution connects fast and furious Linux coding with risk-averse enterprise customers.

Keep Your Options Open

Good technology doesn't make you pick sides. Stay flexible with today's most versatile tools and standards. **BY DON MARTI**

Last time we did a special issue with a focus on software development, we called it the Cross-Platform Development issue. But really, the overwhelming majority of software that runs on Linux is cross-platform.

Sure, there are some Linux-exclusive tools, like the kernel debugger Kprobes (page 22). But the rest of the stuff in this issue, from Mozilla's Sunbird (page 14) and Firefox (page 86) to the versatile compiler suite GCC (page 78), is all wonderfully choice-preserving. Want to switch architectures? Use a different operating system? Even swap out your company's business model?

One software vendor I know decided to go from being a "pure software play" to a hardware company and pulled it off in a matter of months. There are even companies that will take your software load and turn it into a Linux appliance with your logo and everything, almost as easy as sending in a CD and manual for duplication.

Some OS vendors profit by imposing a high cost of switching. But in the long run, it's good to have users who can walk away. It makes you stay good at what you do and gives you instant feedback when you slip. Today's Linux users can get the same applications on a different platform with a quick visit to TheOpenCD.org or fink.sourceforge.net. We're not staying on Linux just because we'd lose time or mangle data by switching away—can any proprietary OS say the same?

As a software developer you have more options today than ever. You're not just choosing open source or proprietary or deciding between direct sales or channel. Develop for Linux and you can easily offer your software as download, shrink wrap, service or appliance. Get started with development before you have to make a final decision on the business model.

Speaking of choices, Greg Kroah-Hartman has a warning for you: don't try to read files in the kernel (page 38). He and the rest of the core kernel team just don't like it. But guess what? You have the freedom to read files in the kernel anyway. So if you have to do it, do it. An OS developer's decision that something is Bad doesn't apply to you.

In conclusion, beware of any technology that has an "evangelist". If a platform gives you enough choice that you don't have to trust it, it's a good sign that you can.

Don Marti is editor in chief of *Linux Journal*.

LINUX JOURNAL

MAY 2005
ISSUE 133

SUBSCRIPTIONS

E-MAIL: subs@ssc.com • URL: www.linuxjournal.com

PHONE: +1 206-297-7514 • FAX: +1 206-297-7515

TOLL-FREE: 1-888-66-LINUX • MAIL: PO Box 55549, Seattle, WA 98155-0549 USA • Please allow 4–6 weeks for processing address changes and orders • PRINTED IN USA

USPS *LINUX JOURNAL* (ISSN 1075-3583) is published monthly by SSC Publishing, Ltd., 2825 NW Market Street #208, Seattle, WA 98107. Periodicals postage paid at Seattle, Washington and at additional mailing offices. Cover price is $5 US. Subscription rate is $25/year in the United States, $32 in Canada and Mexico, $62 elsewhere. POSTMASTER: Please send address changes to *Linux Journal*, PO Box 55549, Seattle, WA 98155-0549. Subscriptions start with the next issue. Back issues, if available, may be ordered from the Linux Journal Store: store.linuxjournal.com.

Rackspace — Managed Hosting backed by Fanatical Support.™

Servers, data centers and bandwidth are not the key to hosting enterprise class Web sites and Web applications. At Rackspace, we believe hosting is a service, not just technology.

Fanatical Support is our philosophy, our credo. It reflects our desire to bring responsiveness and value to everything we do for our customers. You will experience Fanatical Support from the moment we answer the phone and you begin to interact with our employees.

Fanatical Support has made Rackspace the fastest-growing hosting company in the world. Call today to experience the difference with Fanatical Support at Rackspace.

LINUX JOURNAL READERS' CHOICE 2004

Thanks for
honoring us with the
2004 Linux Journal
Readers' Choice Award for
"Favorite Web-Hosting Service"

rackspace™
MANAGED HOSTING

1.888.571.8976 or visit us at **www.rackspace.com**

If You Can't Reach Esc, How Do You Quit vi?

Here is a picture of my three-week-old son Nicholas Robert getting his first taste of Linux. Love your magazine—keep up the fantastic work!

--
Dan Behman

Redundant Storage Idea

I am the System/Network Admin in an office environment of relatively new PCs (about 25) and a few servers. When I started, it was 100% Microsoft. Now our department relies heavily on two Slackware servers. One of the duties of the bigger server is a Samba file server that everybody has access to. I had the idea one day of a pseudo-RAID-5 implementation over the network. All these new PCs with 80GB hard drives, using only ~3GB, because everybody is using the Samba share.

What if someone took existing code from LVM and the highly stable RAID-x kernel modules and made a project that allowed you to create a file of a fixed size on each one of those Windows machines, say 40GB per machine, and then mount it as one large volume to be re-shared out to those who save everything to the X drive?

Because the Windows PC will have a single fixed 40GB file, it will be easy to configure remote or local backup software (I selected BackupPC) to ignore that file when performing a backup.

Your total storage with the above example would be (n–8)*40 where n is the number of PCs. 680GB of redundant storage is a big step up from the 80GB RAID 1 file server in use now. This would not be the best solution for large, frequently used files as the network would easily become a bottleneck, but for ever-increasing storage demands from archiving sales records and raw images for Web site products that are modified once then kept forever, it's a great solution. At least I think so. Thank you for a great magazine.

--
Chris Turner

If you can do that with good performance, stability and security, you'll never have to buy your own beverages at USENIX conferences.—Ed.

Focus, Please

I have been a subscriber to *Linux Journal* for several years now. One thing that I have noticed is that *LJ* doesn't seem to have a particular area to focus on. Are your articles intended for desktop users or for server administrators? Please choose your niche and stick to it.

In the meantime, I've just renewed my subscription for another year. I'd really appreciate seeing more desktop-oriented articles in the future, and less of the server- and network-administrator stuff. I subscribe to other magazines for that.

--
jh

Non-Linux OSes are cluttered with so much junk because people don't learn from other areas of development. The desktop, server and embedded environments have a lot to teach one another.—Ed.

One Computer, Hold the Fan

I'd like to second the suggestion by Ramer W. Streed in the February 2005 Letters section (page 6) for an article on fanless computers. Fans are noisy and irritating. Besides, my wife is a heavy smoker, and computer vents and heatsinks in our house tend to get clogged with smoke particles. Streed's request was for a fanless PowerPC, but I'd settle for anything without a fan, as long as it has reasonable capability and runs Linux.

--
A. T. Young

We might have a little surprise in store for you.—Ed.

Make the Competition Pay for Linux Info

My understanding of having ads in TV, radio and magazines is to generate money. As long as the ad is not ethically incorrect, why not publish it? Why not let Microsoft help pay to spread Linux knowledge? Microsoft had an ad on a Linux forum I visit regularly, and I would faithfully click on it every day. Sometimes twice a day. Would I use a Microsoft product? Not in the near future, but each and every click on that ad helped

keep that forum free. A free forum is a good match to a free operating system.

We live in the information age. It is not the Microsoft vs. Linux issue that so many people try to create. It is monopoly vs. free-flowing information. As a matter of fact, not allowing something because it can compete is doing business the Microsoft way. I would not object to seeing a Microsoft ad in a Linux magazine. As a matter of fact, I would like to see those expensive two-page or fold out ads pushing Microsoft products to pay for a new section called "newbie corner", or something like that. If nothing else, I would rather flip over that ugly Windows logo than pay more for my magazines.

--
Brad Peters

Cuddly Penguin

Here's a picture of my son Gabriel, age 1½ showing his affection to our beloved Penguin. Maybe we'll see it in one of your issues! All the best from one of your subscribers.

--
Paul

Cats Still Love SUSE Packaging

Greetings from Guatemala, Central America. Since 1999 this girl is in love with Linux and her cat!

--
David Salgado

Low-Priority Bug Report

In the diff -u section in the February 2005 issue, Zack Brown reports the bug found by Pavel Macheck in the 2.4 kernel: in year 9223372034708485227, or 9.22×10^{18}, all 2.4 kernels will immediately stop.

This is a great discovery that should be presented to the astronomical community. Following J.D. Barrow & F.J. Tipler in *The Anthropic Cosmological Principle* book, at the same time 2.4 kernel Linuxes will die, Neutron Stars will cool to 100 degrees Kelvin and planets' orbits will collapse via gravitational radiation. All of this will happen long after the Sun abandons the Main Sequence (5×10^9 years) and all the stars become white dwarfs (1×10^{12} years).

I think this is a new astronomical landmark that all Linux users should ask to include in the astronomical almanac of the foreseen history of the Universe.

--
Guigue

Two Penguins, One Baby

I first congratulate you and your team for your excellent *Linux Journal*—I wait for it every month. I attach a picture of my seven-day-old son, Sebastian, who seems to sleep very well with his favorite penguins. He doesn't like worms, bugs or horses (maybe trojan ones).

I'll let you guess which OS he most likely will be familiar/comfortable with in the near future. Penguins are a very common animal here in Chile. There is a group of them about 40km from where I live.

--
Marcelo Maraboli

Mmmm, Penguins

There is this fantastic confectioner in Walpole, New Hampshire that makes these cute little chocolate penguins. They also have a shop for gifts: www.laburdick.com. They will ship boxes and baskets and of course the penguin.

--
Daniel Hoviss

Certified by Whom?

I was working on a newly purchased server, and in the documentation package, I came across this CD. Nothing special about the disc itself, but the label was rather entertaining. Notice that the disc is both Microsoft Certified and Powered by Linux. Hope you find it as entertaining as I did.

--
Kris Linville

Which Laptop?

You've been fiddling with Linux laptops. Doc has an EmperorLinux Toucan (aka IBM

LETTERS CONTINUED ON PAGE 95

Photo of the Month: NASA Site Tour

These two future kernel hackers had not seen a computer taller than themselves until I took them to see NASA Ames' 10,240 CPU Linux system, Columbia. Tony (foreground), an avid collector of spongy penguins given out at various Linux events, now thinks computers were created by penguins, or vice versa. Ronnie, on the other hand, no longer just asks to go to Daddy's work. He now wants to see Daddy's "Big Work".

--
Wayne Vieira

Photo of the Month gets you a one-year extension to your subscription. Photos to ljeditor@ssc.com.—Ed.

MonarchComputer.com

Visit us online TODAY at
www.monarchcomputer.com

Get Half-Life 2 *FREE* by upgrading to AMD Athlon™ 64!

Visit our website to see how!

PayPal

Buy Online or by phone:

1-800-611-0875
1-8-MONARCHPC

Paypal - Visa - Mastercard - Discover - AMEX

Monarch makes it quick and easy to upgrade with FREE setup and testing on Motherboard Combos and $18.00 build fee on Barebones.

FREE INSTALLATION SETUP & TESTING BY CERTIFIED TECHS

FREE TECH SUPPORT!

AMD Motherboard Combos

Asus A8V-E Deluxe Mainboard with AMD Athlon™ 64 processor 3000+ (939) — Only $292

Abit AV8-3rd Eye Mainboard with AMD Athlon™ 64 processor 3200+ — Only $331

Tyan S2882G3NR Mainboard with AMD Opteron™ processor 244 — Only $621

Tyan S2882G3NR (Thunder K8S) with AMD Opteron™ processor 248 — Only $859

Abit AV8-3rd Eye K8T800 w/ AMD Athlon™ 64 processor 3200+ (939) — Only $334

Tyan S2882G3NR (Thunder K8S) MB w/ AMD Opteron™ processor 252 — Only $1243

Mainboard - Processors - Heatsink and Fan with Memory Options - FREE INSTALLATION AND TESTING
LASTEST BIOS loaded for easy upgrades - AMD Athlon™ XP, Athlon™ MP, Athlon™ 64, Athlon™ 64 FX, and Opteron™ Combos Available

AMD Barebone Systems

Go to www.monarchcomputer.com. select Barebones from the Menu. Choose AMD Athlon™ XP or AMD Athlon™ 64. Then configure your barebones online or call 1-8-MONARCHPC.

Lian-Li PC-V1200 Server Case
w/Wheels w/460W PS
Tyan Thunder K8W S2885ANRF
AMD Opteron™ processor 244 1.4 GHz
Starting @ **$921**

Antec Plusview 1000AMG
w/460W PS
Abit AV8-3rd Eye K8T800
AMD Athlon™ 64 processor 3500+
(939 - 90nm)
Starting @ **$575**

SliverStone SST-TJ06-B (Black)
E-ATX w/460W PS
Tyan S2882G3NR Thunder K8S
AMD Opteron™ processor 248
Starting @ **$1138**

Antec Performance One P160 Case
w/420W PS
Abit KV8 PRO Motherboard w/
AMD Athlon™ 64 processor 2800+ (754)
Starting @ **$438**

***AMD Athlon 64 and Athlon 64 FX are the ONLY Windows®-compatible 64-bit PC processor

www.monarchcomputer.com

AMD 64-bit CPUs

AMD Athlon™ 64 OEM CPUs

AMD Athlon™ 64 2800+ (754) $122.00
AMD Athlon™ 64 3000+ (754) $149.00
AMD Athlon™ 64 3200+ (754) $194.00
AMD Athlon™ 64 3400+ (754) $223.00
AMD Athlon™ 64 3500+ (939) $272.00
AMD Athlon™ 64 3700+ (754) $329.00
AMD Athlon™ 64 3800+ (939) $424.00
AMD Athlon™ 64 4000+ (939) $643.00
AMD Athlon™ 64 FX-55 (939) $828.00

AMD Athlon™ 64 CPUs Feature:
- HyperTransport™ technology
- Enhanced Virus Protection for Microsoft® Windows® XP- SP2
- Cool'n'Quiet™ technology
- AMD64 technology

AMD Opteron™ OEM CPUs

AMD Opteron™ 144 1.8GHz $173.00
AMD Opteron™ 146 2.0GHz $211.00
AMD Opteron™ 148 2.2GHz $270.00
AMD Opteron™ 150 2.4GHz $404.00

AMD Opteron™ 244 1.8GHz $203.00
AMD Opteron™ 246 2GHz $307.00
AMD Opteron™ 246 HE (55W) 2GHz $441.00
AMD Opteron™ 248 2.2GHz $441.00
AMD Opteron™ 250 2.4GHz $669.00
AMD Opteron™ 252 2.6GHz $825.00
AMD Opteron™ 844 1.8GHz $677.00
AMD Opteron™ 846 2GHz $677.00
AMD Opteron™ 846HE (55W) 2.0GHz $847.00
AMD Opteron™ 848 2.2GHz $847.00
AMD Opteron™ 850 2.4GHz $1130.00
AMD Opteron™ 852 2.6GHz $1469.00

NEW AMD OPTERON™ 252 & 852
Available at our website
www.monarchcomputer.com

Components and Upgrades
1000s of In-Stock Components

100756
Apex, FoxConn Mid-Tower TU-150 400W PS USB 2.0 Ports (Black)
$67.00

150137
Western Digital 250 GB WD2500JB 7200 RPM 8MB IDE
$151.00

140270
1 GB DDR (266) PC-2100 REG ECC Corsair (CM72SD1024RLP-2100)
$229.00

140662
512 MB DDR (266) PC-2100 REG-ECC Mushkin Pro (990742)
$136.00

190148
eVGA GeForce FX 5200 128MB DDR/PCI/S-Video DVI/No-VGA
$62.00

150239
Western Digital 74 GB SATA 10K Raptor (WD740GD)
$182.00

100963
Ultra X-Connect ULT31558 ATX 500 Watt Power Supply w/2-80mm Fans (Blue)
$92.00

110407
Asus A7V8X-X KT400 AUDIO/LAN DDR 400 8X /4X AGP 333 FSB XP
$54.00

Visit www.monarchcomputer.com for products in this ad, current inventory and up-to-the-minute pricing. Call or email for special orders

Educational and Government POs Welcome.

GSA Schedule Contract GS-35F-0202P

We ship to the Continental U.S., Alaska, Hawaii, APOs, Puerto Rico, and Canada

Commercial leasing available for purchases as low as $1000.

Prices subject to change without notice. Monarch Computer not responsible for typographical errors.

Monarch Has The *LOWEST PRICES* Custom 64-Bit Servers, Workstations & Desktops

AMD 64 Athlon FX | AMD 64 Athlon | AMD 64 Opteron

Exlusive Athlon™ 64 FX Launch Partner

NEW! Monarch's EMPRO line makes buying AMD Opteron™ Workstations and Servers Easier than ever before!

A FULL LINE OF WORKSTATIONS, RACK SERVERS, TOWER SERVERS

Choose a preconfigured System Special, upgrade any component or accessory, even customize your system from the inside out - see the dynamic, real-time pricing on your configuration, and save a quote to lock in pricing for up to 7 days!

Part:80340
Monarch Furia Deluxe Workstation Special
w/ AMD Athlon™ 64 3800+
1GB PC3200 DDR
PNY Quadro FX500
ONLY $1802.00

Part:90999
Monarch Centira Ultimate Desktop Special
w/ AMD Sempron™ 3100+
512MB DDR
ATI 9600SE AGP
ONLY $739.00

NEW SYSTEM MODELS!

Value - The most value for the lowest price

Deluxe - The best balance of performance and price

Ultimate - The best of the best in cutting-edge technology

8 New System Lines,Including:

Centira™ - Starts @ **$359** (Athlon™ XP or Sempron™)

Solia™ - Starts @ **$459** (Athlon™ 64 or Sempron™)

Furia™ - Starts @ **$1085** (Athlon™ 64 or Athlon™ 64 FX)

Empro™ - Starts @ **$1494** (Opteron™)

www.monarchcomputer.com/empro

The AMD Opteron processor— built upon forward-thinking AMD64 technology—provides flexibility with a 1–8-way scalable design.

The AMD Athlon™ 64 processor improves security against certain types of viruses, with Enhanced Virus Protection for the Microsoft® Windows® XP SP2

GET QUICK QUOTES
Online or by phone:
www.monarchcomputer.com
1-800-611-0875
1-8-MONARCHPC
Paypal - Visa - Mastercard - Discover - AMEX

The Official Linux Journal Web Server
Monarch Computer Systems

Now available with AMD Opteron™ processor 252!

Part# 80544

- Lian-Li PC-V1200 Aluminum Quiet Tower (Black) -550W Antec Power Supply)
- Tyan Thunder K8W S2885ANRF Board AGP Pro 4X 8X SATA/IEEE
- (2) AMD Opteron™ Processors 252
- 2 GB (4 pcs 512 MB) DDR (400) PC-3200 REG ECC Corsair Memory (TwinX1024RE-3200LL)
- (4) Western Digital 74 GB 10K RPM Raptor (WD740GD) 8 MB Cache SATA HDDs
- 3ware Escalade 9500S-4LP 4 Port SATA RAID Controller w/ RAID 5 Setup
- Asus 5232ASQT-BLK 52X32X52 CD-RW (Black
- Plextor PX-712A/SW-BL DVD±R/RW
- Mitsumi Floppy 7-in-1 Smart Card reader (Black)
- Creative Audigy 2 ZS 24BIT Advanced HD Sound
- PNY QuadroFX 3000 AGP 8x/4x 256MB
- Linux Fedora Core 3 A64 Installed
- 1 or 3 year warranties available

The *NEW* MONARCH ULB for 2005
Custom Opteron Workstation

As Reviewed in the Dec. 2004 issue of Linux Journal!

On SALE!
Starting @ $1276

As configured at left:
$5934

$100 Off Reviewed Price!

Browse now to monarchcomputer.com for a full list of available options

Commercial leasing available for purchases as low as $1000.

Prices subject to change without notice. Monarch Computer not responsible for typographical errors.

GSA Schedule Contract GS-35F-0202P

We ship to the Continental U.S., Alaska, Hawaii, APOs, Puerto Rico, and Canada

On the WEB

» If you're curious about how Joe Ruffolo and Ron Terry used the various open-source software mentioned in their article (page 82) to connect the Utah College of Applied Technology satellite campuses, see their follow-up Web article, "A Reading List for Linux in the Classroom" (www.linuxjournal.com/article/8124). There, they point to good sources for more information on Samba, OpenLDAP and other software, so you can do your own project.

» Didn't make it to Germany for CeBit 2005? We didn't either, but Stefan Cars did. Check out his show review, "CeBit 2005: On the Scene in Hannover" (www.linuxjournal.com/article/8125), to see what you missed. And with "6,207 exhibitors (52% from abroad) and a net display area of 309,000 square meters", we're guessing you missed a lot.

» Our Web article series on embedded development wrapped up recently with "An Introduction to Embedded Linux Development, Part 4" (www.linuxjournal.com/article/8122). In the final installment, Ben Anderson and Richard Sevenich discuss how to use the Background Debug Mode provided in Motorola processors. Meanwhile, Larry Finger wraps up his Web series with "Linux in a Windows Workstation Environment, Part 3" (www.linuxjournal.com/article/8126), a description of using "our Linux server to provide VPN tunnels that secure our users' transmissions over a Wi-Fi network that is required to be unsecure".

diff -u

What's New in Kernel Development

Wichert Akkerman, the former Debian Project leader, noticed some pretty weird behavior under 2.6.10-ac10. Doing a `df` command, he found that his disk usage was reported to be -7378697629483827736. Suspecting an error, he posted to the linux-kernel mailing list, but although several folks offered speculation on what had happened, Wichert had fixed the problem with e2fsck and was unable to do further tests to confirm one explanation or the other. This sort of thing happens from time to time—a strange, unexplained anomaly. Maybe it will appear again in some later kernel version and be diagnosed, or maybe it was a hardware glitch.

Mitch Williams recently found that files in **SysFS** could not be appended to. Any attempt to do so would overwrite the old data with the new. Even opening the file and seeking to the end before writing would have the same effect. **Greg Kroah-Hartman** confirmed that this was not at all the intended behavior, especially considering that SysFS would overwrite the data without giving any error message. Mitch had a patch ready to correct both cases—appending and seeking—and after some small patch-splitting discussion with Greg, it looks as though SysFS's behavior will change so file operations appear more normal.

A new **security@kernel.org** mailing list has been created. The purpose of this list is to receive reports of security exploits before they become commonly available, so the Linux developers can create and distribute fixes before attackers can create and deploy attacks. One of the features of this list is that subscription is by invitation only, and the archives are not made immediately available, as they are with the regular linux-kernel mailing list. **Linus Torvalds**, who has said he personally would much prefer a completely open style of development, has joined the list, with the idea that the greater secrecy urged by folks like **Marcelo Tosatti** regarding nonpublic security issues might turn out to be a good idea after all. In any event, he's willing to try it and see. This sort of issue inevitably will be controversial, especially among strong advocates of the open-source development model.

Jake Moilanen has created a **genetic algorithm** library within the kernel to help tune the **input/output scheduler**, as well as the **process scheduler**. Traditionally, these schedulers (especially the process scheduler) have been notoriously difficult to get right, because of the tremendous variety of user behavior. How can developers be sure that any particular algorithm will work best under any particular set of user activities? Really, they can't. Something like Jake's work, if feasible, could pave the way for an entirely new method of tuning kernel parameters. At the same time, genetic algorithms tend to be unpredictable in their results, and unpredictability is not necessarily desirable in a kernel. I imagine developers would be resistant to including this sort of patch unless it could produce very large and measurable performance improvements. Even then, they might want to include only the results of the genetic tuning and not the actual genetic algorithm itself. Time will tell.

In the saga of **Software Suspend**, **Pavel Machek** recently enabled **swsusp** for SMP machines. Before now, this had not been supported, but apparently starting with 2.6.11 it should be possible to use Software Suspend successfully on SMP systems. Little by little, the swsusp code advances, and the controversy and acrimony of competing code bases that we saw over the past year is starting to fade away. Of course, Software Suspend is an inherently tricky problem, because some hardware simply won't cooperate. In such situations controversy will be inevitable, and the difficulty of knowing the best way to tackle a given problem tends to become a question of unpleasant trade-offs. But, swsusp certainly is looking very promising right now.

There's been a lot of maintainership activity recently among new and existing kernel projects alike. **Andrew Vasquez** is now the official maintainer of the QLogic QLA2XXX FC-SCSI driver. **Tony Luck** has taken over IA-64 maintainership from **David Mosberger-Tang**. **Matthias Kunze** has taken over the apparently unmaintained Enhanced Linux Progress Patch and forward-ported it to 2.6.10. **Adrian Bunk** has taken over the util-linux project from **Andries Brouwer**, after Andries had put out a call for a new maintainer back in September 2004.

Related to the issue of maintainership, the **MAINTAINERS file** may start identifying mailing lists that can receive posts only from subscribers. Traditionally, Linux development lists are open to all posters to encourage bug reports from as many users as possible, but not all kernel-related projects agree with this policy. For those who don't, folks like **Domen Puncer** have been submitting patches to identify those lists as subscriber-only. Earlier, Domen had tried removing mailing lists like the linux-arm-kernel list from the MAINTAINERS file for this reason, but after some negative feedback from folks like **Alan Cox**, he opted for his current approach instead.

—ZACK BROWN

Cyclades' AlterPath™ Manager is the first secure enterprise IPMI manager to boost operational efficiency and productivity with industry leading features:

- **Vendor-independent IPMI management**
- **Enterprise security and authentication (SSH v2, LDAP, RADIUS, TACACS+, Kerberos)**
- **Centralized access, control and auditing**
- **Event notification and alarms**
- **Scalable to 5000 devices and 256 simultaneous connections**

- **Remote IT infrastructure administration**
- **Centralized OOBI™* management**
- **Remote incident resolution**
- **Web-based access**
- **Remote logging and monitoring**
- **Supports IPMI 1.5 and 2.0**

We've worked our magic. Now you can work yours.

To receive a **FREE** white paper on IPMI, visit us at www.cyclades.com/ipmiwp

*OOBI, or Out-of-Band Infrastructure, integrates management of serial ports, KVM, KVM/IP, intelligent power distribution and IPMI devices in a secure, consolidated management solution for remote IT infrastructure administration.

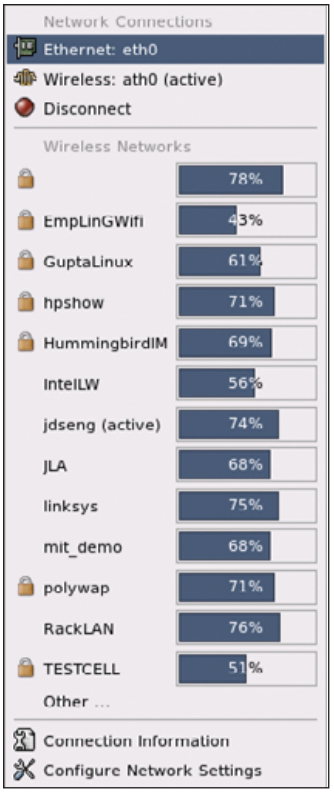
Over 85% of Fortune 100 choose Cyclades.

www.cyclades.com/ljb

1.888.cyclades · sales@cyclades.com

cyclades

Best Network Pickings



Netapplet is a neat little network interface control applet that lets users select networks and interfaces. Because the air in most civilized places is now thick with Wi-Fi signals, available networks are represented by horizontal bars representing signal quality and handy little padlock icons representing WEP locks. That means you can warwalk (or wardrive, or warfly) and select open networks on the fly. This makes it a must-have app for every mobile Linux user.

It was written, coincidentally, by contributing editor Robert Love, along with his colleague Joe Shaw, but I discovered it while walking the floor at the latest LinuxWorld Expo, looking for cool stuff. It was the coolest thing I saw at the show.

Robert points out that netapplet has a few other neat features too. It automatically will fall back to a new wireless network when your current network goes away. It stores WEP keys in an encrypted gnome-keyring for automatic reconnection and works around a bunch of wireless driver bugs to make the experience better. By the way, it works with wired networks and PPP dial-up connections as well.

Robert (who wrote *Linux Kernel Development* and works for Novell) wrote netapplet for SUSE, but packages are available for other distros.

More information is available at primates.ximian.com/~rml.

—DOC SEARLS

Ten Years Ago in *Linux Journal*

Internet services were the flavor of the month for the May 1995 issue, and Eric Kasten covered the basics of setting up the CERN and NCSA Web servers. Hacking the source to fix a vulnerability listed on CERT, along with editing the Makefile, were key steps for getting the latter going.

Piers Cawley covered setting up the Majordomo mailing-list manager, including how to add a patch to force Majordomo to send out a digest if messages are sitting around for too long. Daniel Hollis covered setting up a Linux server for an ISP. The hardware included an Intel 486DX4/100 processor, a 16-port serial card and 28.8k and 14.4k modems.

Walnut Creek CD-ROM advertised a two-CD Slackware set. The minimum memory requirement was 4MB, and according to the ad, a "typical installation" with development tools, Sendmail and X would require about 40MB.

Finally, Linus Torvalds released Linux 1.2.0 with a parody of Microsoft licensing terms, including the "I've got too much money" license. The real license in the actual code stayed GPL, of course.

—DON MARTI

Freedom-Compatible HDTV Boxes: Time Is Running Out

Linux user group mailing lists are buzzing with announcements for personal video recorder "Build-Ins". The Electronic Frontier Foundation (EFF) has put together easy-to-follow instructions for turning a PC plus a high-definition TV card into a Linux-based personal video recorder, using Knoppix and MythTV.

Seth Schoen wrote about the threat of the proposed Broadcast Flag regulation two years ago in *Linux Journal*. Unfortunately, for anyone interested in Fair Use or do-it-yourself HDTV projects, the US FCC adopted the regulation, which will put draconian "robustness" requirements on any device that handles an HDTV signal, in effect making it illegal to support a GPL device driver. The regulation is scheduled to go into effect on July 1, 2005.

The EFF, along with other organizations including the American Library Association, Consumers Union and the Medical Library Association, has filed a lawsuit to challenge Broadcast Flag. Lawyers for both sides argued the case before the US Circuit Court of Appeals for the District of Columbia in February 2005.

Instead of biting your nails waiting for the courts, get parts and build a PVR while you still can. Turn your friends who want a cool system like yours into technology freedom supporters. One group, Bay Area Debian, turned the "build-in" into an excuse for a group trip to Fry's, the discount electronics store where aisles sometimes turn into informal peer-to-peer Linux hardware support seminars.

Check out the EFF's Broadcast Flag page. Their "cookbook" is a great first family Linux project: eff.org/broadcastflag.

To order your pcHTDV card ($169.98 US), visit www.pchdtv.com.

—DON MARTI

Your Network.

Your Network...

...on TOE

Enabling networks to cover more ground faster.

Plug SBE's TCP/IP Offload Engine (TOE) into your system to see the performance boost for yourself...

All TOEs should process TCP/IP at network speeds, provide full segmentation and assembly, terminate multiple simultaneous sessions, and minimize transaction latency, without host intervention. However, not all TOEs are alike...depending on the individual manufacturer's target market, the devices vary in their ability to fully handle these essential criteria.

Adding a TOE to your existing system is only a cost-effective option if it can truly heighten your network performance at a fraction of the cost of purchasing an additional server. Only one TOE board has proven to provide peak performance across all four metrics of TOE effectiveness. While other TOE vendors offer solutions capable of satisfying one, two or maybe three of the critical TOE performance metrics,

SBE is today's only source to deliver Gigabit Ethernet throughout at line rate, over 70% reduction in CPU utilization, 32 microsecond transaction latency, and support for high session count applications, all on one PCI or PMC-based board.

Skeptical? Burned by inferior TOE solutions in the past? Well, now you can take advantage of our risk-free offer to see the results of SBE's TOE for yourself...

Contact us to qualify for a free test drive of the SBE TOE board in your application. Plus, ask us about how to win a free iPOD® mini.*

Phone: 800-214-4723
Email: info@sbei.com
Web: www.sbei.net/tryTOE.htm

SBE®

Linux On Demand

flexibility on demand | 925-355-2000 | info@sbei.com | www.sbei.com

* Restrictions apply. While supplies last. iPOD mini is a registered trademark of Apple Computer, Inc.

Sunbird and iCalendar

Mozilla's Sunbird calendar combines the advantages of centralized, cooperative Web applications with the speed and usability of a cross-platform desktop tool. **BY REUVEN M. LERNER**

When I first started to write server-side software, I laughed at the thought that I was writing applications. After all, I was writing only small bits of code; nothing I did could hold a candle to what a real program, running on the desktop of someone's computer, could do.

Of course, things have changed quite a bit in the computer industry since those early days. Today, Web-based applications are not only an established fact of life, but they seem to be playing an increasingly prominent role in our daily lives. Recently, I began to look into software that I could use to prepare my US income taxes. I shouldn't have been surprised to discover that many companies now are offering Web-based tax calculation programs. The term ASP, or application service provider, was hot several years ago, when it seemed as if all software would work over the Web. Although there have been some obvious success stories, there also were many failures, for technical and business reasons alike.

It's easy to understand why Web-based applications are attractive to a business: you no longer have to test your software on every platform but instead only on a handful of browsers. You no longer need to support many different versions of the software, because only one version is accessible at any given time. Bug fixes and software updates can be integrated into the system almost continuously. The software is available from anywhere with an Internet connection, instead of only on the computer on which it was installed. The list goes on and on. From many perspectives, this approach makes more sense than stamping out thousands of CDs, testing the software on hundreds of configurations and staffing a large call center to support all of those configurations.

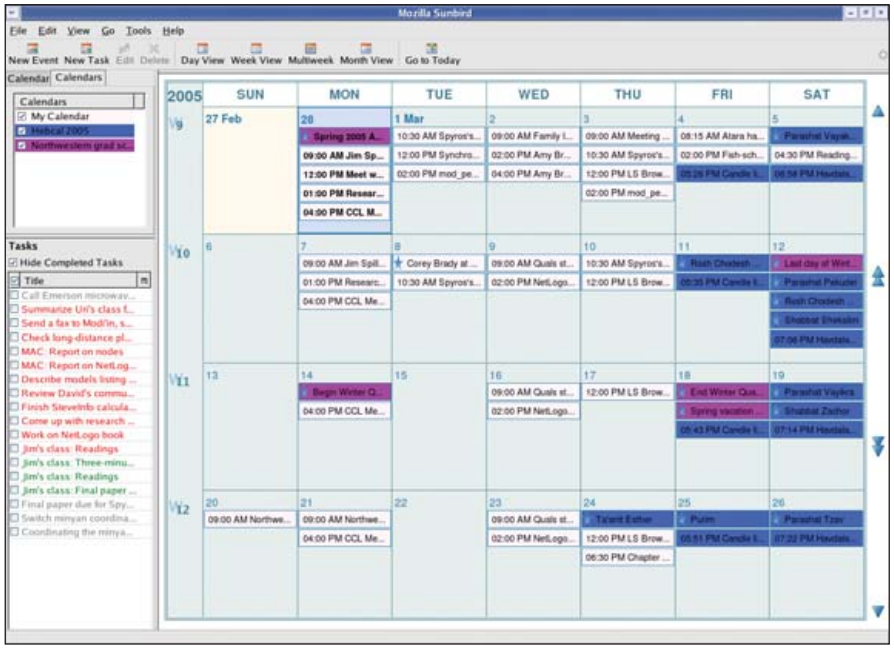


Figure 1. The main Sunbird window, in multiweek mode. Two of my three calendars (Hebcal 2005 and Northwestern grad school) are color-coded, both in the definition pane and the main window. Also notice how items in my to-do list are color-coded, indicating whether they are on time, late, in need of attention or ongoing.

But for all of the hoopla, Web applications still are limited compared to their desktop counterparts. Because all serious processing is done on the server, including writing to and reading from databases and files, instantaneous feedback from the interface is almost impossible. Even with the fastest servers and a lot of clever magic, such programs still can seem somewhat tedious. Google's new maps system (see the on-line Resources), for example, demonstrates that it is possible, albeit difficult, to create Web applications that feel much like their desktop counterparts.

Those of us without Google's resources increasingly are turning to another solution, namely using hybrid software—desktop applications that rely heavily on Web technologies. It used to be that Web technology could be described, with a fair degree of precision, as HTML-formatted documents retrieved by way of HTTP using a URL. Web browsers were, for a long time, the only programs that made serious use of these standards.

Today, however, a growing number of desktop programs make use of HTML, HTTP and URLs, even though they aren't Web browsers. They use URLs to locate remote resources, HTML for its simple, universally understood method of creating hyperlinked documents and HTTP because it is reliable, simple, universal and cacheable. There aren't too many examples of word processors and spreadsheets using these protocols—at least, not that I'm aware of—but one hybrid program has been playing an increasingly prominent role in my

EmperorLinux

...where Linux & laptops converge

The Meteor: 3lb Linux

- Sharp Actius MM20/MP30
- 10.4" XGA screen
- X@1024x768
- 1.6 GHz Transmeta Efficeon
- 20-40 GB hard drive
- 512-1024 MB RAM
- CDRW/DVD (MP30)
- 802.11b/g wireless
- Ethernet/USB2
- ACPI hibernate
- 1" thin

The SilverComet: 4 lb Linux

- Sony VAIO S270
- 13.3" WXGA+ screen
- X@1280x800
- 1.5-2.0 GHz Pentium-M
- 40-100 GB hard drive
- 256-1024 MB RAM
- CDRW/DVD or DVD-RW
- 802.11b/g wireless
- 10/100 ethernet
- ACPI hibernate
- USB2/FireWire

The Toucan: 5 lb Linux

- IBM ThinkPad T series
- 14.1" SXGA+/15.0" UXGA
- X@1400x1050/X@1600x1200
- ATI FireGL graphics
- 1.6-2.1 GHz Pentium-M 7xx
- 40-80 GB hard drive
- 512-2048 MB RAM
- CDRW/DVD or DVD-RW
- 802.11b/g wireless
- 10/100/1000 ethernet
- APM suspend/hibernate

The Rhino: 7 lb Linux

- Dell Latitude D810/M70
- 15.4" WUXGA screen
- X@1920x1200
- NVidia Quadro or ATI Radeon
- 1.73-2.13 GHz Pentium-M 7xx
- 30-80 GB hard drive
- 256-2048 MB RAM
- CDRW/DVD or DVD+/-RW
- 802.11b/g wireless
- 10/100/1000 ethernet
- USB2/SVideo/serial

Since 1999, EmperorLinux has provided pre-installed Linux laptop solutions to universities, corporations, and individual Linux enthusiasts. We specialize in the installation and configuration of the Linux operating system on a wide range of the finest laptop and notebook computers made by IBM, Dell, Sharp, and Sony. We offer a range of the latest Linux distributions, as well as Windows dual boot options. We customize each Linux distribution to the particular machine it will run upon and provide support for: ethernet, modem, wireless, PCMCIA, USB, FireWire, X-server, CD/DVD/CDRW, sound, power management, and more. All our systems come with one year of Linux technical support by both phone and e-mail, and full manufacturers' warranties apply. Visit www.EmperorLinux.com or call 1-888-651-6686 for details.

Custom Configurations

Linux Pre-Installed

Technical Support

www.EmperorLinux.com 1-888-651-6686

YOUR HIGH PERFORMANCE COMPUTING SOLUTION HAS ARRIVED.

VXRACK™ with the Intel® Xeon™ processor helps you simplify computing operations, accelerate performance and accomplish more in less time.

1 Choose one of the 3 convenient rack sizes

VXR-96
Rack accomodating up to
96 VXBlades/**192 Processors**
36TB of aggregated **Storage**
1.15TB of Global Memory
Power Distribution Included
Patented Architecture
Advanced Cooling System
Integrated InfiniBand Cable Mgnt.

$ 1,750.00*

VXR-128
Rack accomodating up to
128 VXBlades/**256 Processors**
48TB of aggregated **Storage**
1.5TB of Global Memory
Power Distribution Included
Patented Architecture
Advanced Cooling System
Integrated InfiniBand Cable Mgnt.

$ 2,190.00*

VXR-72
Rack accomodating up to
72 VXBlades/**144 Processors**
27TB of aggregated **Storage**
864GB of Global Memory
Power Distribution Included
Patented Architecture
Advanced Cooling System
Integrated InfiniBand Cable Mgnt.

$ 1,590.00*

CIARA TECHNOLOGIES

For more Information call
or visit us at

VXB-7221B
Intel SE7221B Motherboard
800MHz Front Side Bus
Intel® Pentium® 4 3.2GHz
1GB DDR2 400 Memory
Single 40GB 7200RPM ATA Drive
One **PCI/Express Slot Available**
Dual 10/100/1000 Intel Lan Port
350W Power Supply

$ 985.00

VXB-7501W
Intel SE7501W Motherboard
533MHz Front Side Bus
2 x Intel® Xeon™ 3.06GHz
2GB DDR 333 ECC Reg.Mem
Single 40GB 7200RPM ATA Drive
One **PCI/X Slot Available**
Dual 10/100/1000 Intel Lan Port
350W Power Supply

$ 2,355.00

INTEL® EM64T XEON™ AVAILABLE NOW

VXB-7520J
Intel SE7520J Motherboard
800MHz Front Side Bus
2 x Intel® EM64T Xeon™ 3.2GHz
2GB DDR2 400 ECC Reg.Mem
Single 40GB 7200RPM ATA Drive
One **PCI/Express Slot Available**
Dual 10/100/1000 Intel Lan Port
500W Power Supply

$ 2,950.00

2 Choose one or more type of VXBlade

3 Add, Mutiply,That's it. Easy as 1, 2, 3...

For example you choose the following: One VXR-96 with 48 Dual Intel® EM64T Xeon™ and 40 Single Intel® Pentium®4. You take 1 (VXR-96) + 48 (VXB-7520J) + 40 (VXB-7221B)...That's it

VXRACK

THE FUTURE OF CLUSTER TECHNOLOGY

CIARA TECHNOLOGIES...A GLOBAL SOLUTION PROVIDER.

Ciara Technologies is a world-class computer systems manufacturer. Ciara designs, develops, manufactures, markets, services, and supports a variety of computer systems including graphic workstations, rackmount and tower servers, networked storage and the newly acclaimed VXRACK™ Cluster Technology. The company's state of the art supercomputer cluster is based on the Intel IA32 and IA64 architectures and utilizes Linux operating systems. We are proud to be recognized by Intel as an "Intel Premier Provider". Choosing Ciara is choosing a single point of contact for all your IT requirements. All our products are built under the ISO 9001 standards and regulations. The growth of Ciara enabled the company to move its 300+ employees,in February 2003, to an ultra-modern plant of 576,000 sqft.. Ciara now has the capability of producing more than 500,000 units per year.

866-7VX-RACK (866-789-7225)
WWW.VXRACK.COM

intel inside XEON™

life, Mozilla Sunbird.

Sunbird (Figure 1) is the standalone version of the calendar extension that can be installed along with either Firefox or Thunderbird. Integration with these two programs is far from perfect, and I sometimes want to run or restart one without the other. So I installed Sunbird over the summer and have been pleased with each new release as it is made available.

Now, you might think there is nothing inherently useful about having a calendar use Web technologies. But in the case of Sunbird and the iCalendar standard, there is a major benefit—namely, the ability to create calendars for public consumption. This month, we begin a several-month journey through the creation, distribution and sharing of calendars based on the iCalendar standard. Along the way, we'll see not only how to work with iCalendar, but how the hybrid applications can provide a powerful combination of features and an enhanced user experience.

iCalendar and Sunbird

iCalendar is an Internet standard for sharing calendar information across different computers. The basic idea is simple: if all users at my office keep track of their schedules on their own computers, it makes things efficient for those individuals but no better for the group than if they were using a pocket diary. Scheduling meetings still would be a hassle. Moreover, group events would have to be entered once on each person's calendar—meaning that when a meeting moves from Monday to Wednesday, each team member needs to adjust his or her individual calendars accordingly.

iCalendar was designed to solve this problem by standardizing the calendar files themselves such that those files can be transferred from one program to another. The original vision, as far as I can tell, pictured people using programs that implement iCalendar on their computers and sharing that information with others by way of the network and Internet. The reality has taken some time to catch up with this theory, but a variety of programs now are available that do implement parts of the iCalendar suite.

I should note that the entire iCalendar Project has been the victim of some bad and unlucky naming problems. The file in which data is stored and that can be used to interchange information is called vCalendar, just as the electronic business-card format is known as vCard. But many people and applications, including Sunbird, refer to the file format as iCalendar, even though the file identifies itself as vCalendar. As you can imagine, the term iCalendar has been shortened to iCal, which is especially unfortunate, given that Apple's Mac OS X operating system comes with a program called iCal that uses the vCalendar file format. Because the use of vCalendar to describe these files seems to have gone the way of the dodo, I simply use the word iCalendar to refer to both the file format and the overall standard.

Download and install the appropriate version of the Sunbird standalone calendar program; see Resources for the URL. If you're a bit more daring, you can install one of the nightly builds; I actually am using Sunbird as my primary calendaring application, so I have been using the official builds. If you prefer to have your calendar integrated into either Firefox or Thunderbird, go to the main download page and choose the appropriate extension and version that you would like to install. If you install an extension to Firefox or Thunderbird, you need to restart the host program before continuing.

Sunbird allows you to create two types of items, events and tasks. Events normally appear on the calendar itself and can include holidays and meetings. Tasks normally appear on the left side of the screen and indicate things that you should get done, with an optional starting and ending date. Sunbird changes the color of tasks according to how soon they need to be done; overdue tasks are in red, current ones are in blue and future ones are in green. Gray tasks are in the far-off future, and crossed-out ones—if you choose to display them—are completed tasks.

Both events and tasks can be repeating, meaning we can schedule a meeting for every Wednesday at 4 PM over the next ten weeks rather than entering ten individual events in the calendar. We can enter exceptions to such recurring events as needed, and we can set them to recur every few days, months or years, with "every few" being user-definable.

iCalendar Files

The way in which Sunbird structures these events and tasks strongly mirrors the vCalendar files in which they are stored. Although you might expect a modern Internet standard to use XML, iCalendar's file format consists of name-value pairs separated by a colon (:). Each event or task has its own begin or end line, and the entire calendar file similarly is nested between overall begin and end lines. Normally, each name-value pair in an iCalendar file sits on a single line. However, indenting a line with any whitespace means that it continues the data from the previous line, as in:

```
name:value
name2:
 value2
name3
 :value3
```

The above example defines three name-value pairs, each making slightly different use of whitespace. Sunbird normally uses the third option, such that each name is on a line by itself with its associated value indented on a subsequent line. Sunbird, as with other Mozilla products, puts all of its data files in a profile directory whose name is created randomly when you first start the program. The iCalendar files themselves are placed in the Calendar subdirectory within the profile directory.

The beauty of iCalendar is you aren't expected to have all of the calendar data in one file or even on one computer. An iCalendar-compliant program displays the union of all calendar data from all of the data files it has been instructed to read. You thus can have several different calendar files on your own computer, each of which reflects a different aspect of your life, for example, personal vs. professional. You also can retrieve calendar data files from other sources, including over HTTP, meaning that group calen-

dars can be stored on a public server but displayed on your own computer.

When you first start Sunbird or when you first create a calendar with it, the program creates a CalendarDataFile.ics file. If you have more than one calendar, you end up with a number of such files on your system. Each file has the name CalendarDataFileN.ics, where N represents the number of the calendar you have created.

The structure of the file itself is pretty simple. For example, here is an iCalendar file with a single event, namely this month's *Linux Journal* deadline:

```
BEGIN:VCALENDAR
VERSION
 :2.0
PRODID
 :-//Mozilla.org/NONSGML Mozilla Calendar V1.0//EN
BEGIN:VEVENT
UID
 :05e55cc2-1dd2-11b2-8818-f578cbb4b77d
SUMMARY
 :LJ deadline
STATUS
 :TENTATIVE
CLASS
 :PRIVATE
X-MOZILLA-ALARM-DEFAULT-LENGTH
 :0
DTSTART
 :20050211T140000
DTEND
 :20050211T150000
DTSTAMP
 :20050209T132231Z
END:VEVENT
END:VCALENDAR
```

As you can see, the file begins and ends with VCALENDAR declarations. Each event is surrounded by BEGIN:VEVENT and END:VEVENT. Each event then has a unique ID; a summary, which normally is displayed in the calendar; a status; a class, which indicates whether you want to share this calendar information with others; and then the starting and ending times. It also has a timestamp showing when the event was last modified.

Timestamps in iCalendar files adhere to a slightly strange format of YYYYMMDD representing the date and then a T followed by the 24-hour clock time, followed by an optional time zone and a Z. Because I currently am living in Chicago, the timestamp represents not the time at which I made the entry, but the time it is in the time zone six hours ahead of me, one hour later than GMT (1Z).

What happens if I have a monthly deadline, and I want to include that in this calendar event? In Sunbird, I can go into the recurrence tab in the event editor by double-clicking on the event. There, I indicate that I want this event to repeat once every month, which changes the interface such that I'm now asked if it should be on the 11th of every month—that is, on the same date—or on the second Friday of every month, relative to the month. If I choose the first,

my iCalendar file looks like this:

```
BEGIN:VCALENDAR
VERSION
 :2.0
PRODID
 :-//Mozilla.org/NONSGML Mozilla Calendar V1.0//EN
BEGIN:VEVENT
UID
 :05e55cc2-1dd2-11b2-8818-f578cbb4b77d
SUMMARY
 :LJ deadline
STATUS
 :TENTATIVE
CLASS
 :PRIVATE
X-MOZILLA-ALARM-DEFAULT-LENGTH
 :0
X-MOZILLA-RECUR-DEFAULT-UNITS
 :months
RRULE
 :FREQ=MONTHLY;INTERVAL=1
DTSTART
 :20050211T140000
DTEND
 :20050211T150000
DTSTAMP
 :20050211T132231Z
LAST-MODIFIED
 :20050211T153505Z
END:VEVENT
END:VCALENDAR
```

Notice how an RRULE property has been added, with values of FREQ=MONTHLY and INTERVAL=1. You might imagine that if I were to change this deadline to be every two weeks, it would be FREQ=WEEKLY and INTERVAL=2. This is true, except that it also adds a BYDAY=FR field, indicating that the event happens on Fridays.

If I choose to make this event occur on the second Friday of each month, the iCalendar file looks like this:

```
BEGIN:VCALENDAR
VERSION
 :2.0
PRODID
 :-//Mozilla.org/NONSGML Mozilla Calendar V1.0//EN
BEGIN:VEVENT
UID
 :05e55cc2-1dd2-11b2-8818-f578cbb4b77d
SUMMARY
 :LJ deadline
STATUS
 :TENTATIVE
CLASS
 :PRIVATE
X-MOZILLA-ALARM-DEFAULT-LENGTH
 :0
X-MOZILLA-RECUR-DEFAULT-UNITS
 :months
RRULE
 :FREQ=MONTHLY;INTERVAL=1;BYDAY=2FR
DTSTART
 :20050211T140000
DTEND
 :20050211T150000
DTSTAMP
 :20050211T132231Z
LAST-MODIFIED
 :20050211T153824Z
END:VEVENT
END:VCALENDAR
```

Notice how RRULE property now is FREQ=MONTHLY, because this happens every month, with an INTERVAL=1. Also notice that BYDAY=2FR has been added, meaning that the event takes place on the second Friday of each month.

Finally, let's take advantage of Sunbird's ability to have remote calendars by moving this file out of our directory and onto another system. I move the CalendarDataFile7.ics file, which was so named because it was the seventh calendar I created, to /tmp. I then copy it to my Web site so it can be available at the URL http://reuven.lerner.co.il/CalendarDataFile7.ics. I double-check that the file is available from this URL by trying to download it using wget. When I see that it works fine, I know I can put this information into Sunbird.

Now I go into Sunbird and delete the ATF calendar; Sunbird won't let you remove the filename of an existing local calendar. I then choose subscribe to remote calendar from the File menu, and enter the URL at which I have placed my .ics file. Once the calendar has been downloaded, I see the *LJ* deadline event on my calendar each month, exactly as if it were on my local machine. And, in fact, if you look in the Calendar directory, you can see that the file *is* on your local machine. It has been downloaded and installed into that directory, and it can be refreshed whenever you request. Simply right-click on the calendar name and select reload remote calendar.

Conclusion

Our investigation of hybrid desktop-Web applications has begun with an examination of the iCalendar/vCalendar file format, using the Mozilla standalone Sunbird application. We were able to create different types of events and then move the resulting iCalendar file from our local machine to a remote server.

However, our calendar is static, meaning that someone has to modify it by hand or by uploading a calendar file each time it changes. In my next column, we will learn how to create iCalendar files dynamically, using a Web/database application. We then will look at different ways in which calendars created on one computer can be written to a server and shared with other people.

Resources for this article: www.linuxjournal.com/article/8128.

Reuven M. Lerner, a longtime Web/database consultant and developer, now is a graduate student in the Learning Sciences program at Northwestern University. His Weblog is at **altneuland.lerner.co.il**, and you can reach him at reuven@lerner.co.il.

servers DIRECT™

More products, better service, guaranteed.

GO STRAIGHT TO THE SOURCE!

SERVERS DIRECT - YOUR DIRECT SOURCE FOR SCALABLE, COST-EFFECTIVE SERVER SOLUTIONS.

1U Entry Level Server
SD1-1204i

Excellent general purpose server for organizations with the need for a low, entry level price

- Intel® Xeon™ Processor 2.8GHz 533 FSB
- 1U chassis with 350W power supply
- Supermicro X5DPA-TGM+ board
- Kingston 512MB PC2100 DDR ECC Reg. RAM (2x256MB)
- Seagate 80GB SATA 7200RPM hard drive
- 2 x 1" Hot-swap SATA drive bays
- Integrated ATI Rage XL SVGA PCI video controller
- 1x Gigabit Ethernet, 1x 10/100 Ethernet
- RAID 0, 1 Supported

$999

1U Application Server
SD1-1303S

Designed for high-performance servers clustering and business critical applications with EM64T solution.

- Intel Xeon Processor 3.0Ghz with 800FSB 1MB Cache (Dual Processor Option)
- Intel Extended Memory 64 Technology
- 1U Chassis with 420W Power Supply
- Supermicro X6DVL-EG2 board with Intel® E7320 (Lindenhurst VS) Chipset
- Kingston 512MB DDRII-400 ECC Reg. RAM (2x256MB)
- Western Digital Raptor 36Gb SATA w/10K RPM hard drive
- 2 x 1" Hot-swap SATA drive bays
- ATI RageXL 8MB Graphics
- Dual Intel® 82541GI Single-port Gigabit Ethernet Controller

$1399

2U Database Server
SD2-1205S

Cost effective and an ideal solution for high perfoming database server

- Intel Xeon Processor 3.0Ghz with 800FSB 1MB Cache (Dual Processor Option)
- Intel Extended Memory 64 Technology
- 2U Chassis with 550W power supply
- Supermicro X6DHT-G board w/Intel® E7520 (Lindenhurst) Chipset
- Kingston 512MB DDR266 ECC Reg. RAM (2x256MB)
- Zero-Channel RAID with Adaptec 2020SA (SATA) ZCR Card
- Seagate 80GB SATA 7200RPM hard drive
- 6 x 1" Hot-swap SATA Drive Bays
- ATI Rage XL SVGA PCI video controller with 8MB of video memory
- Dual Intel® 82541GI Gigabit Ethernet Controllers

$2099

3U Storage Server
SD3-1206I

Excellent scalable general use SCSI 3U server with Dual Xeon 800Mhz performance

- Intel Xeon Processor 3.0Ghz with 800FSB 1MB Cache (Dual Processor Option)
- Intel Extended Memory 64 Technology
- 3U SC833S-R760 Chassis with Triple-Redundant 760W power supply
- Supermicro X6DH8-XG2 board with Intel® E7520 (Lindenhurst) Chipset
- Kingston 512MB DDRII-400 ECC Reg. RAM (2x256MB)
- Zero-Channel RAID w/ Adaptec 2010S ZCR Card
- Seagate 36GB SCSI 10K RPM U320 SCA hard drive
- 8 x 1" hot-swap Ultra320 SCSI Drive Bays (Expandable to 15 Drives)
- ATI Rage XL SVGA PCI video controller with 8 MB of video memory
- Intel® 82546GB Dual-port Gigabit Ethernet Controllers

$2699

4U Enterprise Server
SD4-1302S

Easily scalable storage solution with hot-swap functionality for growing business

- Intel Xeon Processor 3.0Ghz with 800FSB 1MB Cache (Dual Processor Option)
- Intel Extended Memory 64 Technology
- 4U Chassis with 460W Redundant power supply
- Supermicro X6DHT-G board w/Intel® E7520 (Lindenhurst) Chipset
- Kingston 1024MB DDR266 ECC Reg. RAM (2x512MB)
- 3Ware 8port SATA Controller Card
- Western Digital 250GB 7200RPM hot-swap SATA drive with 8MB Cache
- 16x1" Hot-swap SATA drive bays
- ATI Rage XL SVGA PCI video controller with 8MB of video memory
- Dual Intel® 82541GI Gigabit Ethernet Controllers

$3099

5U Advanced Storage Server
SD5-1202S

Storage server with 24 hot-swap hard disk bays suitable for 6TB of pure data storage capacity

- Intel Xeon Processor 3.0Ghz with 800FSB 1MB Cache (Dual Processor Option)
- Intel Extended Memory 64 Technology
- 5U Chassis with 950W Tripple Redundant power supply
- Supermicro X6DHT-G board w/Intel® E7520 (Lindenhurst) Chipset
- Kingston 1024MB DDR266 ECC Reg. RAM (2x512MB)
- 3Ware 12port SATA Controller Card
- Western Digital 250GB 7200RPM hot-swap SATA drive with 8MB Cache
- 24x1" Hot-swap SATA drive bays
- ATI Rage XL SVGA PCI video controller with 8MB of video memory
- Dual Intel® 82541GI Gigabit Ethernet Controllers

$4199

PROVEN TECHNOLOGY FROM YOUR TRUSTED SERVER SOURCE

Your business requires solid server solutions. With Servers Direct server systems based on the Intel® Xeon™ Processor, you can count on high availability, maximum efficiency and proven performance to help you meet your business reliability requirements.

1.877.727.7127 | **sales@serversdirect.com**

Kprobes—A Kernel Debugger

Looking for a way to use some of the same debugging techniques in the kernel that you apply in user-space code? Here's how to bring debugging support to tricky kernel development problems.

BY R. KRISHNAKUMAR

Kprobes is a mechanism used to register breakpoints and corresponding handlers. After enabling Kprobes support in the kernel, we can debug any instruction at any kernel address. This article explains how to compile a kernel with Kprobes and how to register and unregister Kprobes, using a live example. It also covers the concept of debugging the kernel, plus internal operations of the Kprobes framework and its features.

To get started, suppose we are trying to debug a specific instruction at an address location in the kernel. Using the facilities provided by Kprobes, we can execute three functions, namely, pre-handler, post-handler and fault handler. The pre-handler function is executed before the execution of the instruction at the debugged memory location takes place. The post-handler executes after the instruction being debugged is executed. The fault handler is executed if the instruction leads to a fault.

To explain further, let's look at an example. Suppose we want to debug the instruction at location *x*. Let the instruction at location *x* be *i*. The function to be executed before *i* is executed, the pre-handler, is named pre_x. The function to be executed after the *i* is executed, the post-handler, is named post_x. The fault handler itself is fault_x.

Before *i* is executed, Kprobes runs the pre_x function. In the pre_x function we can do some necessary debugging actions, such as checking the contents of various registers and manipulating the registers. After the pre_x finishes executing, *i* executes, followed by post_x. The fault handler comes into the picture when the instruction *i* causes an operating system fault. If the fault occurs due to the execution of *i*, the fault handler, fault_x, is called.

Features

A debugging console is not necessary when using Kprobes. This is a significant design point, because it results in minimal system dependencies for operation. It therefore allows debugging to be performed at interrupt time, during context switches, when the system is disabled for interrupts and so on.

In addition, no forced serialisation of system processes is required for operation. In particular, in an SMP environment no interprocessor serialisation is required.

Another important feature of Kprobes is that data can be extracted by a probe handler and saved in a buffer. This is significant for later examination of data from a crashdump or data dumped to the console at a consistent time.

How to Enable Kprobes Support in the Kernel

After being out-of-tree patches for a long time, Kprobes finally was included in the vanilla Linux kernel. This article covers the core Kprobes functionality included as of kernel version 2.6.9. Many other features are supported by Kprobes, and they are available as patches from the Kprobes Web site (see the online Resources).

Download the vanilla kernel from www.kernel.org. While configuring the kernel, go to the Kernel Hacking submenu. Enable Kernel debugging, and then choose the Kprobes option. Compile the kernel with this configuration and boot it.

After we have enabled Kprobes, we can use various kernel APIs to register and unregister it. The function used to register Kprobe is register_kprobe. This function takes the pointer to a structure called struct kprobe. The definition of the structure is:

```
struct kprobe {
        struct hlist_node hlist;
        kprobe_opcode_t *addr;
        kprobe_pre_handler_t pre_handler;
        kprobe_post_handler_t post_handler;
        kprobe_fault_handler_t fault_handler;
        kprobe_break_handler_t break_handler;
        kprobe_opcode_t opcode;
        kprobe_opcode_t insn[MAX_INSN_SIZE];
};
```

In the struct we can specify the following:

1. The address on which Kprobe has to be set (addr).
2. The pre-handler to be executed (pre_handler).
3. The post-handler to be executed (post_handler).
4. The fault handler to be executed (fault_handler).

To unregister Kprobe, you can use unregister_kprobe, which takes the same argument as register_kprobe.

The prototype of register_kprobe and unregister_kprobe is simple:

```
int register_kprobe(struct kprobe *p);
void unregister_kprobe(struct kprobe *p);
```

You can find these definitions in include/linux/kprobes.h.

Live Action

Let's look at a real example of the process of kernel debugging using Kprobes. We begin by inserting the function we are going to debug. The code to do this is as follows, I have added the line numbers for reference:

Opteron & Xeon EM64T Blade Servers

- Lower Your Data Center Costs -

2003 Best System Integration Solution
3U Blade Server

3U- 14 Xeon Blade Server - Highest Density Available

Low-Power Consumption

4U-10 Dual Opteron or
Xeon EM64T PCI-X Blade Servers

Point to tsti.com

1U Dual Opteron/Xeon EM64T
Rackmount, IPMI 1.5 Ready

Choose from a broad line of Tatung rackmount servers.

3U and 4U Blade Servers
Superior Tatung architecture integrates up to 14 Xeon® processor server blades in each 3U chassis. The 3U blades can be configured in up to 196 servers per 42U rack. The 4U can hold up to 10 dual processor blades per chassis and 100 PCI-X servers can be configured with your choice of up to 200- 64 bit Opteron™ processors, 32bit Xeon processors or Intel's new Xeon EM64T, 64bit processors per 42U rack. Tatung blades use significantly less power than comparable systems while eliminating more than 80% of the wires.

Each hot-swappable blade is powered by the latest low-power consumption processor and has built-in dual gigabit ethernet switches and can be controlled with Tatung's advanced management software.

1U and 2U Xeon and Opteron Servers
High Performance single or dual Intel Xeon EM64T or AMD 64-bit Opteron processor rackmount servers are now shipping. Configure the 1U with up to three hard drives and a 2U with up to nine hard drives.

Tatung is building rackmount servers now. Call today and have us build a system for you.

The Intelligent Choice in Server Solutions

TATUNG®
SCIENCE & TECHNOLOGY, INC.

USA: 1-800-659-5902 • www.tsti.com
Europe: www.tatung.nl
Asia: www.tatungserver.com

```
 1 /* Filename: first.c */
 2
 3 #include <linux/module.h>
 4 #include <linux/init.h>
 5
 6 int hello_to_debug(void)
 7 {
 8         printk("\nFrom the function - %s\n",
 9                                 __FUNCTION__);
10         return 0;
11 }
12
13 static void exit_to_debug(void)
14 {
15         printk("\nModule exiting \n");
16 }
17
18 static int init_to_debug(void)
19 {
20         printk("\nKeeping the function to debug"
21                "\nat the kernel address %p\n",
22                hello_to_debug);
23         return 0;
24 }
25
26 EXPORT_SYMBOL(hello_to_debug);
27 module_init(init_to_debug);
28 module_exit(exit_to_debug);
29
30 MODULE_AUTHOR ("Krishnakumar. R,
31                <rkrishnakumar@gmail.com>");
32 MODULE_DESCRIPTION ("Kprobes test module");
33 MODULE_LICENSE("GPL");
```

Suppose we need to debug the function given in line 6, hello_to_debug. Begin by compiling the above code and insert it as a module. The EXPORT_SYMBOL directive at line 26 makes sure that the rest of the kernel code can see this function.

Now, insert Kprobe at the location to be debugged, the function hello_to_debug:

```
 1 /* Filename: kprobes.c */
 2
 3 #include <linux/module.h>
 4 #include <linux/init.h>
 5 #include <linux/kprobes.h>
 6
 7 static struct kprobe kpr;
 8 extern int hello_to_debug(void);
 9
10 static void __exit exit_probe(void)
11 {
12         printk("\nModule exiting \n");
13         unregister_kprobe(&kpr);
14 }
15
16 static int before_hook(struct kprobe *kpr,
17                        struct pt_regs *p)
18 {
19         printk("\nBefore hook");
20         printk("\nThis is the Kprobe pre \n"
21                "handler for instruction at \n"
22                "%p\n", kpr->addr);
23         printk("The registers are:\n");
24         printk("eax=%lx, ebx=%lx, ecx=%lx, \n"
25                "edx=%lx\n", p->eax,  p->ebx,
26                p->ecx,  p->edx);
27         printk("eflags=%lx, esp=%lx\n",
28                  p->eflags,  p->esp);
29         return 0;
30 }
31
32 static int after_hook(struct kprobe *kpr,
33                       struct pt_regs *p,
34                       unsigned long flags)
35 {
36         printk("\nAfter hook");
37         printk("\nThis is the Kprobe post \n"
38                "handler for instruction at"
39                " %p\n", kpr->addr);
40         printk("The registers are:\n");
41         printk("eax=%lx, ebx=%lx, ecx=%lx, \n"
42                "edx=%lx\n", p->eax,  p->ebx,
43                p->ecx,  p->edx);
44         printk("eflags=%lx, esp=%lx\n",
45                p->eflags,  p->esp);
46         return 0;
47 }
48
49 static int __init init_probe(void)
50 {
51         printk("\nInserting the kprobes \n");
52         /* Registering a kprobe */
53         kpr.pre_handler =
54             (kprobe_pre_handler_t)before_hook;
55         kpr.post_handler =
56             (kprobe_post_handler_t)after_hook;
57         kpr.addr =
58            (kprobe_opcode_t *)(&hello_to_debug);
59         printk("\nAddress where the kprobe is \n"
60                "going to be inserted - %p\n",
61                kpr.addr);
62         register_kprobe(&kpr);
63         return 0;
64 }
65
66 module_init(init_probe);
67 module_exit(exit_probe);
68
69 MODULE_AUTHOR ("Krishnakumar. R,
70                <rkrishnakumar@gmail.com>");
71 MODULE_DESCRIPTION ("Kprobes test module");
72 MODULE_LICENSE("GPL");
```

Line 57 specifies the address location where Kprobe should be set. Lines 53 and 55 specify the pre-handler and the post-handler functions, which should be activated corresponding to the address location. Line 62 registers Kprobe. So, when the above code is compiled and inserted as a module, Kprobe is registered at the hello_to_debug function. When the module is

Your corporate colors and logo

Engineered to your specifications

"From testing and design to nationwide deployment, the service from MBX has been first class."

- Ray S., Crate and Barrel

MBX is the leader in custom appliances. Many premier application developers have chosen MBX as their partner because of our experience, flexibility and accessibility. Visit our website or better yet, give us a call, our phones are personally answered by experts ready to serve you.

MBX RP-1010 LCD / Custom Branded Appliance

- Intel® Pentium 4 Processor® at 2.4E GHz
- 1U Short Depth Rackmountable Chassis
- 256MB PC2700 DDR Memory
- 512MB Solid State Flash Drive
- Dual Onboard Gigabit NIC's
- Front Panel LCD with 6 Button Control
- Custom Branded With Your Logo
- 3 Year Warranty
- Worldwide Deployment and Support
- Custom Branded Packaging Available
- Configurations in 2U and 4U Available
- Custom OS and Software Install
- Custom Chassis Color Available
- No Minimum Quantity Required

$999 or lease for **$31**/mo.

MBX™ MOTHERBOARDEXPRESS

www.mbx.com/oem

1.800.939.6191

unloaded, Kprobe is unregistered, as shown in line 13.

Now we have to invoke the function we are debugging. This is done with the following code:

```
 1 /* Filename: call.c */
 2
 3 #include <linux/module.h>
 4 #include <linux/init.h>
 5
 6 extern int hello_to_debug(void);
 7
 8 static void __exit exit_to_debug(void)
 9 {
10          printk("\nModule exiting \n");
11 }
12
13 static int __init init_to_debug(void)
14 {
15          printk("\nCalling the function \n");
16          hello_to_debug();
17          return 0;
18 }
19
20 module_init(init_to_debug);
21 module_exit(exit_to_debug);
22
23 MODULE_AUTHOR ("Krishnakumar. R,
24                 <rkrishnakumar@gmail.com>");
25 MODULE_DESCRIPTION ("Kprobes test module");
26 MODULE_LICENSE("GPL");
```

Line 16 here calls the function we are debugging. The Kprobes framework invokes the pre-handler prior to the execution of the function, and the post-handler is invoked after the execution of the instruction under debug. We then can print the register contents and Kprobe information. The following is the transcript of messages I received after compiling and inserting the above modules.

Inserting the first module:

```
[root@kk code]# /sbin/insmod first.ko

Keeping the function to debug
at the kernel address c883a000
```

Inserting the Kprobes placing module:

```
[root@kk code]# /sbin/insmod kprobes.ko

Inserting the kprobes

Address where the kprobe is
going to be inserted - c883a000
```

Calling the function under debug:

```
[root@kk code]# /sbin/insmod call.ko

Calling the function

Before hook
This is the Kprobe pre
handler for instruction at
c883a000
The registers are:
eax=17, ebx=c47ba000, ecx=c1264090,
edx=c47ba000
eflags=296, esp=c884000f

After hook
This is the Kprobe post
handler for instruction at c883a000
The registers are:
eax=17, ebx=c47ba000, ecx=c1264090,
edx=c47ba000
eflags=196, esp=c883a09e

From the function - hello_to_debug
```

Breakpoints and Debuggers

To understand better how Kprobes works, we should know the general concept of breakpoints, because Kprobes makes use of the same mechanism. A breakpoint is a mechanism provided by the hardware in most processors that we can use for debugging. For now, we are going to consider the x86 architecture. The instruction set for the processor provides a breakpoint instruction, and this instruction generates a breakpoint exception. Thus, control is transferred to the breakpoint exception handler. Most debuggers use this facility.

Suppose the debugger makes use of the breakpoint mechanism to debug. If it has to debug an instruction at a particular location, it replaces the corresponding instruction with the breakpoint instruction. The breakpoint instruction then generates the exception. The debugger contains a provision to be informed whenever such an exception is generated. The debugger then takes the necessary debugging steps, such as printing out the register values and manipulating them, as well as replacing the instruction with the original instruction. After this, execution of the instruction proceeds as usual.

Pre-Handler

When we register a pre-handler, what actually happens is Kprobes replaces the instruction at the memory location with a breakpoint instruction. The instruction that was present there is saved for later reference.

The following lines from the function int register_kprobe(struct kprobe *p) in the kernel/kprobes.c do this:

```
p->opcode = *p->addr;
*p->addr = BREAKPOINT_INSTRUCTION;
```

Hence, whenever control reaches the particular location, the breakpoint exception occurs. The default breakpoint exception handler is modified by Kprobes. The modified exception handler checks whether the address has an instance of Kprobe associated with it. If there is an associated Kprobe, the exception handler executes the pre-handler. Otherwise, control is transferred to the normal breakpoint exception handler. If Kprobe is registered for that particular location, it prepares the

processor to call the post-handler, which takes over once the pre-handler has executed.

The function responsible for handling the breakpoint is listed below:

```
asmlinkage int do_int3(struct pt_regs *regs,
                       long error_code);
```

and the function that invokes the pre-handler is here:

```
static inline int kprobe_handler(struct pt_regs *regs);
```

Post-Handler

The post-handler is executed after the instruction with which we associate the probe has executed. To facilitate this, the Kprobes framework gets some help from the hardware, specifically from a processor feature called trap generation.

If we have set the trap flag of the processor, it generates a trap exception after every instruction. After the pre-handler is run, the Kprobes framework sets the trap flag. It then replaces the breakpoint instruction with the original instruction. The function that prepares for the post-handler is presented below:

```
static inline void prepare_singlestep(struct kprobe *p,
             struct pt_regs *regs);
```

After the instruction we are debugging has executed, the processor generates a trap exception. The function responsible for the exception handling of the trap generation looks like this:

```
asmlinkage void do_debug(struct pt_regs * regs,
                         long error_code);
```

and the function that does the necessary activities for the Kprobes post-handler is:

```
static inline int post_kprobe_handler(struct pt_regs *regs);
```

The post_kprobe_handler function calls the post-handler that we have registered for that particular probe.

Fault Handler

The fault handler is executed whenever a fault is generated when executing the instruction under debug. The function responsible for Kprobes' activities on faults looks like this:

```
static inline int kprobe_fault_handler(struct pt_regs *regs,
                           int trapnr);
```

This function is called under two circumstances:

1. Whenever a general protection fault occurs, do_general_protection, and we know that it has been generated by a Kprobes instruction.

2. Whenever a device-not-available fault generation occurs, and we know it has been generated by a Kprobes instruction.

In either of these cases, the fault handler can be used to discover what went wrong.

Conclusion

The Kprobes patch helps a kernel developer debug any address within the kernel. Various patches are available from the Kprobes home page, including ones for setting watch points and for debugging user address locations. With proper use, Kprobes can become a powerful weapon in any kernel developer's arsenal.

Acknowledgements

I am thankful to Richard J Moore and Andrew Morton for their valuable comments on the draft version of this article, to Manish P Fadnavis for his support and to Pramode C E, Shine Mohammed Jabbar and Chitkala Sethuraman for their feedback.

Resources for this article: www.linuxjournal.com/article/8136.

R. Krishnakumar loves to hack the Linux kernel. He works for Hewlett-Packard and received a Bachelor's in Technology from Govt. Engg. College Thrissur. His home page is located at puggy.symonds.net/~krishnakumar. You can reach him at rkrishnakumar@gmail.com.

Crossing Platforms

Marcel takes a tour of a variety of games from a classic genre. Quick! Jump! BY MARCEL GAGNÉ

Careful, François, you're going too fast! You're going to fall in...too late. Ah, *mon ami*, you have lost another life to the deep-blue sea. In fact, you have lost all four, which means it is my turn. I know we are playing a game, but we also are sampling the fare on tonight's menu, *mon ami*, testing it as part of *Chez Marcel*'s extensive quality control. Still, our guests will be here shortly, and I have yet to select a wine.

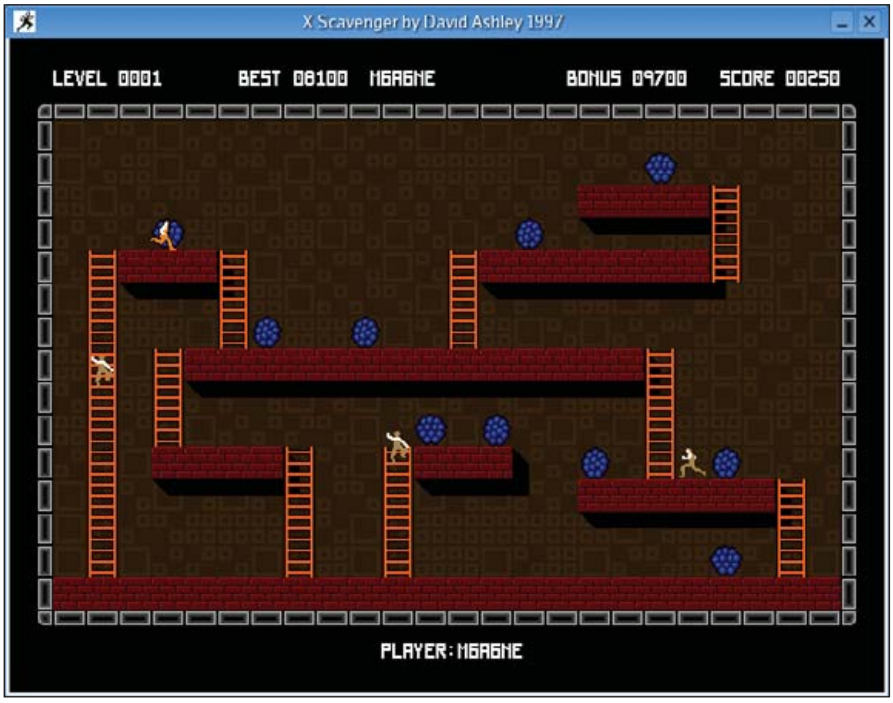


Figure 1. *Scavenger* is an excellent homage to the old *Lode Runner*.

They are here already? My apologies, *mes amis*, we are somewhat distracted. Welcome to *Chez Marcel*, where fine wines always accompany fine Linux fare. Quick, François, to the wine cellar. Given the light nature of tonight's menu, we need something that's light and easy-drinking. Bring back the 2001 Modello Italian red.

My faithful waiter and I were looking at examples of a particular type of video game often referred to as a platform game. The idea of these games is fairly simple, and most tend to be scrolling two-dimensional affairs. While moving from level to level, searching for or collecting objects as you go, your player must run, jump and avoid objects or enemies. To do so invariably involves climbing or jumping across platforms that make it possible to move from one place to another. This type of game has been around for years but continues to be popular, and new ones are created all the time. This also is true in the Linux world.

Many years ago, when this humble Chef was quite a bit younger, I was hooked on a game called *Lode Runner*. Dave Ashley's excellent *Scavenger* is similar to this classic game and is great fun to play (Figure 1). The idea is simple. You run around climbing ladders, shimmying across ropes and running from one platform to another, collecting gems while avoiding the bad guys. Depending on what is beneath your feet at the time, you also have the ability to dig holes, either for escape or to trap enemies.

Get your copy of *X Scavenger* by heading to the *Scavenger* Web site (see the on-line Resources), where you'll find the latest source. I have found that many contrib sites also offer precompiled binaries of the program, so you may want to look in your favorite distribution's offerings. That said, building *Scavenger* from source is not difficult. It's a slight variation on our classic extract-and-build process:

```
tar -xzvf xscavenger-1.4.4.tgz
cd xscavenger-1.4.4/src
xmkmf
make
su -c "make install"
```

The command name is scavenger, so that is what you have to run to start the game. It starts in demo mode, so you can watch the action and get a feel for how it works. Pressing F1 starts your game at your highest level; if this is the first time you are playing, you start at level 1.

Game controls are on the numeric keypad, though the cursor keys work for moving left, right, up and down. Two additional functions are provided for digging to the left and to the right. If you don't like the default key mapping, which is a problem on a notebook keyboard, and already are playing a game, press Esc and then press the spacebar. This takes you to the configuration menu. Now, press F10 to remap your keyboard. I left the cursor keys as is for movement, but I chose the A key to dig left and the D key to dig right.

While you are busy remapping your keyboard, you might notice some other interesting options here. For instance, you can choose to demo various levels. Pressing F7 or F8 lets you go up or down levels in the demo mode. Possibly my favorite feature, though, is

Figure 2. Your job, soldier, is to rescue the MIA blobs. Any questions?

the level editor. Simply press F3 and you can create your own *Scavenger* levels, a great way to eat up your time.

There's nothing like a nice, friendly game to whet your appetite. It is extremely tempting for me to add a smiley somewhere in the last sentence, partly due to the next platform game I want to tell you about, a game of smileys gone bad. The game is called *Blob Wars: Metal Blob Solid*, from Parallel Realities, and it's one of the stranger games I've played.

Blob Wars is a rescue-the-MIAs game with a bizarre twist. All of the characters are smileys, blobs if you will, and your task as Blob Soldier Bob is to traverse various unfriendly but occasionally nice-looking areas searching for other soldiers being held prisoner. Your job is to run your smiley soldier, firing various weapons—make sure you pick up the laser—at your opponents, jumping from platform to platform as you try to complete your mission without being destroyed (Figure 2).

Precompiled RPMs and Debian packages are available from the Parallel Realities Web site, as is a source package should you need or want to compile the program yourself. You need the various SDL development libraries, but otherwise it's quite simple, only four steps:

```
tar -xzvf blobwars-1.02-1.tar.gz
cd blobwars-1.02
make
su -c "make install"
```

To play, launch the command `blobwars`. When the game starts, you are presented with a menu from which you can start a new game or continue one already in progress. The Options menu lets you switch from windowed to full-screen mode. You also can adjust the volume of both the sound and music track, as well as the brightness level. In addition, you can decide whether you want to have the blood and gore turned on. Yes, as you fire on enemy blobs, they scream and explode into a bloody fireworks display. Sure, it's a little over the top, but the game is great fun. Really.

You then are presented with a map identifying several

Yes, as you fire on enemy blobs, they scream and explode into a bloody fireworks display.

Figure 3. Can Tux save his beloved Penny from the evil Nolok?

locations where MIAs are being held. Each location constitutes a mission. In the course of the rescue operation, a number of useful objects are there for the taking. These could be a jet pack, laser guns, grenades or keys to open doors. Sometimes you pick up these objects from fallen enemies. When you find an MIA, simply walk over to it and it is beamed away from the action. As a *Star Trek* fan from way back, I loved the transporter graphics and sound effects.

Could we possibly do justice to a menu of Linux platform games without including Tux, our favorite mascot? *SuperTux* is a classic jump-and-run platform game in the style of, you guessed it, *Super Mario Bros*. Under Tobias (Tobgle) Glaesser's lead, but originally created by Bill Kendrick, the current alpha release of *SuperTux* is guaranteed to provide you with hours of fun. Don't let that first alpha milestone fool you—this is a great game, one you absolutely need to check out.

The story goes like this: Tux and Penny (Tux's love interest) are out on a nice date when Tux is knocked out and his lovely Penny is penguinnapped by the evil Nolok. Held prisoner in Nolok's equally evil fortress, Tux must brave all sorts of perils to save his lovely lady. It is your job to help Tux succeed in his quest (Figure 3).

Binary packages for a variety of distributions and platforms are available at the *SuperTux* Web site (see Resources), so there may be no need to build from source. Should you decide to go that route, though, building *SuperTux* is a classic extract-and-build five-step:

```
tar -xjvf supertux-0.1.2.tar.bz2
cd supertux-0.1.2
./configure
make
su -c "make install"
```

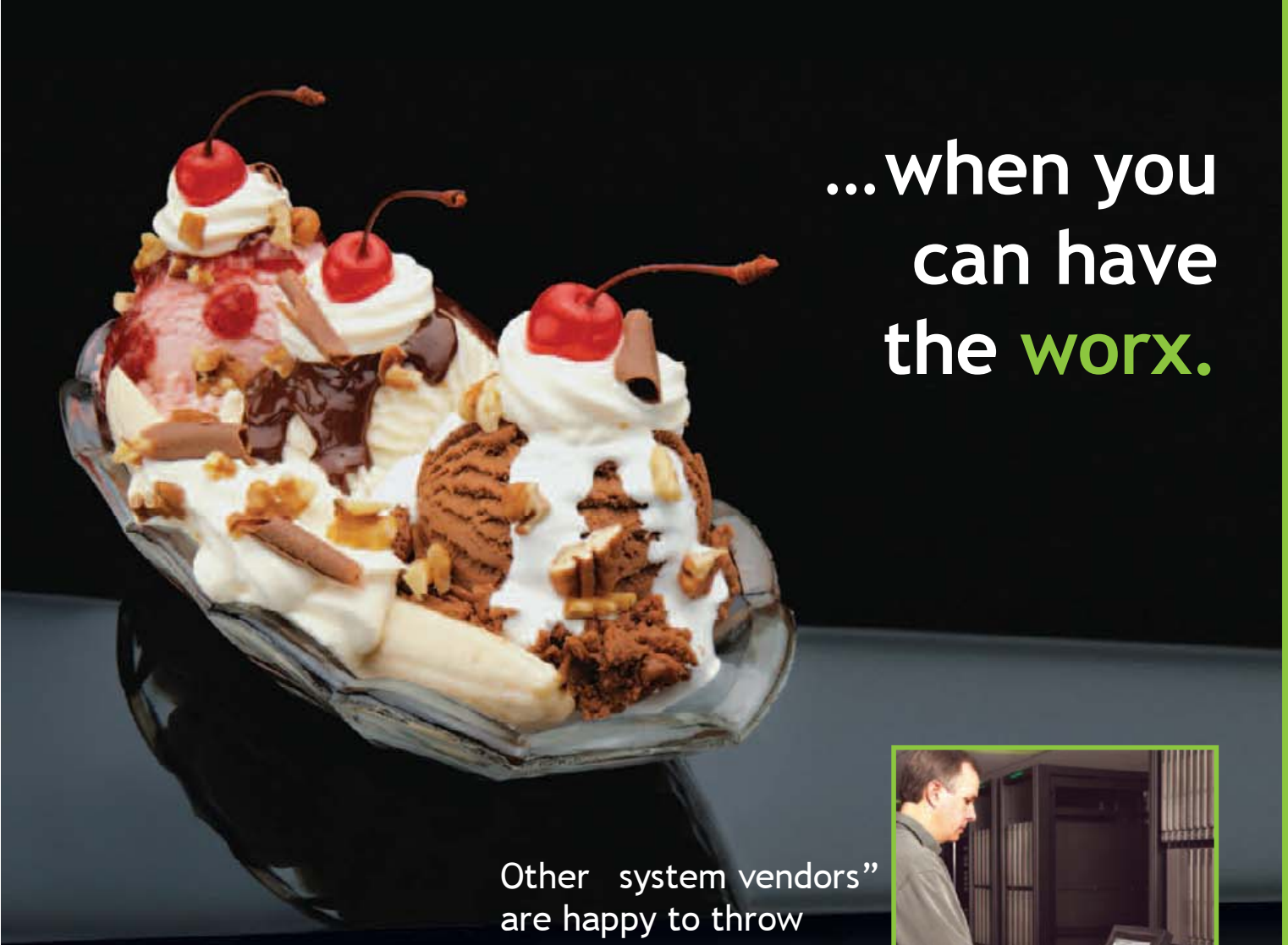
The game can be played entirely with cursor keys, but joysticks and gamepads also are supported. Start the program by running the `supertux` command. The program starts with a handful of choices, such as jumping right into the game, loading a couple of bonus levels or creating your own, a little something for later. You also can set various options, including OpenGL support, sound and music settings and so on.

The play is fast and fun. Jump over the many enemies to avoid them. Objects are at different levels, so you climb or jump onto platforms to get from one obstacle to another. Collect gold coins as you go. Smash ice blocks with Tux's head to discover power flowers or ice balls that transform him into SuperTux, a being of enhanced strength and power! When you complete a level, you are transported to the next, more complicated level. I'm still on level four as I write this.

Oh, no! Knocked out by that sliding ice block again. I think it's definitely time for a refill, François. Given the theme of this last game, I think you should serve the Baked Alaska now. I also see that closing time is nearly upon us, but have no fear, *mes amis*. We will keep the doors open and the wine flowing. There is no hurry. Besides, the Baked Alaska needs to be eaten and Penny needs to be saved before François and I can close the restaurant for the night. Therefore, *mes amis*, let us all drink to one another's health. *A votre santé! Bon appétit!*

Resources for this article: www.linuxjournal.com/article/8129.

Marcel Gagné is an award-winning writer living in Mississauga, Ontario. He is the author of the all-new *Moving to the Linux Business Desktop* (ISBN 0-131-42192-1), his third book from Addison-Wesley. He also is a pilot, was a Top-40 disc jockey, writes science fiction and fantasy and folds a mean Origami T-Rex. He can be reached at mggagne@salmar.com. You can discover a lot of other things, including great WINE links, from his Web site at www.marcelgagne.com.



...when you can have the worx.

Other "system vendors" are happy to throw a processor in a box and call it good.

Not Linux Networx.

Our cluster systems are more than plain vanilla. That's why each Linux Networx system is engineered with our Active Cooling™ hardware technology, Total Cluster Management™ tools, Xilo™ scalable cluster storage, application optimization, and Certified Cluster Services.

The result is a high productivity computing experience unlike any other.

Linux Networx®

PROVEN

An important ingredient of any cluster system is the processing technology.

AMD 64 Opteron™

AMD Opteron™ processors provide a highly scalable architecture and support large memory addressability to deliver next-generation performance as well as a flexible upgrade path from 32- to 64-bit computing.

AMD

tech article

To learn how our cluster computing products can help your business and receive a free technical report on high productivity computing, visit www.linuxnetworx.com/theworxlj

Securing Your WLAN with WPA and FreeRADIUS, Part II

The new generation of security for wireless networks doesn't simply fix WEP's flaws—it enables you to use your RADIUS server to authenticate wireless users. BY MICK BAUER

Last month, I described the new wireless LAN security protocol, Wi-Fi Protected Access (WPA). I showed how it adds strong and flexible authentication, plus dynamic encryption-key negotiation, to the otherwise-insecure WEP protocol. I also showed how WPA's component protocols, including 802.1x, the various flavors of EAP and RADIUS, interrelate. In this month's column I start to show how to create your own authentication server for WPA and other 802.1x scenarios, using FreeRADIUS and OpenSSL.

Quick Review

WPA, you may recall, is more modular than WEP. Whereas authentication and encryption in WEP both are achieved through a shared secret used by all wireless clients, authentication in WPA normally is achieved by using the 802.1x protocol. The pre-shared key (PSK) mode, which works more like WEP, also is an option. With WPA, unique encryption keys for each client are generated dynamically and refreshed periodically by the access point.

802.1x is a flexible authentication protocol that depends on the Extensible Authentication Protocol (EAP). Many different flavors of EAP, including EAP-TLS and PEAP, are supported in WPA-enabled products. If you choose to skip 802.1x and deploy WPA in the much simpler PSK mode, which gives you dynamic encryption key generation but exposes authentication credentials by transmitting them in clear text, all you need to do is configure your access point and wireless clients with the same pre-shared key.

If, however, you want to use WPA to its full potential by employing the much-stronger authentication mechanisms in 802.1x, you need a RADIUS server. Commercial tools are available for this work, such as Funk Software's Steel Belted RADIUS. But if you prefer a free and open-source RADIUS application, FreeRADIUS supports all major flavors of EAP and is both stable and secure. Here's how you can put it to work.

Our Usage Scenario

Naturally, I don't have enough space to describe all possible uses of FreeRADIUS with 802.1x or even specifically with wireless scenarios. Therefore, let's start with a description of an example usage scenario that subsequent procedures can implement.

The most important choice to make when implementing WPA is which flavor of EAP to use. This is limited not only by what your RADIUS server software supports but also by your client platforms. Your wireless access point, interestingly, is EAP-agnostic—assuming it supports 802.1x and/or WPA in the first place. It simply passes EAP traffic from clients to servers, without requiring explicit support for any particular EAP subtype.

What your client platform supports is a function both of your client operating system and of its wireless hardware. For example, a Microsoft Windows XP system with an Intel Pro/2100 (Centrino) chipset supports EAP-TLS and PEAP, but EAP-TTLS isn't an option. But if you run Linux with wpa_supplicant (see the on-line Resources), you have a much wider range of choices available.

In our example scenario, I cover EAP-TLS. EAP-TLS require client certificates, which in turn require you to set up a certificate authority (CA). But there are several good reasons to use EAP-TLS. First, EAP-TLS is supported widely. Second, TLS (X.509 certificate) authentication provides strong security. Third, it really doesn't require that much work to use OpenSSL to create your own CA.

Our example scenario, therefore, involves Windows XP clients using EAP-TLS to connect to a WPA-enabled access point. The access point, in turn, is configured to authenticate off of a FreeRADIUS 1.0.1 server running Linux.

Getting and Installing FreeRADIUS

SUSE 9.2, Fedora Core 3 and Red Hat Enterprise Linux each has its own FreeRADIUS RPM package, called freeradius. Debian Sarge (Debian-testing) has a DEB package by the same name. With Red Hat, Fedora and Debian-testing, additional packages are available if you want to use a MySQL authentication database. In addition, Debian-testing has a few other features broken out into still more packages. With all four distributions, however, the only package you should need for 802.1x authentication is the base freeradius package. If your favorite Linux distribution doesn't have its own FreeRADIUS package, or if it does but not a recent enough version to meet your needs, you can download the latest FreeRADIUS source code from the Web site (see Resources).

Compiling FreeRADIUS is simple: it's the common `./configure && make && make install` routine. If you're new to the compiling game, see the source distribution's INSTALL file for more detailed instructions. You should execute the `configure` and `make` commands as some non-root user and execute only `make install` as root.

Notice that, by default, the configure script installs

Designed for dependability.

The Intel® Xeon™ Processor offers enterprise-proven reliability, plus performance features like Processor redundancy, enhanced server management and higher bandwidth network redundancy. ZT Servers with the Intel Xeon Processor can keep your business up and running –and productive. All day, every day.

ZT OPTIMUM 5U Storage Server X9444

Intel® Xeon™ Processor 3.20 GHz
- Upgradeable to Dual Intel® Xeon™ 3.60 GHz 2MB L2 Cache Processor 800MHz FSB Processor
Intel® E7520 chipset Server Board
1.0GB ECC Registered DDR333 SDRAM (Up to 16GB)
24 x Seagate® 300GB SATA/150 7200RPM Hard Drive (Total 7.2 TB Storage)
24 x 1" SATA Hot-Swappable Drive Bays
Dual 12 Port 3ware 24 Channel Serial ATA Controller (RAID 0, 1, 5, 10 , JBOD Support)
1.44MB Floppy Drive
16x DVD±RW & CD-RW Combo Drive
ATI Rage XL SVGA 8MB PCI Graphics
Dual Intel® 10/100/1000 Gigabit Network controller
5U Rackmount Server Chassis w/950W Redundant Power Supply
3-Year Limited Warranty

$9,299

- 1-year or 3-year onsite service available
- Please call for latest pricing and discounts on quantity orders
- Your Ultimate Solutions Provider
- Aggressive Price
- VARs are welcome !

ZT Optimum 1U Server X9440

Intel® Xeon™ Processor 3.20 GHz
- Upgradeable to Dual Intel® Xeon Processor 3.60 GHz 2MB L2 Cache 800MHz FSB
Intel® SE7520JR2 (E7520 Chipset) Server Board
1.0GB ECC Registered DDR2-400MHz SDRAM (2 x 512MB)
2 x Seagate® 73GB Ultra320 10,000RPM SCSI Hard Drive (RAID 1 CONFIGURATION)
3 x Hot-swap Ultra320 SCA 1" SCSI Bays
Slim Floppy Drive & Slim CD-ROM
Integrated Dual Channel Ultra320 SCSI Controller
ATI Rage XL 8MB PCI Graphics Controller
2 x Intel® 10/100/1000 Gigabit Network Controller
SR1400NA 1U Rackmount Chassis w/500W Power Supply
3-Year Limited Warranty

$2,349

ZT Optimum 1U Server X9441

Intel® Xeon™ Processor 3 GHz
- Upgradeable to Dual Intel® Xeon Processor 3.60 GHz 2MB L2 Cache 800MHz FSB
Intel® E7320 chipset Server Board
1.0GB ECC Registered DDR333 SDRAM (Up to 8GB)
2 x Seagate® 200GB SATA/150 (8MB Cache) 7,200RPM Hard Drive (RAID 1 CONFIGURATION)
2 x Hot-swap Serial ATA/150 Bays
1.44MB Floppy Drive
52x32x52 CD-RW & 16x DVD-ROM Combo Drive
ATI Rage XL 8MB PCI Graphics Controller
Intel® 10/100/1000 Gigabit Network Controller
1U Rackmount Chassis w/350W Power Supply
3-Year Limited Warranty

$1,329

ZT Optimum 1U Server X6683

Intel® Pentium® 4 Processor 530 with HT Technology (3 GHz, 1MB L2 Cache, 800MHz)
- Upgradeable to Intel® Pentium® 4 Processor 3.60 GHz, 800MHz FSB Processor
Intel® E7221 Chipset Server Board
512MB DDR2 533MHz SDRAM
2 x Seagate® 200GB SATA/150 7,200RPM Hard Drive (RAID 1 CONFIGURATION)
1.44MB Floppy Drive
16x DVD-ROM Drive
Integrated High Quality Server Graphics
2 x 10/100/1000 Gigabit Network Controller
1U Rackmount Chassis W/350W Power Supply
3-Year Limited Warranty

$999

ZT Optimum 3U Storage Server X9443

Intel® Xeon™ Processor 3 GHz
- Upgradeable to Dual Intel® Xeon Processor 3.60 GHz 2MB L2 Cache 800MHz FSB
Intel® E7520 chipset Server Board
1.0GB ECC Registered DDR333 SDRAM (Up to 16GB)
15 x Seagate® 300GB SATA/150 7200RPM Hard Drive (Total 4.5 TB Storage)
15 x 1" SATA Hot-Swappable Drive Bays
2 x 3Ware 8 Ports 16 Channel Serial ATA Controller (RAID 0, 1, 5, 10 , JBOD Support)
ATI Rage XL SVGA 8MB PCI Graphics
2 x Intel® 10/100/1000 Gigabit Network Controller
3U Rackmount Server Chassis w/760W Triple-Redundant Power Supply
3-Year Limited Warranty

$5,699

(984-7687)

866-ZTGROUP

http://www.ztgroup.com/go/linuxjournal

Please contact our system specialists for any customize systems, latest pricing & quantity orders via email to Shopper@ztgroup.com or call (866)984-7687

Special Discount for Education, Government & Corporate

Quality Assured • 24x7 available • Lifetime Tech Support

*** Price subject to change without notice**

FreeRADIUS into subdirectories of /usr/local. Because the Makefile has no uninstall action, I recommend leaving this setting unchanged, as it simplifies removing FreeRADIUS later, should that become necessary.

Creating a Certificate Authority

Before we configure FreeRADIUS, we need to create some certificates. And before we create any certificates, we must create our CA. My book *Linux Server Security* contains a section in Chapter 5 titled "How to Become a Small-Time CA", which goes into more depth than I can go into right now, but here's a crash course nonetheless.

First, what is a CA and where should it reside? A CA is a system that acts as the root of a public key infrastructure. It's the central authority that vouches, by way of digital signatures, for the authenticity of all certificates issued in your organization. It also periodically issues certificate revocation lists (CRLs), lists of certificates the CA no longer vouches for, for example, certificates issued to people who've left the organization, servers that are no longer on-line and so on.

None of this requires your CA to act as an actual server; in fact, it's better if it doesn't. For a CA to be trustworthy, it must be protected carefully from misuse. My own CAs, therefore, tend to reside on systems I only periodically connect to the network, such as VMware virtual machines.

You already may have a CA that you've used to create certificates for Web servers, stunnel or other applications that use TLS. If so, you can use it for WPA too. If not, here's how to create a CA. First, make sure your designated CA system has OpenSSL installed. OpenSSL is a standard package on all popular Linux distributions, not to mention FreeBSD, OpenBSD and the like. One quick way to make sure you have OpenSSL is to issue the command `which openssl`—this returns the path to your OpenSSL command, if it's installed.

Next, change your working directory to wherever your system keeps OpenSSL's configuration and certificate files. On SUSE, this is /etc/ssl, but this location varies by distribution. Doing a search for the file openssl.cnf should bring you to the correct place.

Now, open the file openssl.cnf with your text editor of choice. We need to tweak some default settings to make certificate creation speedier later on. Listing 1 shows the lines in openssl.cnf I like to change.

Next, we should edit the CA creation script to change our CA's root directory to something other than demoCA, that is, to match the dir variable we just changed in openssl.cnf. I use the script CA.sh, which on SuSE systems is located in /usr/share/ssl/misc but may reside elsewhere on your system. The line you need to change is `CATOP=./micksCA`.

If you changed your working directory to edit this file, change back to your SSL configuration directory, for example, /etc/ssl. From there, run the CA.sh script with the -newca option, for example, `/usr/share/ssl/misc/CA.sh -newca`. You then are prompted to create a new root certificate and to type a passphrase for its private key. Choose a difficult-to-guess passphrase, and write it down in a safe place—if you forget it, you'll be unable to use your CA.

After the script is done, your SSL configuration directory should contain a new directory, micksCA in our example. At the root level of this directory is your new CA's public certificate; by default this file is named cacert.pem. As I demonstrate later, you need to copy this file to your FreeRADIUS server and to each wireless client.

There's one more thing you need to do before creating certificates if you've got Windows XP wireless clients. Windows XP expects certain attributes in server and client certificates, so you need to create a file called xpextensions that contains the lines shown in Listing 2.

The xpextensions file is referenced in some of the OpenSSL commands I'm about to show you. It should reside in the same directory as openssl.cnf.

Listing 1. Changes to openssl.cnf for Optimal Certificate Creation

```
# First we change the CA root path in the CA_default
# section to reflect the CA we're about to create

[ CA_default ]
dir             = ./micksCA               # Where everything is kept

# The following lines are further down in openssl.cnf:

countryName_default             = US
stateOrProvinceName_default     = Minnesota
0.organizationName_default      = Industrial Wiremonkeys of the World
```

Listing 2. Contents of xpextensions

```
[ xpclient_ext]
extendedKeyUsage = 1.3.6.1.5.5.7.3.2

[ xpserver_ext ]
extendedKeyUsage = 1.3.6.1.5.5.7.3.1
```

How EAP-TLS Works

In EAP-TLS, a wireless client and your RADIUS server mutually authenticate each other. They present each other with their respective certificates and cryptographically verify that those certificates were signed by your organization's certificate authority. In some ways, this is an elegant and simple way to handle authentication. After you install the CA's public certificate on the FreeRADIUS server, you don't need to configure any other client information explicitly, such as user names, passwords and so on.

That doesn't mean EAP-TLS is less work than user name-password schemes, however. You still need to use OpenSSL to create certificates for all your users and copy those certificates over to them. You also need to ensure that everyone has a copy of the root CA certificate installed in the proper place.

Why buy more Servers when all you need is more Disks?

Linux + Disks + Ethernet = EtherDrive®

Disks go inside servers... right? If you run out of disk space you get another server... right? Well, that used to be the case, but not any more. Now you can expand the disk space on any server with EtherDrive Storage Blades.

EtherDrive Storage Blades are simple and easy to use. And the best part is, you already know how. An EtherDrive Storage Blade is a disk drive mounted on a very small server attached directly to your network. Each blade is a nanoserver, with firmware that puts the disk's storage right on your network. No IP addresses. Just disks on the network accessible by your servers.

Just Disk Drives on Ethernet

The open protocol, ATA-over-Ethernet, allows the most in flexibility and operation simplicity. Since EtherDrive blades just look like local disks, you already know how to use them. Use any file system software. Use any RAID software, or Coraid's open source RAIDBlade appliance. Use any volume management software. It's all up to you to decide how to organize your disks. And, since the protocol is open, you know everything about how it works. The protocol is simple, only 8 pages. The open source device driver means you never have to look at the protocol. But isn't it good to know you can?

Complete Control

You have complete control over the contents of the disk. EtherDrive doesn't store anything on your disks that you don't want. You can take a disk from a running system, install it on an EtherDrive Storage Blade, and mount it. That means you are always in control. No data reformatting. No captive data. Just disk drives on the network. You never have to worry about getting your data off of an EtherDrive Blade if it fails. Just mount the disk on a system or another Blade and you're back in business.

EtherDrive Storage Blades insert into a shelf of 10 slots. Using 400GB ATA disks you can have 4TB in one 3U rack space. You can add up to 4,095 shelves on a single network. That means you can have servers sharing more than 16 Petabytes.....Imagine that.

40,000 Disks on Your Servers

A system that can go from a couple of disks, all the way to 40,000 disks, in whatever increment you want. That's probably more than you'll ever need, but isn't that the idea of scalability? Since our shelves mount in simple relay racks, just like your switches, you never run out of room. Never have your data captive inside one server's chassis. Never have to fork lift obsolete systems. Never have to buy more servers when all you need is more disks.

Processing Power with Each Blade

EtherDrive Storage Blades can go fast, too. Since each blade has its own cpu, memory and Ethernet interface, they can all work independently or in unison. Striping software can read/write blades in parallel. The wider the stripe, the faster the I/O.

Each Blade isn't limited to a single server, either. Many servers can access the same group of EtherDrive Storage Blades. They can share read-only file systems or use available software like Red Hat's GFS to share read/write file systems.

Using EtherDrive Storage Blades you only add pennies to the cost of the raw storage. **Less than $0.65 per Gigabyte.**

CORAID

www.coraid.com
info@coraid.com
1-877-548-7200

Creating Certificates

For EAP-TLS, you need at least two certificates besides your CA certificate, a server certificate for your FreeRADIUS server and one client certificate for each wireless client on your network. Creating certificates is a three-step process:

1. Generate a signing request, that is, an unsigned certificate.

2. Sign the signing request with your CA key.

3. Copy the signed certificate to the host on which it will be used.

Let's start by creating a server certificate signing request using OpenSSL's req command:

```
$ openssl req -new -nodes -keyout server_key.pem \
-out server_req.pem -days 730 -config ./openssl.cnf
```

This command creates the files server_req.pem, which contains the actual request—an unsigned certificate—and server_key.pem, its passphrase-less private key. First, though, you are prompted for your organization's Country Code, State and so on, much of which can use the default values you tweaked in openssl.conf. Pay special attention, however, to Common Name. When prompted for this, type the fully qualified domain name of your server, for example, server.wiremonkeys.org.

Next, let's use our CA key to sign the request by using OpenSSL's ca command:

```
$ openssl ca -config ./openssl.cnf \
-policy policy_anything -out server_cert.pem \
-extensions xpserver_ext -extfile ./xpextensions \
-infiles ./server_req.pem
```

This command reads the file server_req.pem and, after prompting for your CA key's passphrase, saves a signed version of it plus its corresponding private key to the file server_cert.pem. Notice the -extensions and -extfile options—this is why earlier we created the file xpextensions.

Open your signed certificate with the text editor of your choice and delete everything before the line `-----BEGIN CERTIFICATE-----`. Concatenate it and your key into a single file, like this:

```
$ cat server_key.pem server.cert.pem > \
server_keycert.pem
```

Now we've got a server certificate with a key that we can copy over to our FreeRADIUS server. Its private key isn't password-protected, however, so be sure to delete any extraneous copies after you've got it in place.

Now we need to create a client certificate signing request. The OpenSSL command to do this is similar to that used to create server certificates:

```
$ openssl req -new -keyout client_key.pem \
-out client_req.pem -days 730 -config ./openssl.cnf
```

As you can see, we're writing our signing request and key to the files client_req.pem and client_key, respectively. Unlike with the server signing requests, however, we're omitting the -nodes option. Therefore, when you run this command, you are prompted for a passphrase with which the certificate's private key can be encrypted.

Next we sign the client certificate's signing request:

```
$ openssl ca -config ./openssl.cnf \
-policy policy_anything -out client_cert.pem \
-extensions xpclient_ext -extfile ./xpextensions \
-infiles ./client_req.pem
```

Again, this is similar to the equivalent command for our server, except this time the -extensions command references a different entry in xpextensions. Also, if your clients run Linux, you should delete the extraneous stuff in the certificate, like you did with server_cert.pem. You then either can leave the certificate and key files separate or concatenate them. From there, copy your client certificate file(s) to your Linux client system.

If your certificate is to be used by a Windows XP client, you have one more step to take. You need to convert the certificate file(s) to a PKCS12-format file, with this command:

```
openssl pkcs12 -export -in client_cert.pem \
-inkey client_key.pem -out client_cert.p12 -clcerts
```

You are prompted for client_key.pem's passphrase and then for a new passphrase for the new file; you can use the same password as before if you like. You may be tempted simply to press Enter instead, especially given that the WPA supplicant in Windows XP works only when you store its certificates without passphrases. It's very, very bad practice, however, to move private keys around networks unprotected, so I strongly recommend that you not remove the passphrase until after this file is copied safely over to your Windows XP client.

Lest you be tempted to take this opportunity to bash Microsoft, I must note that on Linux both Xsupplicant and wpa_supplicant require you either to use a blank passphrase or store the passphrase in clear text in a configuration file. This is contrary to good certificate-handling wisdom. I hope we some day see WPA supplicants intelligent enough to prompt the user for its certificate passphrase on startup.

The resulting file, in this example client_cert.p12, contains both your signed certificate and its private key. Copy it to your Windows XP client system.

Conclusion

We've installed FreeRADIUS, created a certificate authority, generated server and client certificates and transferred them to their respective hosts. But we're not done yet. We still need to configure FreeRADIUS, our access point and our wireless clients. We'll do all that next time. Until then, be safe!

Resources for this article: www.linuxjournal.com/article/8134.

Mick Bauer, CISSP, is *Linux Journal*'s security editor and an IS security consultant in Minneapolis, Minnesota. O'Reilly & Associates recently released the second edition of his book *Linux Server Security* (January 2005). Mick also composes industrial polka music but has the good taste seldom to perform it.

Appro *HyperBlade* AMD Opteron™ Clusters

Designed for High-Density, High-Performance Technical Computing Applications

- Powerful, modular and scalable architecture
- Blade servers fit all cabinet sizes
- Dual AMD Opteron™ processors per node
- Up to 16GB ECC DDR memory per node
- One PCI-X slot per node
- Up to two internal HDDs
- Remote Management Options
- High-Speed Interconnect Options: Myrinet, Dolphin, Quadrics and Infiniband
- Support for Windows® and Linux Operating Systems

For an overview of how to plan and build a high-performance Linux cluster, get a white paper, sponsored by Appro at www.appro.com.

Appro 1U & 4U - Quad AMD Opteron™ Servers

Cutting-Edge Performance

Appro 1142H : Power-to-Density Server

- Quad AMD Opteron™ Processors
- Up to 32GB ECC DDR 333/400 RAM
- Up to two 500GB IDE, SATA or 292 SCSI
- Slim FDD and DVD / CDRW
- Support for Windows® and Linux Operating Systems

Appro 4148H : Multi-Purpose Server

- Quad AMD Opteron™ Processors
- Up to 32GB ECC DDR 333/400 RAM
- Up to eight 2TB SATA or 1.2TB SCSI
- Four full size PCI-X slots
- 3.5" FDD & 5.25" bay for CD / DVD-ROM or tape devices
- Support for Windows® and Linux Operating Systems

AMD Opteron™ Processors - Integrated AMD HyperTransport™ technology allows for concurrent multiple processors in a single system.
- Shorten run-time cycles and increase bandwidth for processing computing requests.
- 32 bit applications while you migrate to 64 bit computing for long-term investment protection.

APPRO
HPC Cluster Solutions

Appro has everything you need to create a network Blade cluster-ready.
On-site maintenance and installations services are available.

For more information, please visit www.appro.com
or call Appro Sales at 800.927.5464, 408.941.8100.

AMD 64 Opteron™

Things You Should Never Do in the Kernel

How do you read and write files from a kernel module? Wait, make that: "How would you read and write files from a kernel module if that weren't a bad thing to do?" BY GREG KROAH-HARTMAN

On Linux kernel programming mailing lists oriented toward new developers (see the on-line Resources), a number of common questions are asked. Almost every time one of these questions is asked, the response always is, "Don't do that!", leaving the bewildered questioner wondering what kind of strange development community they have stumbled into. This is the first in an occasional series of articles that attempts to explain why it generally is not a good idea to do these kinds of things. Then, in order to make up for the chastising, we break all of the rules and show you exactly how to do them anyway.

Read a File

The most common question asked in this don't-do-that category is, "How do I read a file from within my kernel module?" Most new kernel developers are coming from user-space programming environments or other operating systems where reading a file is a natural and essential part of bringing configuration information into a program. From within the Linux kernel, however, reading data out of a file for configuration information is considered to be forbidden. This is due to a vast array of different problems that could result if a developer tries to do this.

The most common problem is interpreting the data. Writing a file interpreter from within the kernel is a process ripe for problems, and any errors in that interpreter can cause devastating crashes. Also, any errors in the interpreter could cause buffer overflows. These might allow unprivileged users to take over a machine or get access to protected data, such as password files.

Trying to protect the kernel from dumb programming errors is not the most important reason for not allowing drivers to read files. The biggest issue is policy. Linux kernel programmers try to flee from the word policy as fast as they can. They almost never want to force the kernel to force a policy on to user space that can possibly be avoided. Having a module read a file from a filesystem at a specific location forces the policy of the location of that file to be set. If a Linux distributor decides the easiest way to handle all configuration files for the system is to place them in the /var/black/hole/of/configs, this kernel module has to be modified to support this change. This is unacceptable to the Linux Kernel community.

Another big issue with trying to read a file from within the kernel is trying to figure out exactly where the file is. Linux supports filesystem namespaces, which allow every process to contain its own view of the filesystem. This allows some programs to see only portions of the entire filesystem, while others see the filesystem in different locations. This is a powerful feature, and trying to determine that your module lives in the proper filesystem namespace is an impossible task.

If these big issues are not enough, the final problem of how to get the configuration into the kernel is also a policy decision. By forcing the kernel module to read a file every time, the author is forcing that decision. However, some distributions might decide it is better to store system configurations in a local database and have helper programs funnel that data into the kernel at the proper time. Or, they might want to connect to an external machine in some manner to determine the proper configuration at that moment. Whatever method the user decides to employ to store configuration data, by forcing it to be in a specific file, he or she is forcing that policy decision on the user, which is a bad idea.

But How Do I Configure Things?

After finally understanding the Linux kernel programmer's aversion to policy decisions and thinking that those idealists are out of their mind, you still are left with the real problem of how to get configuration data into a kernel module. How can this be done without incurring the wrath of an angry e-mail flame war?

A common way of sending data to a specific kernel module is to use a char device and the ioctl system call. This allows the author to send almost any kind of data to the kernel, with the user-space program sending the data at the proper time in the initialization process. The ioctl command, however, has been determined to have a lot of nasty side effects, and creating new ioctls in the kernel generally is frowned on. Also, trying properly to handle a 32-bit user-space program making an ioctl call into a 64-bit kernel and converting all of the data types in the correct manner is a horrible task to undertake.

Because ioctls are not allowed, the /proc filesystem can be used to get configuration data into the kernel. By writing data to a file in the filesystem created by the kernel module, the kernel module has direct access to it. Recently, though, the proc filesystem has been clamped down on by the kernel developers, as it was horribly abused by programmers over time to contain almost any type of data. Slowly this filesystem is being cleaned up to contain only process information, such as the names of filesystem states.

For a more structured filesystem, the sysfs filesystem provides a way for any device and any driver to create files to which configuration data may be sent. This interface is preferred over ioctls and using /proc. See previous articles in this column for how to create and use sysfs files within a kernel module.

I Want to Do This Anyway

Now that you understand the reasoning behind forbidding the ability to read a file from a kernel module, you of course can skip the rest of this article. It does not concern you, as you are off busily converting your kernel module to use sysfs.

Still here? Okay, so you still want to know how to read a file from a kernel module, and no amount of persuading can

convince you otherwise. You promise never to try to do this in code that will be submitted for inclusion into the main kernel tree and that I never described how to do this, right?

Actually, reading a file is quite simple, once one minor issue is resolved. A number of the kernel system calls are exported for module use; these system calls start with sys_. So, for the read system call, the function sys_read should be used.

The common approach to reading a file is to try code that looks like the following:

```
fd = sys_open(filename, O_RDONLY, 0);
if (fd >= 0) {
  /* read the file here */
  sys_close(fd);
}
```

However, when this is tried within a kernel module, the sys_open() call usually returns the error -EFAULT. This causes the author to post the question to a mailing list, which elicits the "don't read a file from the kernel" response described above.

The main thing the author forgot to take into consideration is the kernel expects the pointer passed to the sys_open() function call to be coming from user space. So, it makes a check of the pointer to verify it is in the proper address space in order to try to convert it to a kernel pointer that the rest of the kernel can use. So, when we are trying to pass a kernel pointer to the function, the error -EFAULT occurs.

Fixing the Address Space

To handle this address space mismatch, use the functions get_fs() and set_fs(). These functions modify the current process address limits to whatever the caller wants. In the case of sys_open(), we want to tell the kernel that pointers from within the kernel address space are safe, so we call:

```
set_fs(KERNEL_DS);
```

The only two valid options for the set_fs() function are KERNEL_DS and USER_DS, roughly standing for kernel data segment and user data segment, respectively.

To determine what the current address limits are before modifying them, call the get_fs() function. Then, when the kernel module is done abusing the kernel API, it can restore the proper address limits.

So, with this knowledge, the proper way to write the above code snippet is:

```
old_fs = get_fs();
set_fs(KERNEL_DS);

fd = sys_open(filename, O_RDONLY, 0);
if (fd >= 0) {
  /* read the file here */
  sys_close(fd);
}
set_fs(old_fs);
```

An example of an entire module that reads the file /etc/shadow and dumps it out to the kernel system log, proving that this can be a dangerous thing to do, can be seen below:

```
#include <linux/kernel.h>
#include <linux/init.h>
#include <linux/module.h>
#include <linux/syscalls.h>
#include <linux/fcntl.h>
#include <asm/uaccess.h>

static void read_file(char *filename)
{
  int fd;
  char buf[1];

  mm_segment_t old_fs = get_fs();
  set_fs(KERNEL_DS);

  fd = sys_open(filename, O_RDONLY, 0);
  if (fd >= 0) {
    printk(KERN_DEBUG);
    while (sys_read(fd, buf, 1) == 1)
      printk("%c", buf[0]);
    printk("\n");
```

```
    sys_close(fd);
  }
  set_fs(old_fs);
}

static int __init init(void)
{
  read_file("/etc/shadow");
  return 0;
}

static void __exit exit(void)
{ }

MODULE_LICENSE("GPL");
module_init(init);
module_exit(exit);
```

But What about Writing?

Now, armed with this newfound knowledge of how to abuse the kernel system call API and annoy a kernel programmer at the drop of a hat, you really can push your luck and write to a file from within the kernel. Fire up your favorite editor, and pound out something like the following:

```
old_fs = get_fs();
set_fs(KERNEL_DS);

fd = sys_open(filename, O_WRONLY|O_CREAT, 0644);
if (fd >= 0) {
  sys_write(data, strlen(data);
  sys_close(fd);
}
set_fs(old_fs);
```

The code seems to build properly, with no compile time warnings, but when you try to load the module, you get this odd error:

```
insmod: error inserting 'evil.ko': -1 Unknown symbol in module
```

This means that a symbol your module is trying to use has not been exported and is not available in the kernel. By looking at the kernel log, you can determine what symbol that is:

```
evil: Unknown symbol sys_write
```

So, even though the function sys_write is present in the syscalls.h header file, it is not exported for use in a kernel module. Actually, on three different platforms this symbol is exported, but who really uses a parisc architecture anyway? To work around this, we need to take advantage of the kernel functions that are available to kernel modules. By reading the code of how the sys_write function is implemented, the lack of the exported symbol can be thwarted. The following kernel module shows how this can be done by not using the sys_write call:

```
#include <linux/kernel.h>
#include <linux/init.h>
#include <linux/module.h>
#include <linux/syscalls.h>
#include <linux/file.h>
#include <linux/fs.h>
#include <linux/fcntl.h>
#include <asm/uaccess.h>

static void write_file(char *filename, char *data)
{
  struct file *file;
  loff_t pos = 0;
  int fd;

  mm_segment_t old_fs = get_fs();
  set_fs(KERNEL_DS);

  fd = sys_open(filename, O_WRONLY|O_CREAT, 0644);
  if (fd >= 0) {
    file = fget(fd);
    if (file) {
      vfs_write(file, data, strlen(data), &pos);
      fput(file);
    }
    sys_close(fd);
  }
  set_fs(old_fs);
}

static int __init init(void)
{
  write_file("/tmp/test", "Evil file.\n");
  return 0;
}

static void __exit exit(void)
{ }

MODULE_LICENSE("GPL");
module_init(init);
module_exit(exit);
```

As you can see, by using the functions fget, fput and vfs_write, we can implement our own sys_write functionality.

I Never Told You about This

In conclusion, reading and writing a file from within the kernel is a bad, bad thing to do. Never do it. Ever. Both modules from this article, along with a Makefile for compiling them, are available from the *Linux Journal* FTP site, but we expect to see no downloads in the logs. And, I never told you how to do it either. You picked it up from someone else, who learned it from his sister's best friend, who heard about how to do it from her coworker.

Resources for this article: www.linuxjournal.com/article/8130.

Greg Kroah-Hartman is one of the authors of *Linux Device Drivers, 3rd edition* and is the kernel maintainer for more driver subsystems than he likes to admit. He works for SUSE Labs, doing various kernel-specific things and can be reached at greg@kroah.com for issues unrelated to this article.

Network backup seems insurmountable if you don't have the right solution.

Arkeia Backup and Recovery. The Right Solution.

A proven, reliable solution. Arkeia pioneered professional network backup software for Linux. Today over 100,000 networks and 4,000 customers depend on Arkeia's data protection because it supports a wide array of professional environments and architectures-including new 64-bit architectures. As a result, Arkeia delivers robust and highly reliable data protection solutions for Linux and mixed environments.

Arkeia is fast. Our innovative multi-flow and multiplexing technologies dramatically increase backup speeds for large file servers—five times faster for 10,000 files within a single directory and up to 50 times faster for 150,000 files!

Since Arkeia software processes up to 200 data streams simultaneously, you can now complete operations within required backup windows.

Arkeia is highly scalable. Our modular architecture easily scales for different network sizes, operating systems and technologies-from a simple two-client configuration to a multi-site enterprise. As a result, our software easily keeps pace with your growing storage and data protection needs.

Arkeia offers the options you want. IT managers told us the options they want most, including:

- D2D (disk-to-disk) backup
- Bare-metal disaster recovery
- NDMP support for NAS backup
- Hot backup plug-ins for open applications and databases

Plus a lot more-all at a price you'll appreciate.

Try Arkeia's backup and recovery solutions for 30 days. FREE!

The best way to prove that Arkeia is the right solution for you is to let you try it—FREE—for 30 days. We'll even include free installation tech support. Simply download the demo version at www.arkeia.com/download.

arkeia®
ENTERPRISE BACKUP SOLUTIONS

wwwarkeia.com

L'Inspired

How can desktop and laptop hardware vendors become profitable again? Linspire suggests grabbing some of the desktop software business.

BY DOC SEARLS

After years gunning up and down the runway, it looks like Linux on the laptop finally may take off. I sensed some lift under the wings just in the first few months of this year. We reviewed HP's first Linux laptop in January 2005, and we're told more are on the way. Novell, which last year made a public commitment to running their entire company on Linux, made that commitment visible at LinuxWorld Expo in February 2005. A large percentage of the computers in the Novell booth were IBM ThinkPads in the same T40 family as my own (an EmperorLinux Toucan). By the time I left, the Novell folks had upgraded my SUSE 9.1 to the company's own SUSE 9.2-based Novell Desktop. I'm still shaking down a few minor glitches, but overall it works remarkably well. As a constant traveler who connects to the Net by ad hoc Wi-Fi, I'm in love (no pun intended) with Robert Love's netapplet, which comes standard with the Novell Linux Desktop. See the description and screenshot in this issue's Upfront section.

At Apachecon last fall, I detected a drop in the percentage of Apple Mac OS X PowerBooks, clearly the preferred box the year prior, and a rise in the percentage of Linux laptops. The consensus, among those I talked to about it, was that Linux was getting better for laptops. Power control with the 2.6 kernel got some of the credit, as did a growing assortment of available device drivers. And, so did a maturing portfolio of applications that offer straight-up alternatives to familiar goods on OS X and Microsoft Windows—or, better yet, application bridges to Linux. OpenOffice.org is an obvious example, but there are others, such as Nvu, the Web-authoring tool I'm using right now.

But the drag persists on the hardware side, where we're still repurposing Windows laptops for Linux. It's not going to be easy to break free of that. The reasons are economic, not technical.

Ever since "PC Compatible" became "Designed for Windows", development of desktops and laptops at all the big hardware OEMs has started with Microsoft, not anybody else. Certainly not Linux. Every laptop "designed for Windows XP" bears Microsoft branding as bold as the maker's own. This is branding of the same literal sort that ranchers practice when they burn their symbols on the hides of cattle. (In fact, cattle ranching is where the "branding" concept came from.)

Branding agreements are just the obvious side of the Windows Laptop Story. The invisible side is much more interesting. For example, I'd been told before—always on a not-for-attribution basis—that Microsoft has a long-standing policy of using "marketing dollars" to keep hardware OEMs profitable. Of course, it's been easy for Microsoft to do that, because their margins have always been huge. But it's not something anybody on either side of the arrangement would be eager to talk about.

One CEO who's not in one of those arrangements is Michael Robertson, founder and CEO of Linspire. I talked with Michael this February at the Desktop Summit, which Linspire hosts, in San Diego. "There are all sorts of economic ties involved here that aren't readily transparent", he said. "They have the OEMs strung out on marketing dollars and kickbacks and stuff like that. Marketing dollars make their business barely viable. We had a top-ten desktop guy come to us and say, 'We really want to do Linux, but we're concerned that if we do this—whether it's allowed or not—it'll upset Microsoft and we'll lose these revenues.'"

But the PC hardware business is becoming more unprofitable by the day. "Look at IBM-Lenovo", Robertson said. IBM agreed to sell its PC business to Lenovo, the People's Republic of China's largest PC builder, at the end of last year. "IBM has lost $33 on every PC they've sold", Robertson said. "We're to a breaking point where some of the OEMs out there are saying, 'Enough is enough. I have to get at least $50 better economics.' What can they do? Manufacture in China? Use cheaper parts? They already do that. They need cheaper software. Same with HP. If the PC business was doing well, Carly would still have a job. What they need now is a piece of the software business."

It isn't easy for any company, even a desperate one, to leave money on the table. But that's exactly what Linspire is asking the hardware OEMs to do. "They have to really believe that the Linux opportunity is sufficiently large to trump that guaranteed check they get from Microsoft", Robertson said.

So far, none of the OEMs has budged. At least not in a way anybody's ready to talk about. Meanwhile, Linspire plugs away. "It's a case of moving up the ecosystem", Robertson said. "We work with VIA, the motherboard company. They're happy to work with us. That domino fell. Then the AMD domino. They're happy to work with us too."

I talked with folks at the AMD and VIA booths at the Desktop Summit, and it was clear that both companies savored the freedom and opportunity that come with opting not to run their goods through the Microsoft mill. Their attitude was much the same as I saw from the makers of Linux-based home media centers at CES one month earlier (see "The No-Party System", in the April 2005 issue of *Linux Journal*).

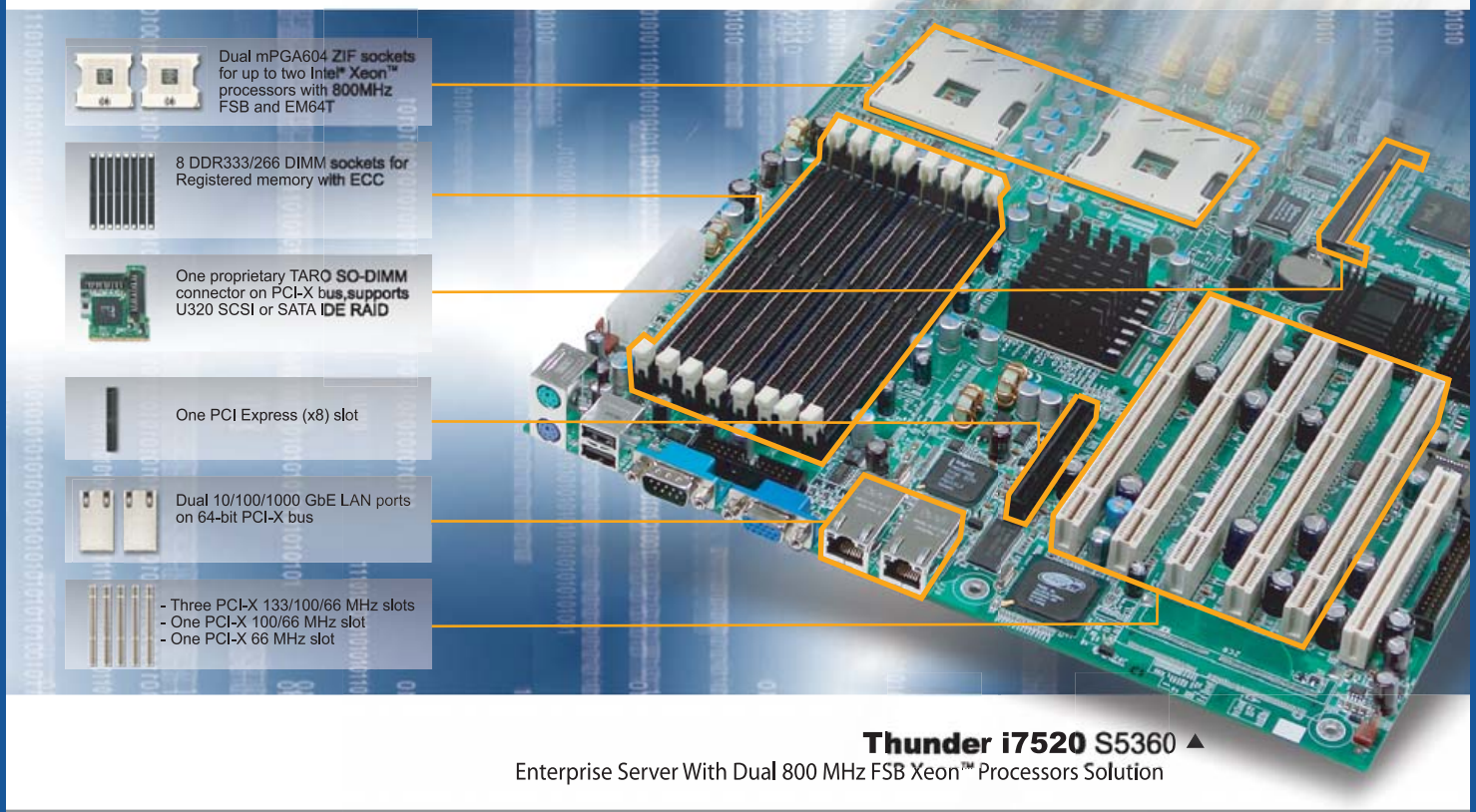
I asked Michael Robertson about NVIDIA and ATI, the big graphics subsystem companies. He smiled and said, "Other dominoes will fall too. The last will be the major record labels. The clue phone has to be ringing so loud they have no choice but to answer it."

Robertson has a lot of experience in the record business. He founded MP3.com and got rich by selling it to a record

TYAN

Fast, Flexible, and Feature-Rich!

PCI Express and EM64T Servers Have Arrived



Thunder i7520 S5360 ▲
Enterprise Server With Dual 800 MHz FSB Xeon™ Processors Solution

intel inside XEON

Thunder i7520 S5360

- Supports two Intel® Xeon™ processors with 800 MHz FSB and EM64T
- 8 DIMMs for DDR266/333 memory
- Three PCI-X 133/100/66 MHz slots, one PCI-X 100/66 MHz slot, one PCI-X 66 MHz slot and one 33MHz PCI slot
- One PCI Express™ x8 slot
- One proprietary SO-DIMM connector on PCI-X bus, supports U320 SCSI or SATA
- Dual GbE LAN

Tiger i7320 S5350

- Supports two Intel® Xeon™ Processor with 800MHz FSB and EM64T
- 8 DIMMs for DDR266/333 memory
- Two PCI-X 64/66 MHz slots; three 32/33 PCI 2.3 slots
- One proprietary SO-DIMM connector on PCI-X bus, supports U320 SCSI or SATA
- Dual PCI Express GbE LAN

TYAN COMPUTER CORP.

Tyan Computer USA
3288 Laurelview Court
Fremont, CA 94538 USA
Tel: +1-510-651-8868 Fax: +1-510-651-7688
Pre-Sales Tel: +1-510-651-8868 x5120
Email: marketing@tyan.com

For more information about this and other Tyan products, please contact Tyan Pre-Sales at (510) 651-8868 x5120, or contact your local Tyan system integrator/reseller.

www.tyan.com

company that went on to kill it. Now he's back in the business again, with a companion to Linspire called MP3tunes. But where Linspire goes straight after Microsoft's business, MP3tunes goes straight after Apple's.

Like Apple's iTunes, MP3tunes is a store where you can sample and buy music. Unlike iTunes, MP3tunes sells music without any digital rights management (DRM). Where Apple sells music encoded at 128kb AAC, MP3tunes sells music encoded at 198kb MP3. Where Apple sells songs at $.99 each and $9.99 per album, MP3tunes' prices are $.88 and $8.88. And where Apple's store lives in an application that runs only on Windows and OS X, MP3tunes lives in a wide-open Web site, in addition to Linspire's Lsongs (which runs on Linux and works like Apple's iTunes). Michael Robertson: "Does the world need another music store? Yes. Because most of them are rental shops. They control what you can do with it, what you can copy it to. I think if you pay money for music, you should be able to use it any way you want, on any device....I like the model where, if you pay for the music, it's yours. Forever."

MP3tunes also distributes music from CD Baby, a company that leaves all rights with the artist and takes only a 9% distribution fee. Customers also get an MP3tunes "music locker" that remembers everything bought from the service. This allows the customers to download the same songs again, as many times as they like, to any device.

Naturally, Robertson sees Apple as no less controlling toward the music business than Microsoft is toward the operating system business:

> It's not iTunes. It's theirTunes. Apple's controlling whether you can play it on this portable device, whether you can copy it to a Linux computer—which you can't, by the way. I did send an e-mail to Steve Jobs, saying "Hey Steve, how about supporting AAC and iTunes on Linux?" To give him credit, he did reply back to me...a one-word answer. It was "Nope." That's it. Not even a hi or a bye. Just "Nope."

MP3tunes is fighting Apple on the playback side as well. Where iTunes is designed to run only on iPods and the PCs to which they attach, MP3tunes supports playing any music on any device.

At the show, they demonstrated that advantage with a new product called MP3beamer. It's a Linux-based music storage system that distributes music over Wi-Fi. You can buy the software on-line for $69.95 ($10 more in a box). Or, you can buy hardware with the software pre-installed. The demo hardware at the show was a compact $399 box from sub300.com—and a Linksys Wi-Fi boom box. The Linksys comes with a display that shows the tunes being played (along with other information) and a remote control that operates the base unit through the boom box. The Linksys also has audio-out plugs you can run into a home stereo.

I also was curious at the show to see how Linspire's newest distro version worked on the VIA-based eCom notebook computer they had loaned me two years ago. Kendall Dawson, a "community liaison" specialist with Linspire, helped me out by bringing a CD drive to the show, attaching it to the notebook (it doesn't have an internal drive) and loading the Linspire 5.0 beta distro. Everything went smoothly, and the rejuvenated notebook immediately met my minimum laptop requirements: 1) it saw and connected to the Net through the Wi-Fi card in its PCMCIA slot; 2) it hibernated when the lid closed and returned to its full waking state when the lid opened; and 3) it let me obtain and install the small pile of software I wanted to run—and did it far more easily than any Linux distro I've ever used.

There were still a few small problems. It didn't recognize my digital camera's memory card when I plugged a reader in to a USB port. Same with the Flash memory stick that also serves as the laptop-side interface for a remote controller I use to operate slideshows. Then again, it's not easy to get my ThinkPad T40 running SUSE (9.1) to do the same things.

The Linspire laptop also immediately ran CNR (Click 'N Run), a software warehouse and installation system for more than 2,000 packages, most of which are free (as in beer). I logged in, it remembered me, and I have been downloading stuff ever since. The system is the first I've seen that actually does a simpler and easier job of installing software than Windows or even OS X. Because Linspire is built on Debian, it manages packages with apt-get. Earlier versions had standalone apt-get functionality, but because users run as root in Linspire, this caused problems. So apt-get functionality is commented out in 5.0. Expert users, of course, can fix that.

What encouraged me most at the show, however, wasn't on the floor. It was at a tour of Linspire's offices. At one end of a long open space cluttered with desks and geek debris was a counter populated by about 15 laptops and notebooks, each with performance charts taped to their lids. All the units were going through QA testing and burn-in on Linspire 5.0. I felt like I was watching *Invasion of the Body Snatchers*, except in this case, the invaders were the good guys.

They're making headway. The $199 Linux box that Fry's Electronics advertises almost every week comes with Linspire as the default OS. Linspire is also on a $498 laptop sold at WalMart.com. Recently I also received a confidential report about Linspire running on desktops in a large company. When I asked Michael Robertson about that, he said, "We're in a lot more companies than you'd think." But he declined to name names to spare those companies unwanted visits from Microsoft sales people.

Two years ago, at the first Summit, my only problem with Linspire was its narrow focus on end users, almost to the exclusion of the established Linux community. Since then, they've become much more friendly to the community, though I think they still could do more. For example, I'd like to see them showing up at other Linux and open-source events and working more with publications and Web sites that serve the Linux community.

The idea here is for Linspire to get adoption by the hard-core experts who read *Linux Journal*. It's to provide those experts with a version of Linux they can give to their non-expert parents and friends. If you want Linux adoption, there's no better leverage than you'll get from a Linux expert.

Doc Searls is Senior Editor of *Linux Journal*.

Introducing C3. The only computer industry event that gives you the ability to...

See

it all. C3 EXPO is the only computer industry event where corporate and channel buyers can see, at one time and in one place, the most innovative products and services driving the technology industry.

Hear

what's important, by attending the most comprehensive seminar program in the IT industry. C3 EXPO will enable you to increase your technical knowledge, expand your management skill sets and gain insights from your colleagues and respected industry professionals.

Speak

to your peers. Network with manufacturers, industry analysts, associations and consultants. C3 EXPO will give you the information you need to compare best practices and to make the most informed decisions regarding your company's technology investments.

THE RIGHT SHOW • THE RIGHT TIME • THE RIGHT PLACE

C3 CORPORATE & CHANNEL COMPUTING EXPO

The Javits Convention Center, NYC
June 28 – 30, 2005

For Attendee and Exhibitor info or Speaking opportunities visit C3EXPO.COM/LJ (Enter Source Code: LJ) or call: 201.543.5060 ext. 830

Our Valued Partners

LINUX JOURNAL

NEC • QUANTUM • NORTHROP GRUMMAN • CRN • EASTMAN KODAK • EPSON • CTIA VAR BUSINESS • INFOWORLD • LOGICUBE • MITSUBISHI DIGITAL • NORTEL • WOMEN IN TECHNOLOGY INT'L • LATINOS IN INFORMATION SCIENCES AND TECHNOLOGY ASSOC.

InfiniBand and Linux

Learn why letting a remote system on the network scribble on your memory is fine, how user-space applications can send data without bothering the kernel and more facts about the new high-performance interconnect. **BY ROLAND DREIER**

After a long gestation, use of InfiniBand (IB) is taking off, and work is under way to add IB support to Linux. At the physical level, IB is similar to PCI Express. It carries data using multiple high-speed serial lanes. The first versions of the InfiniBand specification allowed only for the same signaling rate for each lane, 2.5Gb/s, as PCI Express. The latest version of the specification (1.2), however, adds support for 5Gb/s and 10Gb/s rates per lane. Also, IB supports widths of 1X, 4X, 8X and 12X, while PCI Express supports X1, X2, X4, X8, X12, X16 and X32. The most commonly used IB speed today is 4X at a 2.5Gb/s/lane rate, or 10Gb/s total. But the 12X width combined with the 10Gb/s/lane rate means the current IB spec supports links with an astonishing 120Gb/s of throughput.

Because IB is used to build network fabrics, IB supports both copper and optical cabling, while the PCI Express cable specification still is being developed. Most IB installations use copper cable (Figure 1), which can be used for distances up to about 10 meters. IB also allows a variety of optical cabling choices, which theoretically allow for links up to 10km.

In past years, IB was pitched as a replacement for PCI, but that no longer is expected to be the case. Instead, IB adapters should continue to be peripherals that connect to systems through PCI, PCI Express, HyperTransport or a similar peripheral bus.

The network adapters used to attach systems to an IB network are called host channel adapters (HCAs). In addition to the fabric's extremely high speed, IB HCAs also provide a message passing interface that allows systems to use the 10Gb/sec or more throughput offered by InfiniBand. To make use of IB's speed, supporting zero-copy networking is key;

Table 1. Comparison Chart

Technology	Data Rate	Cables
USB	12Mb/s	5m
Hi-Speed USB (USB 2.0)	480Mb/s	5m
IEEE 1394 (FireWire)	400Mb/s	4m
Gigagbit Ethernet	1,000Mb/s	100m (cat5 cable)
10 Gigabit Ethernet	10,000Mb/s	10m (copper IB cable), 1+ km (optical)
Myrinet	2,000Mb/s	10m (copper), 200m (optical)
1X InfiniBand	2,000Mb/s	10m (copper), 1+ km (optical)
4X InfiniBand	8,000Mb/s	10m (copper), 1+ km (optical)
12X InfiniBand	24,000Mb/s	10m (copper), 1+ km (optical)

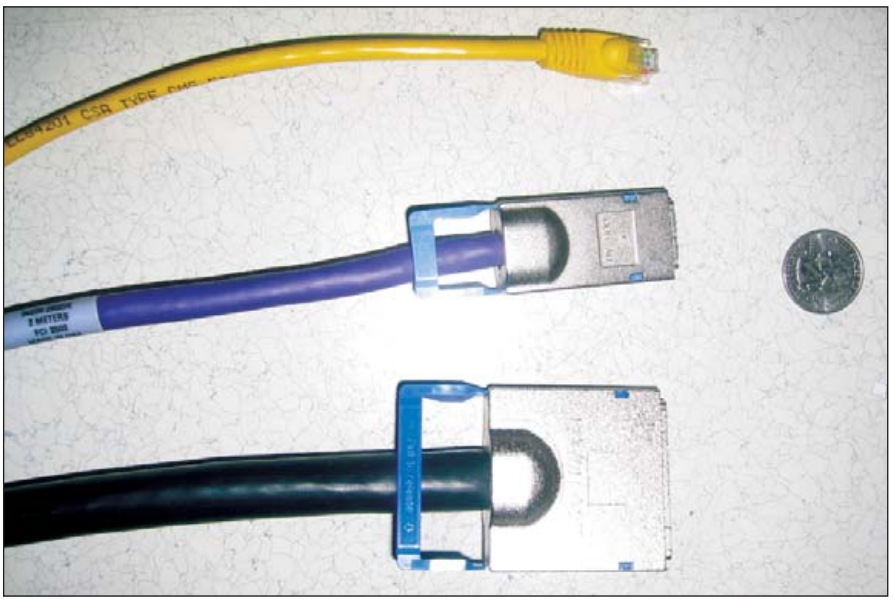


Figure 1. Top to bottom: Cat 5 Ethernet Cable, 4X InfiniBand Cable and 12X InfiniBand Cable (US quarter coin included for scale)

otherwise, applications will spend all their time copying data.

The HCA interface has three key features that make zero-copy possible: a high-level work queue abstraction, kernel bypass and remote direct memory access (RDMA). The work queue abstraction means that instead of having to construct and process network traffic packet by packet, applications post work requests to queues processed by the HCA. A message sent with a single work request can be up to 4GB long, with the HCA taking care of breaking the message into packets, waiting for acknowledgements and resending dropped packets. Because the HCA hardware takes care of delivering large messages without any involvement from the CPU, applications receive more CPU time to generate and process the data they send and receive.

Kernel bypass allows user applications to post work requests directly to and collect completion events directly from the HCA's queues, eliminating the system call overhead of switching to and from the kernel's context. A kernel driver sets up the queues, and standard memory protection is used to make sure that each process accesses only its own resources. All fast path operations, though, are done purely in user space.

The final piece, RDMA, allows messages to carry the destination address to which they should be written in memory. Specifying where data belongs is useful for applications such as serving storage over IB, where the server's reads from disk may complete out of order. Without RDMA, either the server has to waste time waiting when it has data ready to send or the client has to waste CPU power copying data to its final location.

Although the idea of remote systems scribbling on memory makes some queasy, IB allows applications to set strict address ranges and permissions for RDMA. If anything, IB RDMA is safer than letting a disk controller DMA into memory.

Beyond its high performance, IB also simplifies building and managing clusters by providing a single fabric that can carry networking and storage traffic in addition to cluster communication. Many groups have specified a wide variety of upper-level protocols that can run over IB, including:

- IP-over-InfiniBand (IPoIB): the Internet Engineering Task Force (IETF) has a working group developing standards-track drafts for sending IP traffic over IB. These drafts eventually should lead to an RFC standard for IPoIB. IPoIB does not take full advantage of IB's performance, however, as traffic still passes through the IP stack and is sent packet by packet. IPoIB does provide a simple way to run legacy applications or send control traffic over IB.

- Sockets Direct Protocol (SDP): the InfiniBand Trade Association itself has specified a protocol that maps standard socket operations onto native IB RDMA operations. This allows socket applications to run unchanged and still receive nearly all of IB's performance benefits.

- SCSI RDMA Protocol (SRP): the InterNational Committee for Information Technology Standards (INCITS) T10 committee, which is responsible for SCSI standards, has published a standard for mapping the SCSI protocol onto IB. Work is under way on developing a second-generation SRP-2 protocol.

Many other groups also are studying and specifying the use of IB, including APIs from the DAT Collaborative and the Open Group's Interconnect Software Consortium, RDMA bindings for NFS and IB support for various MPI packages.

Of course, without open-source support, all of these fancy hardware capabilities are a lot less interesting to the Linux world. Fortunately, the OpenIB Alliance is an industry consortium dedicated to producing exactly that—a complete open-source IB stack. OpenIB currently has 15 member companies, including IB hardware vendors, server companies, software companies and research organizations.

Work on the OpenIB software began in February 2004, and the first kernel drivers were merged into the main kernel tree in December 2004, right after the tree opened for 2.6.11 following the release of 2.6.10. The first batch of code merged into the kernel is the smallest set of IB drivers that do something useful. It contains a midlayer to abstract low-level hardware drivers from upper-level protocols, a single low-level driver for Mellanox HCAs, an IPoIB upper-level protocol driver and a driver to allow a subnet manager to run in user space.

A few snippets of code from the IPoIB driver should provide some understanding of how one can use the kernel's IB support. To see this code in context, you can look at the complete IPoIB driver, which is in the directory drivers/infiniband/ulp/ipoib in the Linux kernel source.

Listing 1 shows what the IPoIB driver does to allocate all of its IB resources. First, it calls ib_alloc_pd(), which allocates a protection domain (PD), an opaque container that every user of IB must have to hold other resources.

By the way, proper error checking has been omitted from the listings, although any real kernel code must check the return values of all functions for failure. All of the IB functions that allocate resources and return pointers use the standard Linux method for returning errors by way of the ERR_PTR() macro, which means that the status can be tested with IS_ERR(). For example, the call to ib_alloc_pd() in the real kernel actually looks like:

```
priv->pd = ib_alloc_pd(priv->ca);
if (IS_ERR(priv->pd)) {
        printk(KERN_WARNING "%s: failed "
               "to allocate PD\n", ca->name);
        return -ENODEV;
}
```

Next, the driver calls ib_create_cq(), which creates a completion queue (CQ). The driver requests that the function ipoib_ib_completion() be called when a completion event occurs and that the CQ be able to hold at least IPOIB_TX_RING_SIZE + IPOIB_RX_RING_SIZE + 1 work completion structures. This size is required to handle the extreme case when the driver posts its maximum number of sends and receives and then does not get to run until they all have generated completions. Confusingly enough, CQs are the one IB resource not associated with a PD, so we don't have to pass our PD to this function.

Once the CQ is created, the driver calls ib_req_notify_cq() to request that the completion event function be called for the next work completion added to the CQ. The event function, ipoib_ib_completion(), processes completions until the CQ is empty. It then repeats the call to ib_req_notify_cq() so it is called again when more completions are available.

The driver then calls ib_get_dma_mr() to set up a memory region (MR) that can be used with DMA addresses obtained from the kernel's DMA mapping API. Translation tables are set up in the IB HCA to handle this, and a local key (L_Key) is returned that can be passed back to the HCA in order to refer to this MR.

Finally, the driver calls ib_create_qp() to create a queue pair (QP). This object is called a queue pair because it consists of a pair of work queues—one queue for send requests and one queue for receive requests. Creating a QP requires filling in the fairly large ib_qp_init_attr struct. The cap structure gives the sizes of the send and receive queues that are to be created. The sq_sig_type and rq_sig_type fields are set to IB_SIGNAL_ALL_WR so that all work requests generate a completion.

The qp_type field is set to IB_QPT_UD so that an unreliable datagram (UD) QP is created. There are four possible transports for an IB QP: reliable connected (RC), reliable datagram (RD), unreliable connected (UC) and unreliable datagram (UD). For the reliable transports, the IB hardware guarantees that all messages either are delivered successfully or generate an error if an unrecoverable error, such as a cable being unplugged, occurs. For connected transports, all messages go to a single destination, which is set when the QP is set up, while datagram transports allow each message to be sent to a

Listing 1. IPoIB Driver Initialization

```
struct ib_qp_init_attr init_attr = {
    .cap = {
        .max_send_wr  = IPOIB_TX_RING_SIZE,
        .max_recv_wr  = IPOIB_RX_RING_SIZE,
        .max_send_sge = 1,
        .max_recv_sge = 1
    },
    .sq_sig_type = IB_SIGNAL_ALL_WR,
    .rq_sig_type = IB_SIGNAL_ALL_WR,
    .qp_type     = IB_QPT_UD
};

priv->pd = ib_alloc_pd(priv->ca);

priv->cq = ib_create_cq(priv->ca,
             ipoib_ib_completion,
             NULL, dev,
             IPOIB_TX_RING_SIZE +
             IPOIB_RX_RING_SIZE + 1);

if (ib_req_notify_cq(priv->cq, IB_CQ_NEXT_COMP))
    goto out_free_cq;

priv->mr = ib_get_dma_mr(priv->pd,
                         IB_ACCESS_LOCAL_WRITE);

init_attr.send_cq = priv->cq;
init_attr.recv_cq = priv->cq,

priv->qp = ib_create_qp(priv->pd, &init_attr);
```

different destination.

Once the IPoIB driver has created its QP, it uses the QP to send the packets given to it by the network stack. Listing 2 shows what is required to post a request to the send queue of the QP.

Listing 2. IPoIB Driver Send Request Posting

```
priv->tx_sge.lkey              = priv->mr->lkey;
priv->tx_sge.addr              = addr;
priv->tx_sge.length            = len;

priv->tx_wr.opcode             = IB_WR_SEND;
priv->tx_wr.sg_list            = &priv->tx_sge;
priv->tx_wr.num_sge            = 1;
priv->tx_wr.send_flags         = IB_SEND_SIGNALED;
priv->tx_wr.wr_id              = wr_id;
priv->tx_wr.wr.ud.remote_qpn   = qpn;
priv->tx_wr.wr.ud.ah           = address;

ib_post_send(priv->qp, &priv->tx_wr, &bad_wr);
```

First, the driver sets up the gather list for the send request. The lkey field is set to the L_Key of the MR that came from ib_get_dma_mr(). Because the IPoIB is sending packets that are in one contiguous chunk, the gather list has only a single entry. The driver simply has to assign the address and length of the packet. The address in the gather list is a DMA address obtained from dma_map_single() rather than a virtual address. In general, software can use a longer gather list to have the HCA collect multiple buffers into a single message to avoid having to copy data into a single buffer.

The driver then fills in the rest of the fields of the send work request. The opcode is set to `send`, sg_list and num_sge are set for the gather list just filled in and the send flags are set to `signaled` so that the work request generates a completion when it finishes. The remote QP number and address handle are set, and the wr_id field is set to the driver's work request ID.

Once the work request is filled in, the driver calls ib_post_send(), which actually adds the request to the send queue. When the request is completed by the IB hardware, a work completion is added to the driver's CQ and eventually is handled by ipoib_ib_completion().

InfiniBand can do a lot, and the OpenIB Alliance is only getting started writing software to do it all. Now that Linux has basic support for IB, we will be implementing more upper-level protocols, including SDP and storage protocols. Another major area we are tackling is support for direct user-space access to IB—the kernel bypass feature we talked about earlier. There's plenty of interesting work to be done on IB, and the OpenIB Project is open to everyone, so come join the fun.

Resources for this article: www.linuxjournal.com/article/8131.

Roland Dreier is the maintainer and lead developer for Linux InfiniBand drivers through the OpenIB.org Project. Roland received his PhD in Mathematics from the University of California at Berkeley and has held a variety of positions in academic research and high tech. He has been employed by Topspin Communications since 2001.

BELLY DANCE AND FREE SOFTWARE

Professional performers need tools for managing photos and creating attention-getting promotional materials for print and on-line. Manage the photo collection with gthumb, and lay out posters with ease using Scribus.

BY DAWN DEVINE AND MICHAEL BAXTER

Opposite page, clockwise from top left:

The Gypsy look takes a number of forms from traditional or historically flavored looks to more glamorous costumes, such as Hannah's stylish beaded version. Gypsy dancers ring the Mediterranean Sea and take many forms from the stylized Flamenco to the flirty belly dances performed in Turkey and Egypt.

Belly dance appears in a variety of venues. Dancers assemble a wardrobe of costumes to suit different performance needs. Simone wears her modern wild animal costume to performances where she can demonstrate her wit and vivacious style.

Belly dance is most often seen in intimate venues. However, there are the occasional gala shows where dancers pull out all the stops with high-style Egyptian costumes, such as the one Setareh wore on a large stage at a formal dance concert.

Tempest embodies the look and attitude of the new generation of dancers who infuse traditional belly dance style with a modern edge. Her signature style and Goth sensibilities are reflected in her highly personal costume style, which easily could be worn in a variety of venues from nightclubs to Renaissance fairs to belly dance festivals.

The San Francisco Bay Area is a hotbed of innovative technological development and a crucible for the growth of performance arts. One art that thrives in the progressive creative environment of the region is belly dance. From elegant performers draped in bead-encrusted costumes dancing in upscale restaurants to the colorful, turban-bedecked entertainers at Renaissance festivals and street fairs, for more than 100 years belly dancers have shimmied their way into every strata of contemporary pop culture.

Middle Eastern dance arrived in the United States in the last quarter of the 19th century, appearing in cultural exhibits at various world's fairs. Little Egypt was the first dancer to garner fame and prestige while performing at the 1893 World's Columbian Exhibition in Chicago. Her amazing performances were so popular that the entertainment director of the fair, impresario Sol Bloom, hyped her in advertising with his newly coined term, belly dance.

Although more formally known as Middle Eastern dance, the slang term belly dance persists, encompassing a range of styles from traditional ethnic forms, including the Egyptian Raks Sharki, Arabic for "dance of the East", to the highly stylized American Tribal Style and experimental Raks Gothique techniques. Each method is defined by its own unique blend of music, costume and movement vocabulary.

From motion pictures to MTV videos, on television sitcoms and even at local ethnic restaurants, belly dancers are highly sought-after entertainers. Therefore, working belly dancers require professional-grade IT tools to meet their publicity needs with style and panache. Audiences and potential clients have become more sophisticated, demanding a higher degree of polish and professionalism.

Gone are the days when a simple 8" × 10" black-and-white glossy photo served as a dancer's complete marketing package. Teachers need a way to get the word out about classes and performances. Professional dancers have to advertise their skills, services and show times. With the right tools, dancers can develop their own marketing packages. Today's advertising needs include business cards, flyers and Web sites.

Enter Free Software

Dancers now use computers for their own unique set of needs in marketing, music and sometimes also video. Like emergent innovation in dance in the Bay Area, GNU/Linux and other free software truly invite exploration.

This article is based on our collaborations with several Bay Area belly dancers and a digital photography work flow that is done entirely with GNU/Linux software. We describe two example successes using free software tools for belly dance marketing applications. In addition, we explore some really interesting intersections of free software and belly dance, both as an art and as a business. The sense of community, as in GNU/Linux community, has apparent parallels with the dance community. To contextualize this topic further, we talked with several professional dancers to get their take on the role of technology in their art and in their woman-owned businesses. One dancer, Michelle Joyce, says, "I really believe that my Web site is responsible for my professional dance career."

Challenges also are present in dance business promotion, where content also needs to educate. Amy Luna Manderino says of her dance group Shuvani, "Although our talents are diverse, they are all connected through Shuvani, in which we perform Romani music and dance from India, Turkey, Egypt and Spain. The biggest challenge is educating the public about the Roma (Gypsy) Trail. Many people are unaware that Gypsies are an ethnic community with a rich cultural heritage....That's always the challenge when you produce something in an artistic medium that hasn't been seen before, you have to educate people on the concept."

Free Software in Belly Dance Promotion

We use free software for everything from professional photography to document generation, as we discuss here. We have held several belly dance photo shoots at the Creative Camera professional photo studio in Santa Clara, California. This photography was conducted with a digital work flow that uses 100% free software tools. For example, we use The GIMP, gthumb and gtkam (see the on-line Resources).

Free software definitely addresses several concerns mentioned by belly dancers. For example, in talking about Web content development for her business, Tempest says "Thumbnailing, sizing and finishing images is very time consuming and can be especially tedious if the quality of the images is less than ideal." Having spent a lot of time recently batch-processing photos with gthumb, we find this gPhoto2 utility indispensable for Web and print work.

Now we turn to some creative uses of free software in belly dance with two key examples of creating belly dance concert posters. Each example is a little different. In the first, we intentionally wanted to create a new document from scratch for an upcoming event and needed documents suitable for both print and elec-

tronic media. In the second example, an event already was being promoted on the Web, but we needed a print version right away, so some novel repurposing was done with free software tools.

To create a new poster for a spring-themed belly dance event, we held a photo shoot and then edited the studio photos with gthumb and The GIMP. These resulting images were combined with other text and photo elements to create the poster. The poster itself was done on Linux using a desktop publishing (DTP) application called Scribus. Refer to Clay Dowling's article "Linux as a Publishing Platform" on the *Linux Journal* Web site for more details on using Scribus and The GIMP to publish on Linux. Although we describe only some of the big steps we used with The GIMP and Scribus, we were surprised at how easy it was to use this powerful application to position graphic elements precisely and ultimately obtain a professional result.

The first task was to select a suitable main photo of Michelle Paloma-Hudkins, who was promoting a belly dance event held in Sunnyvale, California, in March 2005. We start with gthumb, as shown in Figure 1. After looking over several photos in the series that we did, we selected one. But rather than insert a photo decorated only with a simple color border, it seemed more spring-like to use a fuzzy edge to the photo. This was managed by using The GIMP's Script-Fu decor menu in Figure 2. The output after edge manipulation is shown in Figure 3.

To sell the idea of spring, we wanted to composite into the poster a photo of daffodils from a separate photo shoot. One photo was a field of flowers, over which we intended to lay out announcement copy text. Another photo was a bright close-up of a single daffodil.

The GIMP has powerful layer manipulation capabilities, and we needed only a few of them to accomplish a lot. In Figure 4 we added a layer filled with all-white color and merged this at 50% opacity to create the slightly faded flower field image. The resulting picture is shown in Figure 5.

To assemble all of the graphical elements, we started Scribus. The easiest thing to do was to place the text onto the letter-size layout view in separate chunks, along with graphical items. These could be changed and placed using Scribus tools. An early version of the poster layout is shown in Figure 6. We used the Scribus editing panel for changing the original ASCII text, which imported with a default font, into something different and more appropriately sized for the poster (Figure 7). Other text bodies were handled similarly. This work included colorizing the text, changing the line layout and other typographic qualities.

After text placement and coloring, we added two additional photos from live belly dance events, with Simone and

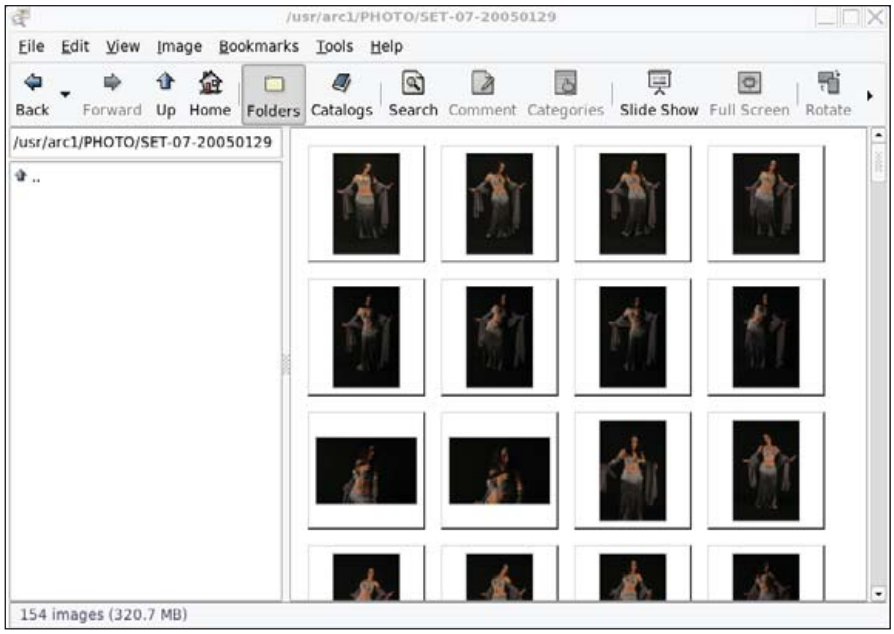


Figure 1. Viewing a photo collection with gthumb to find exactly the right photo.

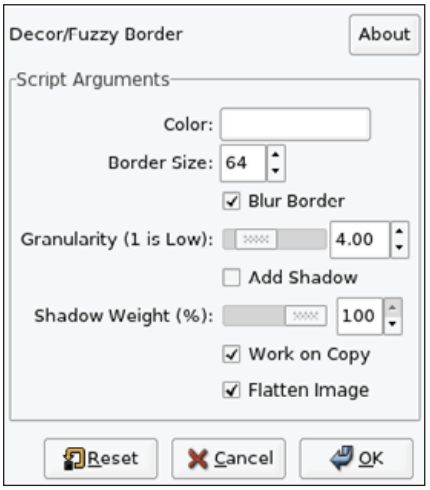


Figure 2. Using a Script-Fu menu in The GIMP, we created a fuzzy border for the chosen image.

Figure 3. The result nicely details a glamorous photo of Paloma, with the fuzzy border effect.

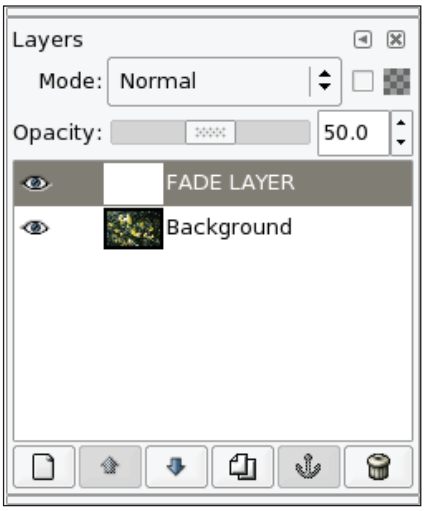


Figure 4. Using layers in The GIMP, we added a white layer to make the background photo appear faded.

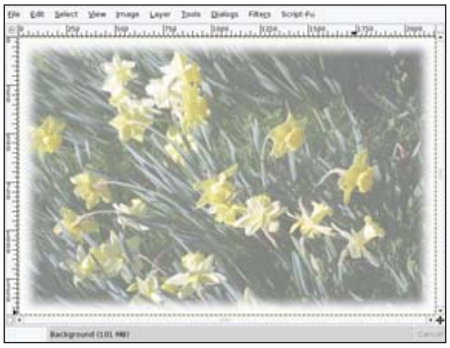


Figure 5. The faded photo is now ready to use as background art for a spring-themed event.

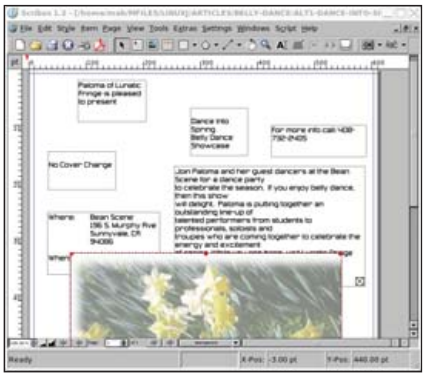


Figure 6. Working on the poster in Scribus. The background photo has been added, but the text chunks still are in the default font.

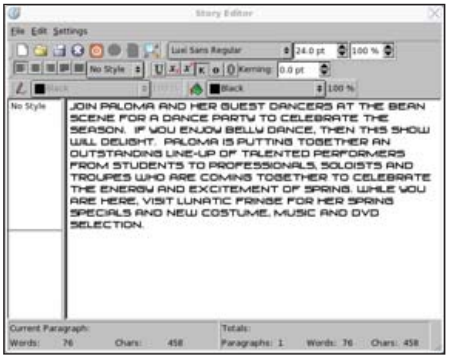


Figure 7. Working with copy in the Scribus editing panel.

Tempest at the bottom. The photograph of the single daffodil was placed in the upper left, near the poster title. With the final edits complete, the bright and engaging poster is shown in Figure 8. Using Scribus, one source document could be used simultaneously for both print applications and as an on-line PDF file for Internet distribution to the public.

Our second example solved a problem in belly dance promotion where pre-existing marketing content had to be used. With photos we'd taken at an earlier photo shoot, Paloma created a Web poster for a belly dance event held in December 2004, called Jingle Bell Raks. We now needed to create a print poster plus a separate high-resolution electronic distribution format. Free software made it possible to support Paloma's marketing concerns where "...obstacles include venues to advertise at low cost. Finding the right bulletin boards to post on to be most effective. Also finding new avenues of distribution to reach outside the Belly Dance community, to draw in the general public."

A high-quality print-oriented conversion program called HTMLDOC (see Resources) came to the rescue here in an unusual application of free software. The robust HTMLDOC program can be compiled from source with ease. What is does is amazing: it reads in HTML, images and other data and then automatically turns that content into PDF files or PostScript as output. It also has powerful book-production and indexing features. Our case was pretty simple, we wanted to create a one-page print poster and PDF file automatically. It was really easy to do, as shown in Figure 9. We simply inserted the name of a single HTML file via the HTMLDOC GUI and then configured the program to make PDF files. The resulting poster as shown with xpdf is depicted in Figure 10.

Figure 8. Final Result in Scribus: Dance into Spring

Belly Dance and Free Software

We have only started an adventure in belly dance, using free software primarily for photography and image-oriented marketing projects. But more is clearly possible. Recalling Reuven M. Lerner's *Linux Journal* articles on Web syndication and content management systems (CMSes), it would appear that free software exists to support solutions for Paloma and Tempest's comments about the complexity involved in a belly dancer trying to create and expand her Web presence. The free software tools for Linux and audio described in Dave Phillips' informative articles offer exemplary tools for the job of arranging belly dance music. The article by Olexiy Tykhomyrov and Denys Tonkonog on the Kino movie-making application (see December 2004's *Linux Journal*) points toward free software for editing digital video. Hence, it too is a potential tool for dancers to have for live belly dance events.

And still more is possible. At the time of this writing, we are preparing to do photography at a large annual belly dance event called Rakkasah West, which "is the largest Middle Eastern Folk Festival and Fantasy Bazaar in the world". We expect to create 10,000–12,000 high-resolution digital photos covering this event over three days. We plan to assemble photo collections automati-

Figure 9. HTMLDOC GUI menu box—adding one filename.

Figure 10. HTMLDOC rules! Jingle Bell Raks, as seen with xpdf.

cally using GNU/Linux tools such as gPhoto2 and the open-source language Python. These tools certainly will be used for actual photo file management. However, we also expect to use the Python scripting capability of Scribus to support automatic generation of CD-ROM disk labels and nearly full custom photo books, indexed by belly dancer and/or troupe. Stay connected with *Linux Journal* for more details about this in the future.

Finally, Linux as a community-developed project has definite resonance with the Belly Dance community. Linux is a gift culture, where numerous programmers and maintainers contribute code for all to have. Belly dancers communicate and network with one another and with clients using the Internet, which is a large community substantially woven together with an infrastructure of free software and open standards. Advocacy for freedom and social consciousness ring no less true in belly dance. As Luna explains, her troupe Shuvani "donates several performances a year to the Voice of Roma, an advocacy group providing support for the Gypsies in war-torn Kosovo. For example, through their program 'The Threads That Connect Us' they provide material support for Romani women in the UN camps to embroider products that are then sold here in the US."

Resources for this article: www.linuxjournal.com/article/8133.

Dawn Devine aka Davina is a freelance writer and author of more than nine titles on Middle Eastern belly dance costume design and construction. She currently is the president of SF/BA MECDA, the San Francisco/Bay Area Middle Eastern Dance Association, an organization that promotes dancers, creates performance opportunities and strives to educate the public. Davina teaches belly dance in the greater San Jose area and in workshops nationwide.

Michael Baxter is technical editor at *Linux Journal* and has been working in computer technology since he was nine, imprinted by a July 4, 1969, viewing of *2001: A Space Odyssey*. He is also an experienced photographer. Michael was more recently imprinted by the Belly Dance Vortex, apparently also on July 4, this time in 2004.

Fd.o: Building the Desktop in the Right Places

Don't be fooled by chatter about desktop wars. Applications and desktop environments are cooperating behind the scenes and using reality-tested standards to make everyone's software work and play well together. **BY MARCO FIORETTI**

End users care only about applications that perform the desired tasks. They come to Linux to have the freedom to pick up these applications one by one. To them, integrated desktop means the freedom to choose any mix of programs and the assurance that they work together. A monolithic desktop environment can limit programmers as well. Making sure that your code cooperates with existing applications is essential to good software, if not the main characteristic that makes it useful. Being forced to use one or two development toolchains to achieve this result makes much less sense.

A sore spot of the GNU/Linux desktop used to be XFree86—development progressed too slowly and performance was not satisfying. Many tools, from fontconfig to zlib, were duplicated to avoid external dependencies. If one driver changed, the whole package had to be released again.

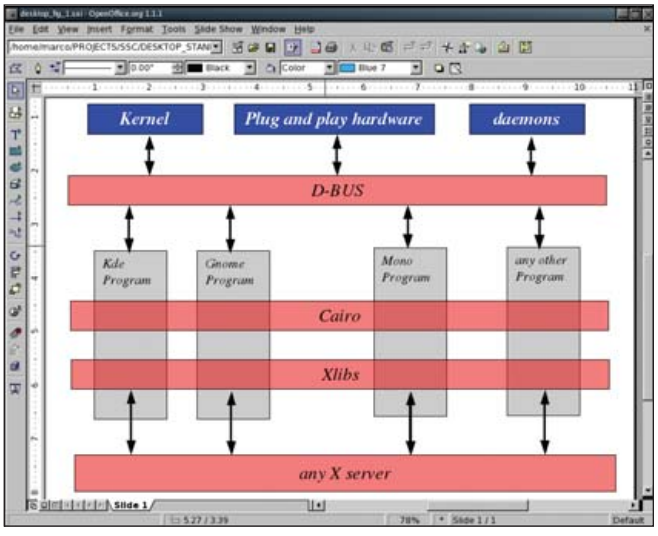


Figure 1. Integration the Freedesktop.org way: servers, libraries and communication protocols that all applications can use, no matter which desktop environment they were born in.

On top of all this, the XFree86 license changed last year to one that appeared to prohibit GPL programs from linking to any of the new code. Several distributions immediately reacted by not shipping the new version with the license problem.

Freedesktop.org (Fd.o) was formed in March 2000 to help developers solve the technical problems outlined above. The goal of this project is to create a base platform upon which every desktop can build. The method is to define independent specifications, complete with working code where needed. Formal standardization is left to other bodies. Following these specs should guarantee real interoperability among applications as early as possible during their development, ideally before it starts. All software will be placed under LGPL or X-style licenses. Fd.o hosts a lot of neat projects, but this article introduces the main tools constituting the so-called Fd.o platform.

Xlibs

The X Window System is a network transparent protocol for graphical display. GUI programs use X to give drawing commands to the software, called the X server, which actually controls the screen. Until last year, servers and libraries usually were found in monolithic packages. Fd.o broke that bundle into parts, however, which now can be developed and packaged separately. The main advantage of this is that programmers and Linux distributions can mix and customize, at will, different implementations of each piece.

Other X improvements include the removal of all in-tree dependencies and the use of autotools as the build system and of the iconv library for all conversions between Unicode and other encodings. The libraries wrapping the X protocol are called Xlibs. Fd.o released its first version of them in January 2004. They adhere to the X standard, so they can be used with any X server.

Even after several optimizations, the size of Xlibs may create problems on low-end platforms. Furthermore, some Xlibs requests block until they receive a reply, even when it is not really necessary. This can interfere with some latency reduction features in the 2.6 kernels. Xlibs also do a lot to

hide the protocol, through caching, layering and similar efforts; these efforts are an advantage in many cases, an overhead in others. Last but not least, support for the creation of X extensions is limited.

The Fd.o proposal to solve these problems is the X C Binding, XCB for short. This second library can be a base for new toolkits and lightweight emulation of parts of the Xlibs API. XCB is designed to work transparently with POSIX thread or single-thread programs. The code maintains binary compatibility with Xlibs extensions and applications and might not require recompilation of extensions. This makes slow, gradual migration from Xlibs to XCB easier, without losing functionality. The next step along this path, the Xlibs Compatibility Layer (XCL), should allow existing applications built on Xlibs to take advantage of XCB.

X Servers

Fd.o hosts two alternatives to XFree86. The first one started as a fork of the XFree86 4.4-RC2 code before the license change. This server is called X.org and is used in the same way as XFree86. The other alternative, called Xserver, is the most promising option in the long run. It is the fork of Kdrive, which started years ago as a lightweight, heavily modified version of XFree86. Kdrive is small, partly because it has less code duplication with the kernel. Size reduction also came about by removing some obsolete features and driver modules. The much smaller code size makes it easier to start from Kdrive to build a whole new server.

The version of Xserver available today still is used mainly as a test bed for new extensions and features, such as transparency or OpenGL acceleration. Memory usage is minimized by performing a lot of calculations at runtime instead of always keeping the results in memory.

The goal of Xserver is to reduce slowness as well as the other phenomena that make looking at a screen unpleasant, including flickers. A new X extension, called Composite, allows double buffering of the entire screen. Of course, no server can be smarter than its dumbest client, but the lighter architecture should make it easier to find and fix slow code, wherever it is. The new server makes no impact at the toolkit level, except when the programmer chooses to take direct advantage of the new extensions.

Cairo

Vector graphics create an image by drawing more or less complex lines and filling in the resulting areas with colors. The corresponding files are small in size and can be scaled at any resolution without losses. Consequently, this technique is interesting for users who want to be sure that what they print is what they see. Unfortunately, X knows how to manage screen pixmaps of text, rectangles and such, but it simply ignores printing or vector graphics. This is one of the reasons why we still do not have 100% consistency between screen, paper and saved files.

The Fd.o solution is Cairo, “a new 2D vector graphics library with cross-device output support”. In plain English, this means the result is the same on all output media. Externally, Cairo provides user-level APIs similar to the PDF 1.4 imaging model.

Through different back ends, Cairo can support different output devices. The first one is screens, through either Xlibs or XCB, and in-memory image buffers, which then can be saved to a file or passed to other applications. PostScript and PNG output already is possible, and PDF is planned. OpenGL accelerated output also will be available through a back end called Glitz. In addition, it will be possible to use Glitz as a standalone layer above OpenGL. Cairo language bindings exist for C++, Java, Python, Ruby and GTK+.

The developers of OpenOffice.org also are planning to use Cairo after version 2.0 of the OOo suite is released, possibly even as a separately downloadable graphics plugin. Still being in active development and minus a completely stable API, Cairo is not included yet in official Fd.o platform releases.

D-BUS

D-BUS is a binary protocol for Inter Process Communication (IPC) among the applications of one desktop session or between that session and the operating system. The second case corresponds to dynamic interactions with the user whenever new hardware or software is added to the computer. The internals of D-BUS were discussed by Robert Love in “Get on the D-BUS” in the February 2005 issue of *Linux Journal*. As far as the desktop end user is concerned, D-BUS should provide at least the same level of service currently available in both GNOME and KDE. To achieve this, both a system dæmon called message bus and a per-user, per-session dæmon are available. It also is possible for any two programs to communicate directly by using D-BUS to maximize performance. For the same reason, because the programs using the same D-BUS almost always live inside the same host, a binary format is used instead of plain XML.

The message bus dæmon is an executable acting like a router. By passing messages instead of byte streams among applications, the dæmon makes their services available to the desktop. Normally there are multiple independent instances of this dæmon on each computer. One would be used for system-level communications, with heavy security restrictions on what messages it can accept. The others would be created for each user session, to serve applications inside it. The system-wide instance of D-BUS can become a security hole: services running as root must be able to exchange information and events with user applications. For this reason, it is designed with limited privileges and runs in a chroot jail. D-BUS-specific security guidelines can be found on the Fd.o Web site (see the on-line Resources).

Most programmers do not need to talk the D-BUS protocol directly. There are wrapper libraries to use it in any desired framework or language. In this way, everybody is able to maintain his or her preferred environment rather than learning or switching to a new one specifically for IPC. End users, again, receive gains in interoperability: KDE, GNOME and Mono programs will be able to talk to one another, regardless of toolkit. As with Cairo, the first versions of the Fd.o platform don’t include D-BUS, because its API is not stabilized yet. But, the developers consider D-BUS to be a cornerstone of future releases. D-BUS also is meant to replace DCOP in KDE 4.

Is This the Right Solution?

Only time will tell if the first implementations of Fd.o are good enough and if the related specifications are valid. In this context, valid means something complete that can be re-implemented from scratch with totally new code, if one feels like doing so. I am convinced, however, that the approach is valid and has more potential than any other "complete desktop" already existing.

The two most frequent complaints I've read so far are 1) the current desktops would lose their identities, becoming "only user-interface stuff" and 2) Fd.o is not standards, simply other implementations. My personal response to the first concern is, even if it happened, would it really be a problem? Most end users wouldn't even realize it, nor would they be concerned at all. They most likely would note the improvements I mentioned at the beginning and be done with it. Making sure that all applications can cooperate, no matter how they were developed, also would make Linux much more acceptable as a corporate desktop, shutting up a whole category of arguments in favor of proprietary solutions.

If protocols and formats stop being tied to specific implementations or toolkits, they can be shared across multiple "desktop environments". Code stability and lightness would directly benefit from this, as would innovation. Completely new programs could interact immediately with existing ones. I therefore hope that this trend is generalized and that more application-independent standards are submitted to Fd.o, covering file formats, sound schemes, color and tasks settings. Imagine one mail configuration file that could be used by any e-mail client, from Evolution to mutt, or one bookmark file usable by every browser from Mozilla to Lynx.

As far as the second objection goes—Fd.o is not standards, simply other implementations—that's exactly how free software and RFCs work. As long as specifications are written alongside the code, concepts can be validated in the field as soon as possible. For the record, this same thing currently is happening with OOo and the OASIS Office standard (see *LJ*, April 2004). The file format started and matured inside StarOffice and OOo, but now it has a life of its own. The committee currently includes representatives from KOffice, and any future office suite can use it as its native format, starting only from the specification.

Some traps do exist along this path, but as far as I can tell, the developers are aware of them and determined to avoid them. The first risk is to develop standards that for one reason or another work well only on Linux, leaving out the other UNIXes. The other is resource usage: all the magic described here would look much less attractive if it required doubling the RAM to run smoothly. As far as we know today, however, this seems to be an unlikely possibility. In any case, this is the right moment to join this effort. Happy hacking!

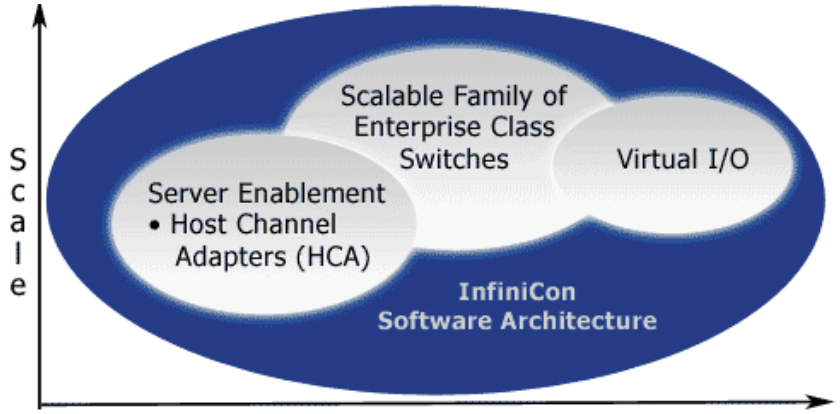
Credits

Many thanks to Waldo Bastian, Keith Packard, Daniel Stone and Sander Vesik for all their explanations.

Resources for this article: www.linuxjournal.com/article/8135.

Marco Fioretti is a hardware systems engineer interested in free software both as an EDA platform and, as the current leader of the RULE Project, as an efficient desktop. Marco lives with his family in Rome, Italy.

InfiniCon Software Release 3.0



InfiniCon Systems released version 3.0 of its architecture for the InfiniBand-based hardware and software platform. The platform includes all host software, switch embedded software and InfiniCon's FastFabric tools, opening the architecture to enable the use of third-party tools and applications. The 3.0 software can be incorporated into server architectures that embed InfiniBand on the motherboard—either on servers or on blade platforms—eliminating the need for a Host Channel Adapter (HCA) to access the InfiniBand network. Release 3.0 also includes support for Linux 2.6, scalability to more than 1,000 node fabrics, Oracle certification, certification of additional commercial MPI packages, additional fabric reliability features, performance enhancements for InfiniBand and Ethernet protocols and additions to FastFabric tools for management needs.

CONTACT InfiniCon Systems, 680 American Avenue, Suite 100, King of Prussia, Pennsylvania 19406, 610-233-4747, www.infinicon.com.

NAG Fortran Library Mark 21

Mark 21 of the NAG Fortran Library includes more than 300 new functions, taking the total to more than 1,500 functions. New functions include a complete chapter covering mesh generation that incorporates routines for generating 2-D meshes with a number of associated utility routines. Extensions have been included for zeros of polynomials, partial differential equations, eigenvalue problems (LAPACK) and sparse linear algebra. The random number generation (G05) function also has been expanded to include a new random number generator, the generation of univariate GARCH, asymmetric GARCH and EGARCH processes, quasi-random number generators and generators for further distributors. The NAG Fortran Library is available for implementations ranging from PCs to supercomputers. Not restricted to a single environment, algorithms can be called from other languages including C++.

CONTACT The Numerical Algorithms Group Ltd., Wilkinson House, Jordan Hill Road, Oxford OX2 8DR, UK, www.nag.com.

IBM eServer Application Server Advantage for Linux

The IBM eServer Application Server Advantage for Linux, also known as Chiphopper, combines support and testing tools that enable ISVs to develop cross-platform Linux products. Chiphopper is a no-charge offering that can be used to take existing Linux-on-x86 (Intel or AMD) applications and test, port and support those applications across all IBM systems. Chiphopper supports applications written directly to the operating system or written to middleware. For applications written directly to the OS, Chiphopper bases portability on the Linux Standard Base (LSB) specification. In addition, Chiphopper supports LSB applications that use open extensions including OpenLDAP, OpenSSL, Kerberos, PHP, Perl and Python. For applications using middleware, Chiphopper supports IBM's WebSphere, DB2 and Rational, providing Java, J2EE, Web services and services-oriented architecture open standards-based support.

CONTACT IBM Corporation, 1133 Westchester Avenue, White Plains, New York 10604, www-1.ibm.com/linux.

Platform for Network Equipment, Linux Edition

Wind River Systems announced the availability of Platform for Network Equipment (NE), Linux Edition. Platform NE supports the Carrier Grade Linux 2.0 specification and Linux 2.6 kernel technology for device software development. It also enables ATCA-based commercial off-the-shelf (COTS) solutions for control and management applications in carrier grade network equipment. In addition, Platform NE provides access to a wide range of third-party runtime and tool vendors, as well as the Eclipse-based Wind River Workbench IDE to support the entire development cycle.

CONTACT Wind River Systems, 500 Wind River Way, Alameda, California 94501, 800-545-9463, windriver.com.

CM4000 Console Server

A new family of console servers is available from Opengear, Inc. The CM4000 serial console server comes in 8-, 16- and 48-port versions that enable control of serial consoles on Windows, Sun and Linux servers. Opengear's CM4000 products also can monitor and control network appliances, including routers, gateways, PBXes and power switches. Remote site servers can be accessed in-band through the enterprise TCP-IP network or directly through a dial-up modem port, both using up to 128-bit AES encryption. The Opengear CM4000 console server also provides filtering and access logging facilities, enabling console logs to be archived off-line. The CM4000s are built with the okvm open-source console and KVM management software, as well as open-source KVM hardware. Both Web browser and command-line management options are available.

CONTACT Opengear, Inc., 7984 South Welby Park #101, West Jordan, Utah 84088, 801-282-1387, www.opengear.com.

Please send information about releases of Linux-related products to Heather Mead at newproducts@ssc.com or New Products c/o *Linux Journal*, PO Box 55549, Seattle, WA 98155-0549. Submissions are edited for length and content.

routerOS: routerBOARD.com

User Management

- support more than 3000 PPPoE or Hotspot clients
  - full radius support for user parameters
    - tx/rx speed, address, filter rules
    - supports radius real time modification of parameters while users are online
- Peer to peer control (P2P)
  - burst time
  - per client P2P tx/rx rules
  - P2P pool
  - complete blocking of P2P

Wireless AP and Backbone

- Wireless monitoring
  - Frequency scanning with detailed report
  - Raw wireless packet sniffer
    - streaming option to Ethereal analyzer
    - option to save to a file format supported by Ethereal
  - Snooper packet inspection
    - analyzes all raw frames received for wireless parameters
    - monitor a single channel or all channels

- Nstreme wireless polling protocol
  - no decrease in speed over long distances (as seen with the 802.11 ack packet bottleneck)
  - polling improves speed and eliminates contention for access to the wireless bandwidth
  - access point control over Nstreme clients tx data to optimize use of the wireless medium
  - radius support for the access control list including bandwidth settings for wireless clients

- Full 802.11a/b/g support

The above is a brief description of a few features, for more information and a fully featured 24 hour demo go to:

www.mikrotik.com

$95

RouterBOARD 44

For the Router Builder !

- rackmount servers and routers
- up to 24 Ethernet ports in a PC
- no more straight/cross cable problems
- server quality VIA VT6105 chips

$195

RouterBOARD 230

No feature left behind !

Integrated router with various interfaces. Use as an AP on a tower with up to 500ft PoE. Includes IDE/CF, miniPCI, USB, PCMCIA, UART, PCI, GPIO, LCD controller, Linux SDK, and more.

$120 **$65**

RouterBOARD 11/14/18

Multi radio tower !

MiniPCI to PCI adapters for multi radio system. Tested with sixteen radios in one Router/AP.

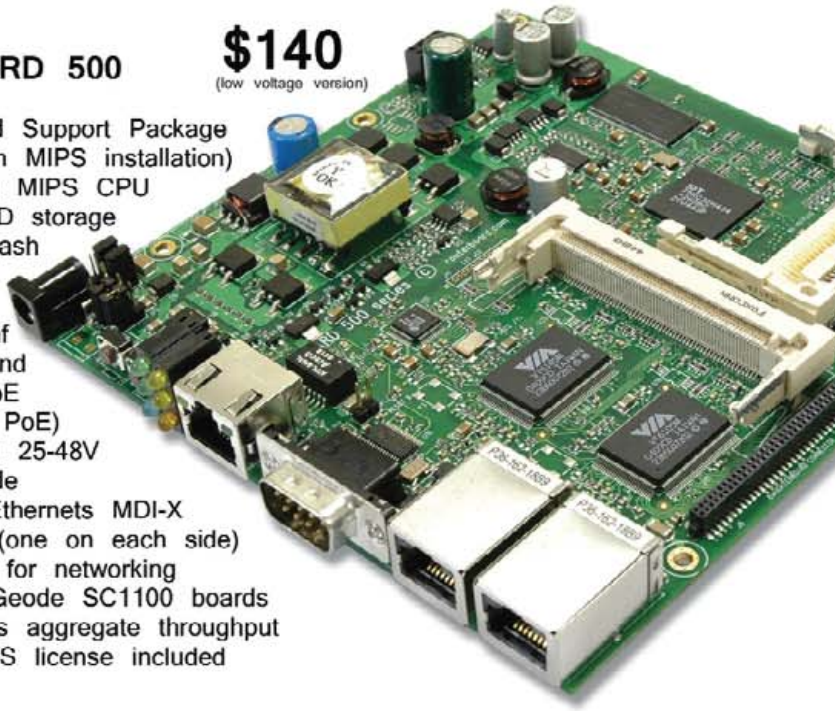
$240

RouterBOARD 500 & RouterBOARD 564

The Wireless Switchboard !

For a complete multi-radio tower system, the RouterBOARD 500 can carry a daughterboard (RouterBOARD 564) which adds six ethernets and four miniPCI.

RouterBOARD 500 **$140** (low voltage version)



- Linux Board Support Package (full Debian MIPS installation)
- 266-400Mhz MIPS CPU
- 64MB NAND storage
- Compact Flash
- 32MB DDR
- Low power
- PoE 802.3af standard and passive PoE (also 12V PoE)
- 10-24V and 25-48V power mode
- 3 10/100 Ethernets MDI-X
- 2 miniPCI (one on each side)
- 2-3x faster for networking than the Geode SC1100 boards
- 200-300MB/s aggregate throughput
- L3 RouterOS license included

Paid for 20GB, Getting 2

I'm trying to set up Linux on a slightly older machine that has a 20GB hard drive. I'm trying to install Fedora on it, but the installation program and Disk Druid recognize it only as a 2GB drive. The same is true of QTParted running off of a boot CD containing Mepis Linux. A Microsoft Windows XP install CD recognizes the size of the CD just fine. Both Linux programs have a problem with the drive whether it is unpartitioned or has an NTFS partition on it. I can't put any Linux partition on it larger than 2GB. Any idea what the problem is?

--

Aaron Roberts, Alr8@georgetown.edu

It's important to back up any data you want to keep from this drive, even from non-Linux partitions, before you try any of the suggestions here. Changing partition tables can be hazardous to your data. You can get some background on this problem from the Large Disk HOWTO by Andries Brouwer at www.tldp.org/HOWTO/Large-Disk-HOWTO.html.

First of all, if the drive is original equipment for the machine, try resetting the BIOS to factory defaults.

--

Don Marti, dmarti@ssc.com

Did you partition the drive with some proprietary disk partitioner? If so, you may want to boot from a bootable Linux CD and wipe the boot sector. If your drive is /dev/hda, as it usually is, use this command:

```
dd if=/dev/zero of=/dev/hda bs=512 count=1
```

Another thing to try is to upgrade the BIOS on that machine.

--

Christopher Wingert, cwingert@qualcomm.com

You can try to force the issue at boot time by supplying a parameter such as:

```
hda=<cyls>,<heads>,<sectors>
```

You usually can obtain these values by looking up the drive's model number on the manufacturer's Web site. Some BIOSes also have a variety of drive reporting mechanisms, such as NORMAL, LBA and LARGE. Try cycling between these and see what you get.

--

Chad Robinson, chad@lucubration.com

Making a Boot Floppy

I need information on how to create a floppy that will allow me to load Linux from a CD-ROM. Can you give me some information about doing this so I can run two OSes (dual boot) on my small system?

--

Everett E. Stone, 103726.2236@compuserve.com

If it's a newer machine, you can boot from CD-ROM. Try the Knoppix distribution.

--

Christopher Wingert, cwingert@qualcomm.com

It's not clear from your question whether your problem is that your CD doesn't boot and you need a bootable floppy to kick start the distribution or if you want to insert or remove the floppy to control whether the system runs from the CD.

If the former, look into Slackware, a popular and well-maintained distribution that still supports bootable floppy options for the load and emergency recovery tools. Otherwise, consider a distribution such as Knoppix, which allows you to run entirely from the CD without loading Linux on your hard drive. This is obviously a much slower option, but it eliminates the need to partition your drive, which is good for taking Linux for a test drive or working from a computer that is not your own.

--

Chad Robinson, chad@lucubration.com

You don't need a boot floppy to set up a dual-boot system. Most of the common distributions will configure dual-boot for you in the installer. See www.linuxjournal.com/article/4619 *to help you select a distribution.*

--

Don Marti, dmarti@ssc.com

Deleting Millions of Files

On my SUSE 9.1 system that is a server for more than 40 Windows machines, my directory /var/lib/dhcp/db contains nearly 1.6 million files with names of the form dhcpd.leases.AAgxyz and so on. If I try to remove them with a single command, such as:

```
rm -f /var/lib/dhcp/db/dhcpd.leases.*
```

the process fails with a `command buffer too long` message. My workaround has been to remove them piecewise, as follows:

```
for i in {a-z,A-Z}
do
  for j in {a-z,A-Z}
  do
    rm -f /var/lib/dhcp/db/dhcpd.leases.$i$j*
  done
done
```

Is there a better way to get rid of such large numbers of files? Any idea why the DCHP dæmon is running away like this?

--

Larry W. Finger, Larry.Finger@lwfinger.net

Determining why the DHCP dæmon is assigning so many leases would require investigating its log files. It's possible that NAK messages are not making it through, MAC addresses are being reassigned or other shenanigans are being perpetrated by the clients.

However, there are certainly easier ways to remove the files en masse. If your dæmon isn't running, you can simply remove and re-create the directory itself, as in:

```
rm -f /var/lib/dhcp/db
mkdir /var/lib/dhcp/db
```

Be sure you restore any permissions and ownership settings the directory originally held. If this isn't to your taste, investigate the find(1) command, which can execute commands on files matching the spec you provide, such as:

```
find /var/lib/dhcp/db -exec rm {} \;
```

Note that the semicolon is required—it tells find where the exec command ends—but that most shells treat it as a special character. The back slash prevents this.

--
Chad Robinson, chad@lucubration.com

You can also use the xargs command, which feeds an arbitrarily long list of arguments to a command:

```
ls  /var/lib/dhcp/db/dhcpd.leases.* | xargs rm
```

Another sometimes-useful technique is to build up a list of commands and preview it with `ls` *to see if you got them right:*

```
ls | awk '{print "rm " $1}' | less
```

Then, replace the `less` *with* `sh` *to do the commands:*

```
ls | awk '{print "rm " $1}' | sh
```

--
Don Marti, dmarti@ssc.com

Server for Home Network?

I am new to Linux, and I would like to know if there is Linux software that would let me set up a file server for my home network that can be monitored and maintained remotely. I have six-plus computers on my local area network running Windows, and I have a spare computer that I would like to set up as a file server for the other computers for storage of MP3s and data. I would like to have this computer set up so that I can connect to it from any of the other computers to do maintenance and updates. I would like some kind of access control for the computers that can access this file server. Any suggestions or links to other sources regarding this project would be a great help.

--
Dan, dpinko@shaw.ca

Try Samba; more than likely, it already is on your machine. SMB is the native file-sharing protocol for Windows, so it is already there too.

--
Christopher Wingert, cwingert@qualcomm.com

Any number of options can allow you to do this. Two good starting points are Samba, a Windows-compatible file and printer sharing service, and Webmin, a Web-based administration tool. Webmin includes modules for configuring and administering Samba and provides its own Web server, so you need not install Apache, unless you want to.

If you must do something Webmin doesn't support, you still can manage the system via traditional command-line tools. Simply enable your distribution's SSH services on the Linux box, and use an SSH terminal from one of the clients to connect to it.

--

Chad Robinson, chad@lucubration.com

A good SSH client to install on Microsoft systems is putty: www.chiark.greenend.org.uk/~sgtatham/putty.

--

Don Marti, dmarti@ssc.com

make or gmake?

I installed Ubuntu on my laptop, which does a fairly skinny install. It has make, but the project I am working on needs gmake. I need to install gmake and compile C++ code on a Linux box; I am compiling nachose-4.02. I did a search in dselect and could not find gmake. I went to the GNU site, and it does not specify gmake, only make. Are they the same?

--

Seamus Rhys, seamusrhys@msn.com

Here is a page specifically about the gmake package for Ubuntu. Download links are provided: higgs.djpig.de/ubuntu/www/hoary/devel/make.

--

Chad Robinson, chad@lucubration.com

Yes, make and gmake are the same program on Linux systems. For historical reasons, some projects specify gmake so that they can be sure to use the full-featured GNU version of make instead of an earlier, limited implementation.

An easy way to add gmake to your system, so you don't have to change the project or install software, is simply to make a symlink in /usr/local/bin:

```
cd /usr/local/bin \
&& sudo ln -s ../../bin/make gmake
```

--

Don Marti, dmarti@ssc.com

Thumbprint Readers?

I was in a large discount store the other day, and I saw a rack of thumbprint readers that tout things like "log on to your computer and Web sites with the touch of a finger, just place your finger on the receiver whenever a password or username is required." A device like this would allow a user or a system administrator to use very large complex passwords for system access.

My office uses Red Hat and another OS; I use Debian. So far, I have not found a product or a HOWTO that recommends a product for use on a Linux machine. Is there a product you can recommend that would work on a Linux system?

--

Tony Freeman, tony.freeman@insightbb.com

This is possible under Linux, but you may need to roll your own solution using a variety of available packages. As far as I know, no distribution comes with support for this enabled out of the box yet, and certainly the software included with the product will be for Windows. To get started, take a look at Pluggable Authentication Modules (PAM). Basically, this is a subsystem that acts as an intermediary between applications, such as your login manager, and authentication sources, such as passwords, certificates and, yes, fingerprint scanners. You may have to configure some system files yourself, but it's possible to make this work using software available today.

--

Chad Robinson, chad@lucubration.com

Getting Sound Working on Fedora

Being a bit of a newbie to Linux—I can install packages but recompiling the kernel might be beyond me—I am looking for some help getting audio support added to my PC. I am using Red Hat FC2 installed on a Compaq Deskpro P550, which has an onboard sound chip ESS Audiodrive 1869. Any help or suggestions would be greatly appreciated.

--

Tom Corcoran, tomc@meridianp2p.com

Your sound card is supported by the ALSA Project, and that's where you should start. It doesn't require a kernel recompile. Although ALSA drivers are included in the kernel, the ALSA Project itself provides external, loadable modules that often are more up to date than those in any kernel package you may have installed. Visit www.alsa-project.org *to get started. The site provides downloadable packages and documentation for installing them.*

--

Chad Robinson, chad@lucubration.com

You shouldn't have to add drivers to a relatively new distribution. Run the sndconfig utility to detect and test your audio hardware, then make sure the volume is turned up using the mixer applet in the GNOME panel.

--

Don Marti, dmarti@ssc.com

Tracking Down malloc Errors

I disagree with the BTS advice "segfault when allocating memory" in the February 2005 issue. I bet ya a nickel that the culprit in this case is the previous malloc followed by writing outside the array boundary. One tool to figure this out is "electric fence", which is open source. There are also good proprietary tools to do this. Maybe you can review a few of them?

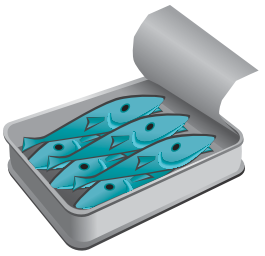
--

Jorg Kewisch, jorg@bnl.gov

Many on-line help resources are available on the *Linux Journal* Web pages. Sunsite mirror sites, FAQs and HOWTOs can all be found at **www.linuxjournal.com**.

Answers published in Best of Technical Support are provided by a team of Linux experts. If you would like to submit a question for consideration for use in this column, please fill out the Web form at **www.linuxjournal.com/lj-issues/techsup.html** or send e-mail with the subject line "BTS" to bts@ssc.com.

Please be sure to include your distribution, kernel version, any details that seem relevant and a full description of the problem.



More sardines. Consider the sardine can: with its perfect sardine symmetry, it's a wonder of space efficiency and performance. But can the same be said of your datacenter? **The Penguin** says it can. Penguin Computing® created BladeRunner™, the extraordinarily dense, Linux-based, blade server. Its innovative design provides for everything you want from your datacenter – networking, processing and storage – in a tidy little package. For you that means huge computational power – in a downright tiny footprint. And now it comes with a downright tiny price tag – that won't tax your budget. So go ahead; put more sardines in the can. Consolidate and prosper. **Love what you do.**

Big return. Smaller investment than ever.

Visit www.penguincomputing.com
Or call 1.888.PENGUIN for more details!

Cyclades AlterPath Manager E200

REVIEWED BY MATTHEW HOSKINS

PRODUCT INFORMATION

Vendor:
Cyclades

URL:
www.cyclades.com/products/25/alterpath_manager

Price:
$8,950 US

THE GOOD

- Simple backup/restore.
- Global console namespace across managed devices.
- Global user management across managed devices.
- Access via SSH and Web.
- Automated firmware updates.

THE BAD

- A little pricey.
- Passwords stored clear text and world-readable.
- Needs more comprehensive documentation.

Cyclades makes a number of excellent products aimed at easing system administration and data center management. These include console (serial and KVM) and remote power management. In the past, these devices were all islands unto themselves needing individual management. Authentication and authorization could be unified easily with a directory service, such as LDAP, Radius, NIS or Kerberos, but the configuration of the devices would need to be managed individually and manually.

In modern complex data center environments, infrastructure must be flexible to keep up with changing circumstances and requirements. A central management system was needed.

Enter AlterPath Manager

Some of Cyclades' most popular products are the TS and ACS serial console management devices. These thin 1U-rackmount enclosures allow secure remote console access to servers and serial-port-equipped appliances, such as filers, routers, firewalls, SAN arrays and switches. The AlterPath Manager (APM) is designed to sit above multiple ACS and TS units and centralize configuration and authentication.

Under the Hood

The APM unit sports an 850MHz Intel Celeron CPU, 256MB RAM, 40GB disk, two 10/100 Ethernet ports and two serial ports, one for the APM's console and another for an optional dial-in modem. Not much horsepower by today's standards, but more than enough for what the APM needs to do. This is all basically off-the-shelf hardware; the APM is primarily a software product that includes integrated hardware.

The hardware is packaged nicely in a sturdy 1U-rack enclosure. Indicators are on the front with all connectors on the rear.

The APM runs a small customized Linux OS. Cyclades' management application is Web-based and runs under the Tomcat Java servlet engine. The servlet engine serves on both HTTP and HTTPS (encrypted) ports, and Cyclades provides simple instructions for disabling the non-encrypted port. All configuration and control of the managed devices is done over encrypted SSH connections.

The APM uses password-style authentication to the managed devices using `expect`. I would have liked to see public key authentication, but passwords are easier to understand for most people and at least it's still all

SERIAL CONSOLES IN THE DATA CENTER

For most computers, the console is the video monitor and a directly attached keyboard. This is where kernel and boot messages go as a system is coming up. The console eventually becomes a login terminal, either graphical or text mode, after a system is fully booted. On servers, however, graphical consoles are not needed and are often unwanted. Consoles on servers usually are used only to recover an ailing system or install a new OS. In these cases, a serial port is used as the console. This provides a very simple device for the kernel to deliver messages without the complexity or wasted CPU cycles of a graphics device. Serial consoles have the added benefit of remote access when used in conjunction with a console server such as the Cyclades ACS series products. These devices literally allow you to use SSH (secure shell) to connect directly to a server's console and manage it from anywhere. Remote access to a server console allows the system administrator to recover and even re-install the OS from anywhere, if the server is running Linux or UNIX. For more information on implementing serial consoles on Linux see my *LJ* article in the August 2004 issue.

encrypted. The root passwords for all managed devices are stored in a MySQL database running on the APM. This database allows connections only from localhost and stores these passwords in clear text. It also appears that the MySQL databases on all APM devices use the same hard-coded database root password. All the database passwords can be found in the world-readable configuration file /var/apm/apm.properties. It needs to be assumed that *any* user with shell access to the APM will have complete control of the managed devices because of the unfettered access to the root passwords. This security situation should be significantly tightened up by Cyclades' developers.

Designing Your Installation

The APM can control any Cyclades TS or ACS console server accessible on your network. All management, as previously stated, is done over encrypted SSH connections. One installation scenario suggested in the APM documentation is to create a private network using the second network port. In this situation, you can allow the APM to serve DHCP and automatically manage the network numbering of the managed devices. This also utilizes the APM as a firewall between your public network and your management network.

Configuring Console Server Devices

Cyclades also provided me with an ACS16 for this review. This device is a small Flash-based Linux box with 16 serial ports that can be used to connect to server consoles, modems, terminals or any other serial devices. Each managed device must have basic networking configuration and a root password set. This is done in exactly the same way as the APM—using an included serial cable and an interactive wizard. If you are planning on using the private network approach mentioned above, simply set the device to use DHCP and set the root password.

Configuring the APM

The initial configuration of the APM is done using a serial cable to a PC or terminal. The APM presents you with a simple configuration utility to get basic networking information, then directs you to continue with a Web browser.

The APM is now ready to configure

and manage devices. Log in to the APM's Web interface and click on Devices, then add. Enter a device name (for example, ACS001), device type, model number, network address and root password. The APM then automatically creates entries for each port on the device named similarly to ACS001_00, ACS001_01 and so on. These names uniquely and globally represent every port on the managed devices. They can be renamed later to something a little more meaningful.

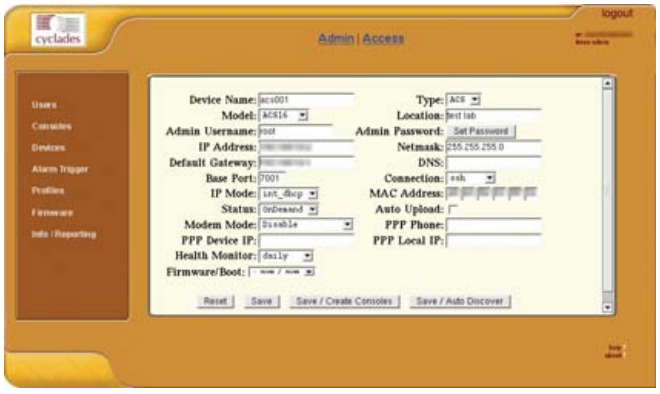


Figure 1. Console Server Device Management

Next, customize one or more Profiles (Figure 1) to describe the various types of devices you intend to connect to the ACS or TS units that this APM will control. The default profile is appropriate for most devices with serial consoles that operate at 9,600 baud, 8 bits, no parity and 1 stop bit.

The next step is to do per-console configuration. Then you're ready to connect to the connected devices. This can be done in one of two ways. From the APM's Web interface you simply can click on the console name under consoles, and the APM launches a Java-based terminal emulator. Alternatively, you can connect directly to a console from any SSH client. If the APM's hostname is myapm, your user name is admin and the console name is myserver, you would issue the following command:

```
ssh admin:myserver@myapm
```

The username:consolename syntax is a Cyclades modification of the SSH server running on the APM. It allows very easy access to the console ports. This is my absolute favorite feature.

User Management

Up to this point, we have been doing everything as the admin user created during the initial configuration. The APM gives you the ability to create users and delegate control of ports. This is useful in a large data center with a complex management structure.

Event Monitoring, Alarms and Logging

The APM has the ability to monitor ports and raise alarms based on what it sees. This is done using pattern expressions. Events are classified as Info, Warning or Severe and are sent by e-mail to users listed in the notify list under each console port configuration.

Every console port has a data buffer and log associated with it. These logs can be viewed with the Web interface.

Firmware Upgrades

Keeping up to date with software and firmware versions is always a task at the forefront of a system administrator's priorities. The APM simplifies this by automating firmware updates of managed devices. Updated firmware packages are downloaded from Cyclades' Web site then installed on the APM. From there they can be pushed to the managed devices.

Backup and Restore

The APM provides a simple command-line tool for backup and restore. This provides an easy-to-use way to back up all configuration, logs and the APM system itself to a remote system using SSH. The restore utility does the exact reverse. So many appliance-style devices do not include this vital feature, but the APM does. It is important not to neglect backup and restore when evaluating any appliance-type device. Any device you depend on for day-to-day administrative operations needs to be classified as critical infrastructure and needs to be held to the same backup, restore and disaster recovery requirements as any other system.

Suggested Improvement

The APM is advertised as a way to unify management of various devices Cyclades produces. These include Power Management (PM Series), KVM (Keyboard Video and Mouse switches over IP) and ACS (Console Management). At this time, there is no integration for PM or KVM devices other than to connect and manage them individually through their console ports. According to Cyclades, future releases of the APM software will include tightly integrated support for PM and KVM ports. Right now, the APM is targeted mainly at managing serial console ports.

One other wish-list feature I would like to see is some ability for the APM to do *all* the initial configuration of a new ACS/TS unit. I would like to be able to unbox a new, factory-fresh ACS, plug it in to the APM's private network or AUX serial port and have the APM do the configuration from the ground up.

Conclusion

The APM does a great job at unifying configuration of Cyclades' various serial console management devices. It also provides a global naming system for console ports, a truly valuable feature. Overall, the APM is a good product, comprising well-designed hardware and software. Some issues should be addressed by the designers as stated above, but these do not affect the overall usability of the device. The security issues I listed above can be worked around by not allowing local shell access to non-administrative users. The APM can manage a maximum of 2,048 console ports (or 42 ACS 48-port units), with a maximum of 256 ports in use at any one time.

Matthew Hoskins is a Senior Linux/UNIX System Administrator for The New Jersey Institute of Technology where he maintains many of the corporate administrative systems. He enjoys trying to get wildly different systems and software working together, usually with a thin layer of Perl (locally known as MattGlue). When not hacking systems, he often can be found hacking in the kitchen. Matt can be reached at matt@njit.edu.

The Official Blender 2.3 Guide: Free 3D Creation Suite for Modeling, Animation, and Rendering

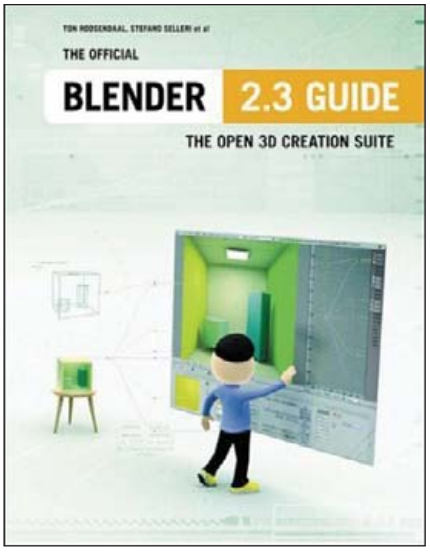
Edited by Ton Roosendaal and Stefano Selleri

No Starch Press, 2004 | ISBN: 1-59327-041-0 | $49.95 US

If you are neither an artist nor an animator but have an interest in trying 3-D design and animation software on your workstation, then Blender 3D is the application for you. As complex as this type of application is, Blender 3D is quite approachable due to the extraordinary support, including documentation, tutorials and demonstrations, produced by the Blender 3D community.

The Official Blender 2.3 Guide is a published compilation of on-line resources produced by the Blender 3D community. As would be expected from an official guide, the contents are authoritative. This book also is comprehensive, targeting the full range of Blender 3D users—this is not a simple command reference.

My only issue with the book's text is the English in some passages is grammatically incorrect to the point that the reader's progress is disrupted, forcing a re-read of a sentence or two. This is to be expected, given that many of the authors do not speak English as a first language and the Blender Documentation Board decided to grant each author stylistic latitude. These occasional lapses are forgiven, though, because the overall quality of the book is so high.



Due to the many supporting images provided throughout the book, readers can learn how to use Blender 3D without a workstation as easily as with one. In fact, I would not hesitate to recommend teaching directly from this text.

Different types of users are going to approach this book from various perspectives. A novice, for example, should start at the beginning and work methodically through the text in order to learn general design concepts and terminology while also learning Blender 3D. An artist already skilled with another 3-D application should start at the beginning as well, if only to cover the Blender 3D interface. However, an experienced Blender 3D artist probably could use the book for its reference sections, to explore little-used techniques and parts of the program.

Because it is a cross-platform application, the authors present important information about running Blender 3D on each platform, including sections about installation and supported graphics cards.

The book comes with a CD that contains Blender 3D 2.32 source code, binaries for several platforms, documentation and representative example work. The book also offers a glossary, but in my opinion, it does not rise to the level of rest of the book. Overall, I highly recommend this book to anybody interested in learning about 3-D design and animation.

—JEFFREY BIANCHINE

NETWORLD+INTEROP

LAS VEGAS • MAY 1–6, 2005

Network Infrastructure and Services
Wireless
Security
Performance
VoIP and Collaboration
Data Management and Compliance

See it All in One Place

ALL SYSTEMS GO

350+ Top Exhibitors on the Exhibit Floor with 8 Targeted Technology Zones and Pavilions

100+ Educational Sessions, Including 6 Comprehensive Conferences Revolving Around 6 Key Themes, 3 Special Interest Days and 36 Tutorials and Workshops

6 Visionary Keynotes by Leading Industry Executives

NetworkWorld Survivor Las Vegas

Visionary Keynotes

John Chambers
President and Chief Executive Officer, Cisco Systems

Hossein Eslambolchi
President—AT&T Global Networking Technology Services, Chief Technology Officer and Chief Information Officer, AT&T

Scott Kriens
Chairman and Chief Executive Officer, Juniper Networks

Sean Maloney
Executive Vice President General Manager, Mobility Group, Intel

Andy Mattes
President and Chief Executive Officer, Siemens Communication Networks

Your Source for Building a Better IT Infrastructure

MediaLive INTERNATIONAL

Register Today at www.interop.com

Use priority code MLAHNV14 and receive $100 off any educational product.

VIA PadLock—Wicked-Fast Encryption

This inexpensive processor offers support for the Advanced Encryption Standard, so you can do state-of-the-art encryption at wire speed.

BY MICHAL LUDVIG

Probably everyone who has used encryption soon realised that the demand for processor power grew instantly. On older systems, the trade-off for using encrypted filesystems is slower file operations; on newer systems, the trade-off is, at minimum, significantly higher CPU loads. Encrypting network traffic with the IPsec protocol also slows things down, and sometimes you may encounter performance problems even on the standard 100Mbps network.

Options exist, however, for working around these encryption/performance trade-offs:

- Don't encrypt: apparently the cheapest solution, but this can become very expensive in the long run.
- Accept the slowdown: the typical approach.
- Use a standalone cryptography accelerator: a PCI card, for example, doesn't help as much as you might expect, however, because the data must traverse the PCI bus more often than necessary.
- Use a CPU with VIA PadLock technology. What's VIA PadLock? Read on.

VIA PadLock

A while back, VIA introduced a simple but slightly controversial approach: select some cryptographic algorithms and wire them directly in to the CPU. The result was the introduction of an i686 class processor that understands some new instructions dedicated to cryptographic functions. This technology is called VIA PadLock, and the processor is fully compatible with AMD Athlons and Intel Pentiums.

The PadLock features available on your machine's processor are determined by its version. Processor versions usually are written as a family-model-stepping (F/M/S) triplet. Family is always 6 for i686 class CPUs. If the model is 9, your CPU has a Nehemiah core; if the model is 10, it has an Esther core. The stepping denotes a revision of each model. You can find your processor's version in /proc/cpuinfo.

Nehemiah stepping 3 and higher offers an electrical noise-based random number generator (RNG) that produces good random numbers for different purposes. The instruction for accessing the RNG is called xstore. As in Intel and AMD processors, the random number generator in VIA processors is supported by the hw_random device driver.

Nehemiah stepping 8 and higher contains two independent RNGs and the Advanced Cryptography Engine (ACE). The ACE can encrypt and decrypt data using the Advanced Encryption Standard (AES) algorithm with three standard key lengths—128, 192 and 256 bytes—in four different modes of operation: electronic codebook (ECB), cipher block chaining (CBC), cipher feedback (CFB) and output feedback (OFB) modes (see the on-line Resources). The appropriate instructions are called xcryptecb, xcryptcbc and so on. Later in this article, I predominantly use their common group name, xcrypt, instead of the mode-specific instruction names.

Esther stepping 0 and higher inherited two RNG units from Nehemiah. ACE was extended with counter (CTR) mode support and MAC (Message Authentication Code) computation. And there are two new acronyms, PHE and PMM. PadLock Hash Engine (PHE) is used for computing a cryptographic hash, also known as a digest, of a given input block, using the SHA1 or SHA256 algorithm. The proposed instruction name is xsha.

The PadLock Montgomery Multiplier (PMM) is responsible for speeding up one of the most time-consuming computations used in asymmetric, or public-key, cryptography: A^B mod M, where A, B and M are huge numbers, usually 1,024 or 2,048 bits. This instruction is called montmul.

As noted above, in the rest of this article I mostly speak about the xcrypt instruction. Principles described further mostly are valid for other units as well, and xcrypt serves only as an example. Also, the terms and concepts covered in this encryption discussion apply to decryption as well.

How to Use PadLock

In contrast to the external cryptography accelerators usually plugged in to PCI slots, the PadLock engine is an integral part of the CPU. This fact significantly simplifies its use, because it is not necessary to bother with accessing the bus or with interrupts, asynchronous operations and so on. Encrypting a block of memory with xcrypt is as easy as copying it over with the movs instruction.

At this point, encryption is almost an atomic operation. Before executing the instruction, the buffer contains plain-text input data; a few clock cycles later, when the execution finishes, we have ciphertext. If a task requested processing of a single block, which is 16 bytes in the case of the AES algorithm, the operation is fully atomic. That is, the CPU doesn't interrupt it in the middle and doesn't do anything else until the encryption is finished.

But what if the buffer contains a gigabyte of plain text to

be processed? It isn't good to stop all other operations and wait for the encryption to finish when it's this large. In such a case, the CPU can interrupt the encryption after every single block of 16 bytes. The current state is saved, and whatever else can be done is done—interrupts can be handled and processes switched. As soon as the encrypting process is restarted, the instruction continues from the point at which it was suspended. That's why I say this is almost an atomic operation: for the calling process it looks atomic, but it can be interrupted by a higher-priority event. The current processing state then is saved into the memory and registers of the running process, which enables multiple tasks to do encryption simultaneously, without the risk of mixing their data. Again, it is an analogous situation to copying memory blocks with the movs instruction.

How Fast Is It?

According to VIA, the maximum throughput on 1.2GHz processors exceeds 15Gb/s, which is almost 1.9GB/s. The benchmarks I have run confirm that such a speed could be achieved in real-world applications and not only in VIA marketing papers, which definitely was a nice surprise.

The actual encryption speed depends on two factors, cipher mode and data alignment. ECB is the fastest, while the most widely used CBC mode runs at about half of the ECB speed. PadLock requires the data to be aligned at 16-byte boundaries, so unaligned data must be moved to proper addresses first, which takes some time. In some cases, the Esther CPU can realign the data automatically, but this still causes some slowdown.

Table 1 shows some numbers from my testing. The OpenSSL benchmark for VIA Nehemiah 1.2GHz produced the following results in kB/s.

The bigger the blocks are, the better the results are, because the overhead of the OpenSSL library itself is eliminated. Encryption of 8kB blocks in ECB mode can run at about 1.7GB/s; in CBC mode, we get about 800MB/s. In comparison to software encryption, PadLock in ECB mode is 120 times faster on the same processor, and CBC mode is 60 times faster.

Thanks to this speedup, the IPsec on 100Mbps network runs at almost full speed somewhere around 11MB/s. Similar speedups can be seen on encrypted filesystems. The Bonnie benchmark running on a Seagate Barracuda in UDMA100 mode produced plain-text throughput at a rate of 61,543kB/s; with PadLock, it was 49,961kB/s, and a pure software encryption ran at only 10,005kB/s. In other words, PadLock was only about 20% slower, while the pure software was almost 85% slower than the non-encrypted run. See Resources for a link to my benchmark page with more details and more numbers.

Linux Support

So far I have developed Linux support for the following packages only for the AES algorithm provided by the xcrypt instruction, because I haven't used the Esther CPU yet. As soon as I get the new processor, I will add support for the other algorithms where appropriate.

Table 1. The Open SSL Benchmark for VIA Nehemiah 1.2GHz, in kB/s

Type	16 bytes	64 bytes	256 bytes	1,024 bytes	8,192 bytes
aes-128-ecb (software)	11,274.53	14,327.79	14,608.64	14,672.55	14,693.72
aes-128-ecb (PadLock)	66,892.82	346,583.52	910,704.21	1,489,932.59	1,832,151.72
aes-128-cbc (software)	8,276.27	12,915.75	13,264.13	13,313.02	13,322.92
aes-128-cbc (PadLock)	48,542.30	241,898.79	523,706.28	745,157.61	846,402.90

Kernel

When the kernel needs the AES algorithm, it loads by default the aes.ko module, which provides its software implementation. To use PadLock for AES, you must load the padlock.ko module instead. You can do this either by hand with modprobe or by adding a single line to /etc/modprobe.conf:

```
alias aes padlock
```

Now, every time the kernel requires AES, it automatically loads padlock.ko too.

Patches are available for kernel version 2.6.5 and above; see the PadLock in Linux home page in Resources. Also, the basic driver will be available in the vanilla 2.6.11 kernel without any patching.

OpenSSL

Those amongst us who are brave enough to use recent CVS versions of OpenSSL already have PadLock support. Users of OpenSSL 0.9.7 have to patch and rebuild the library, or they can use a Linux distribution that has the patch already included in its packages, such as SUSE Linux 9.2.

To see if your OpenSSL build has PadLock support, run this simple command:

```
$ openssl engine padlock
(padlock) VIA PadLock (RNG, ACE)
```

If instead of (RNG, ACE) you see (no-RNG, no-ACE), it means that your OpenSSL installation is PadLock-ready, but your processor is not. You also could see an ugly error message saying that there is no such engine. In that case, you should upgrade or patch your OpenSSL library.

For programs that use OpenSSL for their cryptography needs to enjoy PadLock support, they must use the so-called EVP_interface and initialize hardware accelerator support somewhere at the beginning of their runs:

```
#include <openssl/engine.h>
int main ()
{
    [...]
    ENGINE_load_builtin_engines();
    ENGINE_register_all_complete();
    [...]
}
```

See the evp(3) man page from the OpenSSL documentation for details.

In SUSE Linux 9.2, for example, OpenSSH has a similar patch to let you experience much faster scp transfers over the network.

Binutils

To use PadLock in your own programs, you either can call the instruction by name—for example, xcryptcbc—or write its hexadecimal form directly:

```
.byte 0xf3,0x0f,0xa7,0xd0
```

For backward compatibility with older development tools, it is safer to use the opcode form. Binutils versions 2.15 and newer, however, already understand the symbolic names where appropriate, for example, in gas (GNU assembler) or objdump programs. The binutils' BFD-library responsible among other things for instruction-level operations also is used in the GNU debugger gdb. A sample instruction dump of an encryption function may be as simple as:

```
(gdb) x/3i $pc
0x8048392 <demo1+14>:    lea    0x80495f0,%edx
0x8048398 <demo1+20>:    repz xcryptecb
0x804839c <demo1+24>:    push   %eax
```

As you might have guessed, SUSE Linux 9.2 has PadLock patches in all the appropriate packages, and you can enjoy PadLock support out of the box. If your distribution does not have these patches, check out my Linux PadLock home page in Resources for the available patches.

Programming PadLock

In the following sections, I describe some guidelines for programming PadLock, including details of xcryptcbc. I also explain how to set up PadLock for encrypting a buffer of data with the AES algorithm and a key length of 128 bits in CBC mode. All other instructions of the xcrypt group are used in exactly the same way. Other PadLock functions apply similar rules.

xcryptcbc

xcryptcbc does not have any explicit operands. Instead, every register has a given, fixed function:

- ESI—source address.
- EDI—destination address.
- EAX—initialization vector address.
- EBX—cipher key address.
- ECX—number of blocks for processing.
- EDX—control word address.

Unless written otherwise, all addresses must be aligned at 16-byte boundaries.

ESI/EDI—Addresses of the Source/Destination Data

Both source and destination addresses can be the same, so it is possible to encrypt in place. The size of the destination buffer must be at least the size of the source one. Both must be a multiple of the block size, 16 bytes. Under some circumstances, the Esther CPU allows processing of unaligned buffers, but the operation is slower.

EAX—Initialization Vector Address

The initialization vector (IV) is one of the parameters on which the result of the encryption depends. The size of the IV is the same as the block size, which is 16 bytes. Consult the literature for details about initialization vectors.

EBX—Cipher Key Address

Cipher keys can have one of the following sizes: 128, 192 or 256 bits. The AES algorithm internally uses a so-called expanded key, which is derived from the given cipher key. For 128-bit keys, the expanded key can be computed by PadLock. For longer keys, you must compute it yourself.

ECX—Number of Blocks to Process

The xcrypt instruction always is used with the rep prefix, which enables its repetitive execution unless the ECX register is zero. The value in ECX is decremented after each block is encrypted or decrypted.

EDX—Control Word Address

To let PadLock know exactly how to process the data, we must fill a structure called control word with the following items:

- Algorithm—you can choose only AES.
- Key size—one of the supported sizes.
- Enc/Dec—direction: encryption or decryption.
- Keygen—did we prepare the expanded key or should PadLock compute it itself?
- Rounds—internal value of the algorithm; see the explanation later in the text and in PadLock documentation.

In C, we can use union to allocate the appropriate space for the structure and a bit field to describe and access its items easily:

```
union cword {
    uint8_t cword[16];
    struct {
        int rounds:4;
        int algo:3;
        int keygen:1;
        int interm:1;
        int encdec:1;
        int ksize:2;
    } b;
};
```

Assembler Example

Now that we know all the theory, it's time for a real example.

Encrypting a block of memory with xcrypt is as easy as copying it over with the movs instruction.

To begin, here are some lines of pure assembler:

```
.comm   iv,16,16
.comm   key,16,16
.comm   data,16,16
.comm   cword,16,16

.text
cryptcbc:
    movl    $data, %esi  #; Source address
    movl    %esi, %edi   #; Destination
    movl    $iv, %eax    #; IV
    movl    $key, %ebx   #; Cipher key
    movl    $cword, %edx #; Control word
    movl    $1, %ecx     #; Block count
    rep xcryptcbc
    ret
```

This piece of code encrypts one block of data with a cipher key and an initialization vector, following the parameters set in control word cword. Notice that we use the same address for both source and destination data, therefore we encrypt in-place. Because the field data has a size of only a single block, we set the ECX register to one.

C Language Example

To use PadLock directly in a C program, we can write the PadLock routines to separate assembler source files, then compile to standalone modules and finally link to our binary. It often is easier, though, to use the GCC inline assembler and write the instructions directly in the C code. See Resources for a link to a tutorial on the inline assembler.

```
static inline void *
padlock_xcryptcbc(char *input, char *output,
    void *key, void *iv, void *control_word,
    int count)
{
    asm volatile ("xcryptcbc"
        : "+S"(input), "+D"(output), "+a"(iv)
        : "c"(count), "d"(control_word), "b"(key));
    return iv;
}
```

This code instructs the compiler to load the given values of input, count and other parameters into the appropriate registers. It then is told to issue the xcryptcbc instruction and, finally, to return the value found in the EAX register as a pointer to the new initialization vector.

To be successful here, we also must fill in the control word structure correctly. First of all, it is a good idea to clear the union to avoid using any irrelevant values that might be in the memory:

```
memset(&cword, 0, sizeof(cword));
```

Now let's fill in the fields one by one. The first one in the list is rounds. This item specifies how many times AES processing should be run with the input block, each round using a unique part of the expanded key. To comply with the FIPS AES standard, use 10 rounds for 128-bit keys, 12 rounds for 192 bits and 14 rounds for 256 bits. Should the key_size variable contain the length of the cipher key in bytes, this is how we get the rounds value:

```
cword.b.rounds = 10 + (key_size - 16) / 4;
```

The next field is algo. This is reserved to let you choose future encryption algorithms instead of AES, although AES is the only option at the moment. Therefore, leave zero here.

The keygen field must be set to one if we prepare the expanded key ourselves. Zero means that PadLock should generate it instead, but that is possible only for 128-bit keys:

```
cword.b.keygen = (key_size > 16);
```

The item interm enables the storing of intermediate results after each round of the algorithm is run. I suspect the CPU architects used this field for debugging their core, and I don't see much sense in setting this in the program.

Encryption is distinguished from decryption by the bit encdec. Zero is encryption; one is decryption.

Finally, we must set the key size in the two bits of ksize:

```
cword.b.ksize = (key_size - 16) / 8;
```

That's it. With this prepared control word structure and properly aligned buffers, we can call padlock_xcryptcbc(). If the electrons are on our side, in a short while we receive the encrypted data.

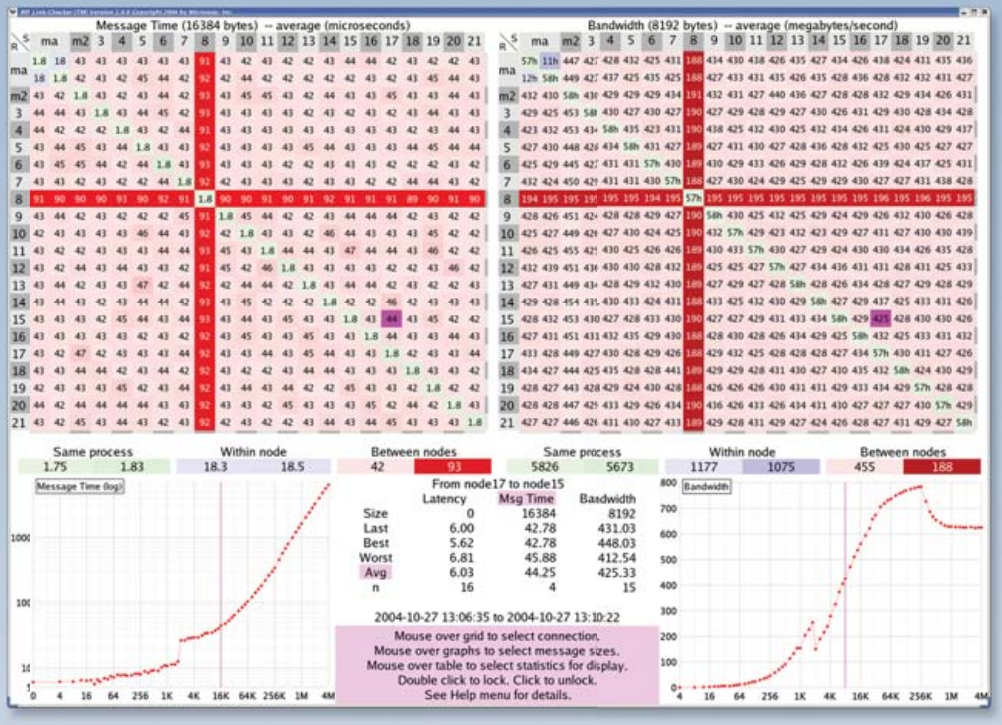
Conclusion

PadLock documentation is available publicly on the VIA Web site; there you can find further information about PadLock programming caveats. The complete example program for encrypting one block of data, including verification of the result, can be found on my PadLock in Linux home page. See Resources for additional links.

Resources for this article: www.linuxjournal.com/article/8137.

Michal Ludvig (michal@logix.cz) recently moved from Prague in the Czech Republic to Auckland on the other side of the world to work as a senior engineer for Asterisk Ltd. He enjoys exploring the secrets of New Zealand with his wife and daughter.

X Marks the Slow Node!



Microway® Quad Opteron™ Cluster with 36 Opteron 852s, redundant power and 45 hard drives in CoolRak™ cabinet.

MPI Link-Checker™ to the Rescue!

A single slow node or intermittent link can cut the speed of MPI applications by half. Whether you use GigE, Myrinet, Quadrix, InfiniBand or InfiniPath HTX, there is only one choice for monitoring and debugging your cluster of SMP nodes:

Microway's MPI Link-Checker™

Our unique diagnostic tool uses an end-to-end stress test to find problems with cables, processors, BIOS's, PCI buses, NIC's, switches, and even MPI itself! The newest release provides ancillary data on inter-process and intra-CPU latency which can vary by a factor of 10 between MPI versions. MPI Link-Checker is also useful for porting applications to new hardware. It provides instant details on how latency and bandwidth vary with packet size. It is available now for a free 30 day evaluation!

Wondering what's wrong with your cluster, or need help designing your next one? Call our HPC staff at 508-746-7341. Visit microway.com to learn about new low latency interconnects including the PathScale InfiniPath HTX Adapter, which delivers unmatched MPI latency of under 1.5 microseconds.

Microway has been an innovator in HPC since 1982. We have thousands of happy customers. Isn't it time you became one?

Call us first at 508-746-7341 for quotes and benchmarking services. Find technical information, testimonials, and newsletter at www.microway.com.

Excellence in Cluster Technology
64-bit Turnkey Solution
WINNER
ClusterWorld

Microway®
23 Years of Expertise Built In

PathScale™
Accelerating Cluster Performance™

Writing a GCC Front End

Language designers rejoice! Now it's easier to put a front end for your language onto GCC.

BY TOM TROMEY

GCC, the premier free software compiler suite, has undergone many changes in the last few years. One change in particular, the merging of the tree-ssa branch, has made it much simpler to write a new GCC front end.

GCC always has had two different internal representations, trees and RTL. RTL, the register transfer language, is the low-level representation that GCC uses when generating machine code. Traditionally, all optimizations were done in RTL. Trees are a higher-level representation; traditionally, they were less documented and less well known than RTL.

The tree-ssa Project, a long-term reworking of GCC internals spearheaded by Diego Novillo, changes all that. Now, trees are much better although still imperfectly documented, and many optimizations are done at the tree level. A side effect of this work on trees was the clear specification of a tree-based language called GENERIC. All GCC front ends generate GENERIC, which is later lowered to another tree-based representation called GIMPLE, and from there it goes to RTL.

What this means to you is that it is much, much simpler to write a new front end for GCC. In fact, it now is feasible to write a front end for GCC without any knowledge of RTL whatsoever. This article provides a tour of how you would go about connecting your own compiler front end to GCC. The information in this article is specific to GCC 4.0, due to be released in 2005.

Representing the Program

For our purposes, compilation is done in two phases, parsing and semantic analysis and then code generation. GCC handles the second phase for you, so the question is, what is the best way to implement phase one?

Traditional GCC front ends, such as the C and C++ front ends, generate trees during parsing. Front ends like these typically add their own tree codes for language-specific constructs. Then, after semantic analysis has completed, these trees are lowered to GENERIC by replacing high-level, language-specific trees with lower-level equivalents. One advantage of this approach is the language-specific trees usually are nearly GENERIC already. The lowering phase often can prevent too much garbage from generating.

The primary disadvantage of this approach is trees are typed dynamically. In theory, this might not seem so bad—many dynamically typed environments exist that can be used efficiently by developers, including Lisp and Python. However, these are complete environments, and GCC's heavily macroized C code doesn't confer the same advantages.

My preferred approach to writing a front end is to have a strongly typed, language-specific representation of the program, called an abstract syntax tree (AST). This is the approach used by the Ada front end and by gcjx, a rewrite of the front end for the Java programming language.

For instance, gcjx is written in C++ and has a class hierarchy that models the elements of the Java programming language. This code actually is independent of GCC and can be used for other purposes. In gcjx's case, the model can be lowered to GENERIC, but it also can be used to generate bytecode or JNI header files. In addition, it could be used for code introspection of various kinds; in practice, the front end is a reusable library.

This approach provides all the usual advantages of a strongly typed design, and in the GCC context, it results in a program that is easier to understand and debug. The relative independence of the resulting front end from the rest of GCC also is an advantage, because GCC changes rapidly and this loose coupling minimizes your exposure.

Potential disadvantages of this approach are the possibilities that your compiler might do more work than is strictly needed or use more memory. In practice, this doesn't seem to be too important.

Before we talk about some details of interfacing your front end to GCC, let's take a look at some of the documentation and source files you need to know. Because it hasn't been a priority in the GCC community to make it simpler to write front ends, some things you need to know are documented only in the source. The documentation references here refer to info pages and not URLs, because GCC 4.0 has not yet been released. Thus, the Web pages reflect earlier versions. Your best bet is to check out a copy of GCC from CVS and dig around in the source.

- gcc/c.opt: describes command-line options used by the C family of front ends. More importantly, it describes the format of the .opt files. You'll be writing one of these.
- gcc info page, node Spec Files (source file gcc/doc/invoke.texi): describes the spec mini-language used by the GCC driver. You'll write some specs to tell GCC how to invoke your front end.
- gccint info page, node Front End (source file gcc/doc/sourcebuild.texi): describes how to integrate your front end into the GCC build process.
- gccint info page, node Tree SSA (source file gcc/doc/tree-ssa.texi): describes GENERIC.
- gcc/tree.def, gcc/tree.h: some attributes of trees don't seem to be documented, and reading these files can help.

tree.def defines all the tree codes and is, in large part, explanatory comments. tree.h defines the tree node structures, the many accessor macros and declares functions that are useful in building trees of various types.

- libcpp/include/line-map.h: line maps are used to represent source code locations in GCC. You may or may not use these in your front end—gcjx does not. Even if you do not use them, you need to build them when lowering to GENERIC, as information in line maps is used when generating debug information.

- gcc/errors.h and gcc/diagnostic.h: define the interface to GCC's error formatting functions, which you may choose to use.

- gcc/gdbinit.in: defines some GDB commands that are handy when debugging GCC. For instance, the `pt` command prints a textual representation of a tree. The file .gdbinit also is made in the GCC build directory; if you debug there, the macros immediately are available.

- gcc/langhooks.h: lang hooks are a mechanism GCC uses to allow front ends to control some aspects of GCC's behavior. Each front end must define its own copy of the lang hooks structures; these structures consist largely of function pointers. GCC's middle and back ends call these functions to make language-specific decisions during compilation. The lang hooks structures do change from time to time, but due to the way GCC expects front ends to initialize these structures, you largely are insulated from these changes at the source level. Some of these lang hooks are not optional, so your front end is going to implement them. Others are ad hoc additions for particular problems. For instance, the can_use_bit_fields_p hook was introduced solely to work around an optimization problem with the current gcj front end.

Writing the Driver

Currently, GCC requires your front end to be visible at build time—there is no way to write a front end that is built separately and linked against an installed GCC. For this step, read through the appropriate section of the GCC manual to find out how to write the build infrastructure needed for your front end. Ordinarily, the simplest way is to copy another front end's files and modify them to suit.

Next, write two files to help integrate your front end into the GCC driver program. The lang-specs.h file describes your front end to the GCC driver. It tells the driver the file extensions that, when seen on the command line, should cause GCC to invoke your front end. It also gives the driver some instructions for what other programs must be run, such as whether the assembler should be run after your front end and how to pass or modify certain command-line options. It may take a while to write this file, as specs are their own strange language. However, examples in the other front ends can help.

The lang.opt file describes any com-

mand-line options specific to your front end. This is a plain-text file written in a straightforward format. Simple options, such as warning flags, can be put in lang.opt and do not require any other code on your part. Other arguments have to be handled by a lang hook you must write.

Next, implement the lang hooks needed to drive the compilation process. The important ones in this category are:

- init_options: the first call made to your front end, before any option processing is done.
- handle_option: called to handle a single command-line option.
- post_options: called after all command-line processing has been done. This lang hook also is a convenient place to determine the name of the input file to parse.
- init: called after post_options to initialize your front end.
- finish: called after all compilation is done. You can use this to clean up after your front end, if necessary.
- parse_file: a lang hook that does all the parsing, semantic analysis and code generation needed for the input file. It does all the actual work of compilation.

Initialization

GCC needs your front end to do some initialization. Most of GCC is self-initializing, but in order to accommodate the needs of different front ends, it is possible to initialize some tree-related global variables in atypical ways. I recommend not trying to delve too deeply into this. It is simpler to define the standard tree nodes in the standard ways and to think up your own names for trees representing, say, the standard types in your language.

During initialization call build_common_tree_nodes, set_sizetype and build_common_tree_nodes_2. set_sizetype is used to set the type of the internal equivalent of size_t; it is simplest to set this always to long_unsigned_type_node.

Other setup steps can be done in this phase. For instance, in the initialization code for gcjx, we build types representing various structures that we need to describe Java classes and methods.

Compiling to GENERIC

Your parse_file lang hook calls your compiler to generate your internal data structures. Assuming this completes without errors, your front end now is ready to generate GENERIC trees from your AST. In gcjx, this is done by walking the AST for a class using a special visitor API. The GENERIC-specific implementation of this API incrementally builds trees representing the code and then hands this off to GCC.

All the details of generating trees are outside the scope of this article. Below are examples, however, showing three major tree types so you can see what each looks like.

Type

One kind of tree represents a type. Here is an example from gcjx of the Java char type:

```
tree type_jchar = make_node (CHAR_TYPE);
TYPE_PRECISION (type_jchar) = 16;
fixup_unsigned_type (type_jchar);
```

You can represent any type using trees. In particular, there are tree types representing records, unions, pointers and integers of various sizes.

Decl

Decl represents a declaration or, in other words, a name given to some object. For instance, a local variable in the source code is represented by a decl:

```
tree local = build_decl (VAR_DECL, get_identifier ("variable_name"),
                         type_jchar);
```

There are decls representing various named objects in a program: translation units, functions, fields, variables, parameters, constants, labels and types. A type decl represents the declaration of the type, as opposed to the type itself.

Expr

Many kinds of expr trees are available that represent the various kinds of expressions in a program. These are similar to C expressions but are more general in some ways. For instance, trees do not distinguish between if statements and conditional expressions—both are represented by a COND_EXPR, with the only difference being that an if statement has void type. Here's how we can build an expression that adds our variable to itself:

```
tree addition = build2 (PLUS_EXPR, type_jchar, local, local);
```

Trees that represent statements are linked together using a special iterator API. Here is how we would chain together two statements, s1 and s2:

```
tree result = alloc_stmt_list ();
tree_stmt_iterator out = tsi_start (result);

tsi_link_after (&out, s1, TSI_CONTINUE_LINKING);
tsi_link_after (&out, s2, TSI_CONTINUE_LINKING);

// Now 'result' holds the list of statements.
```

Other kinds of tree nodes exist; read tree.def and the manual for a more complete understanding. It also is possible for a front end to define its own tree codes; however, if you have your own AST, you should not need to do this.

The overall structure of the program you generate probably is going to resemble a translation unit decl, which would contain types, variables and functions.

Handoff

Once you've built the trees representing a function, a global variable or a type for which you want to generate debugging information, you need to pass the tree to the appropriate function to handle the rest of compilation. Three such functions are

available at present: rest_of_decl_compilation handles compilation for a decl node, cgraph_finalize_function handles compilation for a function and rest_of_type_compilation handles compilation for a type.

Debugging

Although GCC has a fair number of internal consistency checks, it still is too easy to provoke crashes in code that are unrelated to your front end. In many cases, you can move up the stack, printing whatever trees are being manipulated, until you find some discrepancy caused by incorrect tree generation. This technique requires surprisingly little general GCC knowledge in order to effectively debug your code.

GCC has some handy debug functionality. In the debugger you can call the debug_tree function to print a tree. You also can use the -fdump-tree family of command-line options to print trees after various passes have been run.

Experience

My experience writing gcjx has been that lowering its strongly typed intermediate representation to trees is quite simple. The tree back end to gcjx, one back end among several, represents roughly 10% of the total code of the compiler. Although unfinished, it currently weighs in at 6,000 lines of code (raw `wc -l` count)—around the same size as the bytecode back end. One inference to draw from this is if you already have a compiler, the task of attaching it to GCC can be accomplished easily.

As trees are high-level, I haven't looked at any RTL while writing this front end. I haven't spent any time at all thinking about or dealing with processor-specific issues. Unless your language has some esoteric requirements, this ought to hold for you as well.

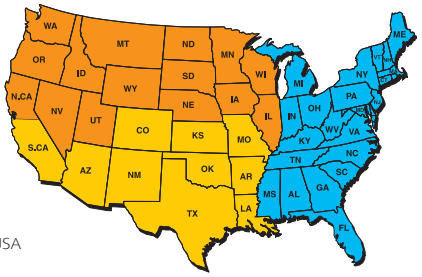
The statically typed AST in gcjx is easily reused. Currently, there are four back ends, and I expect to write more later. For instance, it would be simple to build a back end that writes a cross-reference representation of the program to a database. Or, it would be possible to write a back end that walks the AST searching for typical errors, akin to the FindBugs program. This consideration would be even more compelling for languages, which, unlike Java, don't already have a wealth of analysis tools.

Future Directions

The process of writing a front end certainly could be made even easier. For instance, there is no need to require lang-specs.h. Instead, a front end could install a description file that the GCC driver would read at startup. Similarly, lang.opt probably could be dispensed with. With more work, it even would be possible to compile front ends separately from the rest of GCC.

Resources for this article: www.linuxjournal.com/article/8138.

Tom Tromey has been involved with free software since 1991 and has worked on many different programs. He currently is employed as an engineer at Red Hat, working on GCJ.



LINUX JOURNAL

PO Box 55549
Seattle, WA 98155-0549 USA
www.linuxjournal.com

Linux in the Classroom: an Experience with Linux and Open-Source Software in an Educational Environment

Formerly tied to one proprietary OS, this multicampus college is broadening students' horizons with Linux, Samba, SquirrelMail and more.

BY JOSEPH RUFFOLO AND RON TERRY

Mountainland Applied Technology College (MATC) is one of several campuses that are a part of the Utah College of Applied Technology (UCAT) system. Our department teaches information technology courses to both high-school and adult students. We provide individual technology courses, short-term intensive training (STIT) courses, custom fit courses and an education track that ends with an Associate of Applied Technology degree or Certificate of Completion in Information Technology. We currently have two satellite sites with a probability of adding two more in the 2005–2006 school year.

Until the last few years, we had relied solely on Microsoft operating systems in our classrooms. Unfortunately, when you give students sufficient rights to work on their course material and labs, spyware and viruses are introduced despite your best uses of firewalls and virus scanners. These problems have entered the network through means such as Web surfing, floppies and USB thumbdrives.

Our Evolving Use of Linux

Starting with the 2001–2002 school year, Mountainland Applied Technology College began offering Linux courses at the Orem MATC campus. Over the next several years, our Linux class enrollments and course offerings continued to grow, matching the growing use of Linux at home and in businesses. Coupled with individual Linux distribution evaluations, teaching these courses was a convincing experience that Linux was a viable operating system for the IT department's infrastructure and for hosting courseware on our classroom machines.

At the start of the 2000–2001 school year, individual classroom servers were running either Microsoft Windows 2000 or Novell Netware. We started using a Linux-based firewall/router product called Freesco in the individual classrooms to help deal with the increasing amount of malware and the security inadequacies of Windows.

During the 2001–2002 school year, a Windows 2000 system continued as the main classroom server. Up until this time we still had no main department server. We brought up a system with Linux as a secondary server and started working on integrating Linux into Microsoft Active Directory with great difficulty and little success. We tried using Microsoft Services for Unix, winbind (Samba 2.2.x) and pam_smb, but we still couldn't make things work well and temporarily abandoned our attempt to integrate Linux authentication with Active Directory. During this school year, we also taught our first Linux course for CompTIA Linux+ certification.

During the 2002–2003 school year, we replaced the classroom Windows 2000 server with a Linux system running Mandrake 9.1. We had an epiphany, so to speak, when we changed our paradigm in relation to what to use on our server(s) in the back end. We discovered that using Linux for the back end made it a much easier task to integrate different operating systems, such as Microsoft and Linux, as opposed to using Windows and Active Directory as the back end.

During the second half of the school year, various Linux distributions, such as Red Hat 9 beta and Mandrake 9.1, were beta tested on the desktop. NIS was used for user authentication on Linux clients and Samba, as a domain controller, for user authentication in Windows. Although this setup worked, it was frustrating to deal with multiple passwords for the same user account, and we started looking for a better solution. During this school year, we also taught the first incarnation of our Linux Server Administration course, using what we had learned through our experiences up to this point.

The 2003–2004 school year was a time when Linux technology, other open-source software and our understanding and ability to use it all caught up with their promised potential. The main IT classroom servers were moved from NetWare and Windows 2000 to a single department server running Linux, in this case Mandrake 9.1. We started using OpenLDAP as a central repository for user authentication. LDAP was used as the back-end database for Samba, and we used the pam_ldap modules for Linux client authentication. We started using Linux on the desktop—initially Red Hat 9, Fedora Core 1 or SUSE 9—as the default option in a dual-boot con-

figuration with Windows XP Pro. At this time, the majority of our students chose the Windows option with a few brave souls trying Linux.

We also retained Windows because several courseware packages either required Microsoft products, including the Windows OS, Internet Explorer or Media Player, or the browsers available for Linux were not sufficiently Microsoft-Internet-Explorer-compatible to use some sites. All things considered, we were pleased with the stability of Linux on the desktop. We were impressed particularly with the virtual elimination of problems with viruses, spyware and overly curious students that we suffered with desktop Windows systems.

The 2004–2005 school year has proved to be one of continued and significant improvements. We upgraded the main Information Technology department server to Fedora Core 2 to gain the advantages of the Linux 2.6 kernel. We also started offering students in the Information Technology department e-mail using Postfix, Dovecot and SquirrelMail, with filtering provided by SpamAssassin and ClamAV. On the Microsoft compatibility front, we upgraded to Samba 3, which provides much better integration with OpenLDAP and creates a new opportunity for us in our quest for a true single sign-on solution.

We now have an environment where our users can log in to either Linux or Windows XP Pro using the same user name and password. Linux clients authenticate using pam_ldap, and users have home directories stored on the server, shared via NFS and dynamically mounted at login time using autofs. Windows clients are joined to a domain controlled by Samba, allowing users to authenticate using the same account information, user name and password as they would if they were logging in to a Linux client. The same home directories on the server that are used with Linux are available in Windows through automatic drive mappings. Windows users' roaming user profiles also are stored in their home directories on the server.

To take a step further in the single sign-on arena, users also use the same account information to access their e-mail. We have provided Web-based access to e-mail, which also is stored in their home directories on the server, through SquirrelMail. Standard POP3 or IMAP access is provided by Dovecot.

Fedora Core 2 or Fedora Core 3 is the primary desktop operating system, depending on the lab. Windows XP Pro also is available as a dual-boot option in some labs, but we strongly encourage our students to use Linux. We are finding that, for the most part, our students have had little difficulty making the switch to using Linux as their primary desktop operating system. In many cases, they are enjoying it more than Windows because of the capabilities of KDE and GNOME to be customized to a user's individual taste.

To support software updates and patches between re-imaging, we have set up a centralized software/package repository on our main IT department server that mirrors the updates available on the Web. The lab servers at the remote sites then mirror a copy of the updates from the main server. The individual Linux clients in each lab then are

DESKTOP AND SERVER SOFTWARE

Server Software Configuration

OS Distribution: Fedora Core 2

Email:

- MTA: Postfix
- LDA/MDA: procmail + Dovecot IMAP
- MUA: SquirrelMail
- Filtering: SpamAssassin + ClamAV

User Database: OpenLDAP

User Management: phpLDAPAdmin

Package Management/Updates: APT, APT4RPM, Synaptic and YAM (for repository mirroring)

Web Serving: Apache

File Serving: Samba + NFS

Desktop Software Configuration

OS Distribution: Fedora Core 3

Desktop Environment: KDE 3.3 and GNOME 2.6

Office Suite: OpenOffice.org 1.1.x (We are testing CrossOver Office with Microsoft Office XP as a fallback option.)

Web Browser: Firefox

Package Management: APT + Synaptic

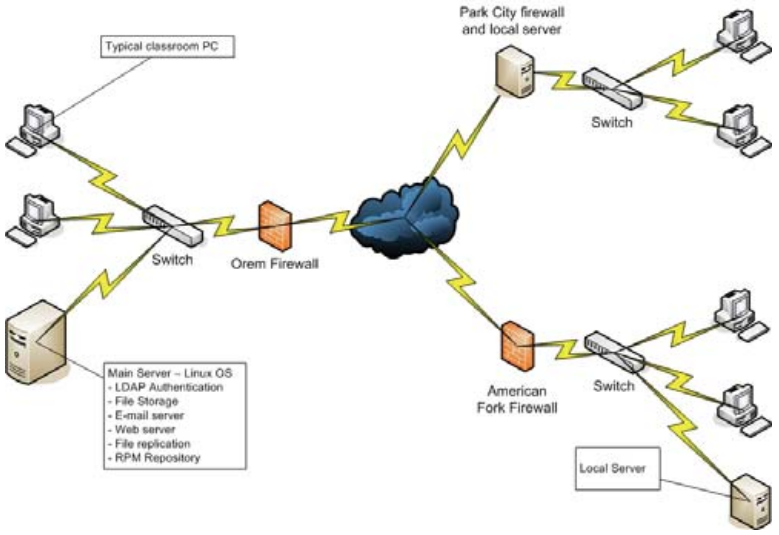


Figure 1. MATC Infrastructure

scheduled to pull the updates from their respective lab server using apt4rpm. This method allows for better bandwidth usage, particularly for our sites connected via T1, and a controlled set of software and patches.

What we have developed on our main campus is a robust infrastructure free from many of the problems related to Microsoft Windows. Samba has replaced Windows as our domain controller for those desktops that need to run Windows. The latest version of OpenLDAP has proven robust, and Samba, Apache, Postfix and PAM take full advantage of its capabilities. Refer to Figure 1 to see the complete MATC infrastructure.

Linux Use at Remote Campuses

The MATC main campus is in Orem, Utah, and we have a secondary campus in American Fork, Utah, approximately 10 miles away, that is connected to the main campus by a T1 line. We also have a lab in Park City, Utah, approximately 50 miles away, that we share through a partnership with the Park City School District.

During the 2003–2004 school year, the IT classroom at the American Fork campus was configured with a system running a Linux-based firewall and a separate server based on Fedora Core 2. That server hosted Linux distribution ISO images, pre-made VMware and VirtualPC images and files related to the courses the students were taking. It also provided some storage space for the students' work. The workstations ran Windows XP Pro, and all students logged in using a single user name and password local to the workstation.

Recently, during the 2004–2005 school year, the American Fork server has been upgraded to Fedora Core 3 with the latest versions of Samba, OpenLDAP and other software. The server now provides DNS and DHCP services, stores the home directories for the students that attend IT classes at that site and acts as the backup domain controller for the Windows clients in that lab. All course-related data is synchronized daily from the main server at our Orem campus using rsync. The firewall provides filtering and NAT masquerading and handles all of the Internet traffic for the workstations in that lab. Linux clients mount the home directories stored on the server using NFS. The main IT department server in Orem provides user authentication for all users.

All LDAP requests, generated either by the Linux clients or the Samba server on behalf of the Windows clients, are tunneled through OpenSSL to provide security. Although funneling all authentications back to our main campus is not an ideal solution, it has turned out to be surprisingly trouble free and highly reliable. We had to resort to this method because our attempts to use slapd to synchronize the LDAP servers between the Orem campus and American Fork campus often were interrupted due to circumstances beyond our control, such as high traffic volume and line unreliability. I must interject that we only provide computer services for our department and not the entire school. The result of these interruptions was the LDAP directory being out of sync between these servers.

The shared lab in Park City is located in the Park City Learning Center. As a member of the Park City School District (PCSD) network, the PCs and network are locked down tightly and administered by the highly competent PCSD IT staff. In discussions with the

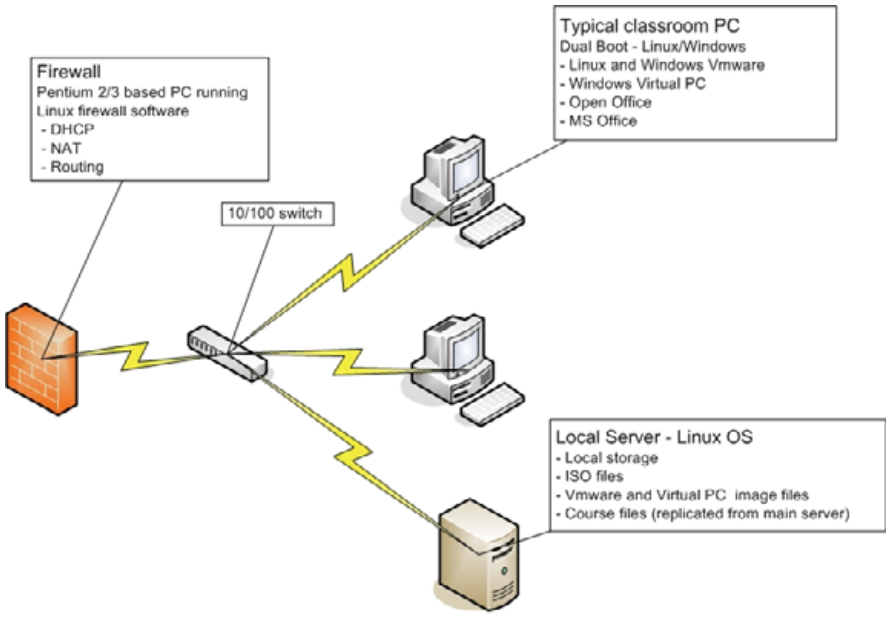


Figure 2. At each remote campus, one Linux file server offers local clients fast access to large files, while a Linux firewall also functions as a DHCP server.

school district IT staff, we reviewed some issues with security that had impacted our ability to teach IT courses during the 2003–2004 school year. We jointly decided on a plan to include a Linux-based server that would provide Network Address Translation (NAT), DHCP and routing, along with hosting Linux distribution ISO images, premade VMware images and files related to the courses the students were enrolled in, plus some storage space for our students. Seventeen lab PCs were imaged to dual-boot into Windows XP or Fedora Core 2. The PCSD IT staff, with the excellent help of Harold Hanson, provided a VLAN that isolated the Linux server, yet allowed us to change connections quickly so that Windows XP users still could authenticate into the PCSD Novell network. This enables us to provide authentication and other services for our students while they are in the lab, while not interfering with the PCSD IT staff's ability to maintain the network for their students.

The setup in the Park City lab is similar to that in the American Fork lab. The lab server provides file, print, name and address services as well as a mirror of the software and patch updates for the Linux clients. The main server at our Orem campus, using SSL to secure the transmissions, still provides all user authentications. Refer to Figure 2 to see our infrastructure template for remote campuses.

Conclusion

We have, over the past few years, developed a system based around Linux and open-source software that allows us to provide computing services for our students to enhance their learning experience in a manner that is both easy to maintain and simple to extend and replicate. It also has been quite inexpensive to implement, maintain and update. For those in the educational realm, cost is extremely important given the limited financial resources available to most secondary and post-secondary institutions. There is no doubt that more and more schools and businesses will move in a direction similar to ours as Linux and open-source software become more recognized and usable. This is one of the primary reasons that we are working so hard to provide Linux and open-source software training. All of our Linux courses have been influenced by our own experiences and include instruction in most if not all of the techniques that we have developed and refined.

Our journey with Linux and open-source software is far from over. We continue to refine and explore new areas to meet our current and future needs. Things we are working on and plan for the future include:

- Testing new groupware solutions, such as eGroupWare and OpenGroupWare.
- Testing Windows applications integration with Linux, using products such as CodeWeavers CrossOver Office.
- Testing and implementing new Linux distributions, such as Fedora Core 3 and future versions of Fedora.
- Increasing use of OpenLDAP as a central user and service information database.
- Using new features of OpenLDAP, including LDAP sync replication.
- Perfecting software updates from our mirrored apt repositories.
- Implementing other centralized administration and management techniques.
- Creating, revising and deploying hardware and software templates for labs and remote campus sites.

Joseph Ruffolo is a faculty member of Mountainland Applied Technology College. He has an IT career spanning 23 years, with expertise in software development, project management and system/software architecture. He currently teaches A+, networking and Linux courses. When he is not working, his hobbies include woodworking, rescuing neglected or abused animals and playing *World of Warcraft*. He can be reached at jruffolo@it.mountainlandatc.org.

Ron Terry currently is serving as the Department Chair of the Information Technology Department of Mountainland Applied Technology College in Orem, Utah, where he also teaches Microsoft, Linux and networking courses. When not teaching, Ron runs a consulting business that specializes in implementing and supporting Linux. He also travels to provide Linux training to businesses and other organizations. When he is not working with computers, as seldom as that may be according to his wife, his hobbies include camping, woodworking and music. He can be reached at rterry@it.mountainlandatc.org.

Ten Mysteries of about:config

Move along, nothing to see here. Some Firefox preferences are just too technical for end users. Oh, you're a *Linux Journal* reader? Come on in.

BY NIGEL MCFARLANE

The Firefox Web browser, built by the Mozilla Foundation and friends is a complicated piece of technology—if you care to look under the hood. It's not obvious where the hood catch is, because the surface of Firefox (its user interface) is polished to appeal to ordinary, nontechnical end users. This article gives you a glimpse of the engine. It explains how the Mozilla about:config URL opens up a world of obscure preferences that can be used to tweak the default setup. They're an improbable collection and therein lies the beauty of Firefox if you're a grease monkey or otherwise technical. At the end you'll know a little more about Firefox, but only enough to be dangerous.

Like any Linux-friendly piece of software, Firefox responds to preset environment variables. You can, for example, set the MOZILLA_FIVE_HOME or MOZ_PLUGIN_PATH variables prior to startup. They both work like the standard PATH, so no surprises there. The per-process space available for environment variables is, however, limited, and a simple textual concatenation of attribute-value pairs is a fairly inflexible way to store data. Firefox has a large set of runtime configuration options, and the environment isn't a suitable storage area.

Firefox configuration is stored in a small attribute-type-value database called the preference system. You can see a delta of this data set in the ~/.mozilla/firefox/*/prefs.js file. That file holds only the nondefault values selected by the user. The rest of the preferences either are unstated or stated in install files that are part of the standard install. For me, they're in /local/install/firefox/defaults/pref, because /local is my playpen of choice.

For a technical person, this system is a bit problematic because the full list of preferences doesn't appear anywhere on disk, and the standard way to change those preferences is to use the Firefox User Interface, which also is incomplete. That interface provides GUI elements (buttons, fields and check boxes) for only the most basic of the preferences available. Firefox isn't trying to be Emacs, after all. The rest of the preferences have to be dug up from elsewhere.

That other place is the special string about:config, which can be typed in the Firefox Location bar where the addresses of Web sites are entered. Briefly recall the taxonomy of W3C addresses: URIs (Universal Resource Identifiers) are a special case of IRIs (International Resource Identifiers). A URI either can be a URL (a Uniform Resource Locator) or a URN (a Uniform Resource Name). It's URLs that we see all the time. They consist of a scheme (typically http), a colon (:) and an address (x.org).

You can define your own scheme. Mozilla does that for "about", which is used to access internal browser resources. Try about:cache, for example. The config resource is a hook into the preference system. When you type about:config, you're navigating to a local resource just as you would navigate to a Web-based resource for an HTTP-based Web page. Figure 1 shows the results of loading the about:config resource.

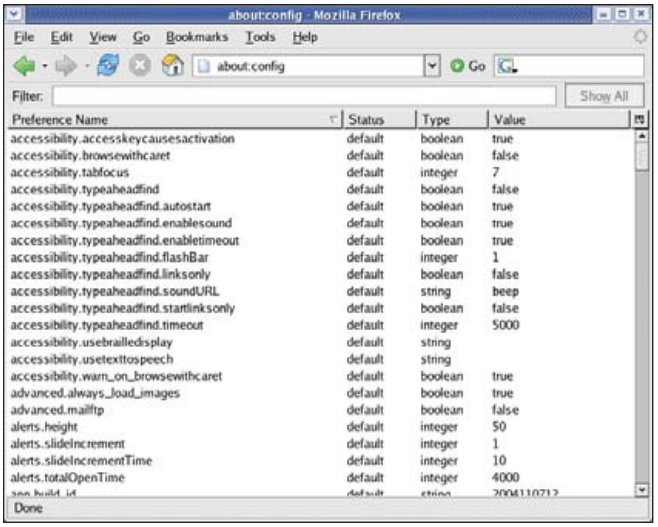


Figure 1. Firefox Showing the about:config Preferences

This preference listing is also a form. Right-click on any preference to modify it or to state a new preference. Shorten the display by entering some text in the Filter box if you want. Many Firefox extensions can provide alternate interfaces to about:config. Feel free to experiment with them.

Nothing is perfect, alas; about:config shows only preferences that *already* have been set or specified *anywhere*. It doesn't show preferences that have meaningful uses, which appear nowhere in the about:config list. To add a value for a new preference that doesn't appear, simply right-click anywhere in the main window, and select New from the context menu, then select the type of preference: string, integer or Boolean.

Without further ado, here's a tour of preferences to which the Firefox UI doesn't give you access. Some are unmasked by about:config; some are not. They're all relatively safe to experiment with. If you get into trouble, go back to about:config and unset the preference, or, in the worst case, shut down Firefox and delete the prefs.js file noted earlier. Everything said to this point also applies to other Mozilla products: the Mozilla Application Suite, Thunderbird and so on. Hesitate before deleting the Thunderbird prefs.js file. It contains important pointers to your e-mail.

Tune the Use of Firefox Caches

Here's a simple preference to begin with. You can explicitly set the size of the memory part of Firefox's Web cache. Here's the preference, which has a type of integer:

```
browser.cache.memory.capacity
```

Set it to the integral number of KB (Kilobytes) that you want as a maximum. By default, this preference is unstated and has a default value of -1, meaning "expand to fill available memory as required". That's a little like the Linux disk buffer cache. You might not want that if you're running OpenOffice.org and Firefox simultaneously and working both applications hard. If you do change this preference, you're going the way of Mac OS 9 and lower, where each application gets an explicit memory allocation. That could be a tuning burden if you go too far with it.

The Mozilla Web cache (both memory and disk) is akin to the function of servers like Squid. That is, both types of cache are smart about the use of HTTP headers for caching purposes. If you're in control of the local Web proxy, there's probably more value in a huge Squid cache than there is in a really big local disk cache. A bigger Firefox cache still gives some performance boost though. You can alter caching use through the integer preference:

```
browser.cache.check_doc_frequency
```

This preference affects when Firefox accesses the cache, not how the cache itself works. The cache caches Web content every opportunity it gets, but if Firefox fails to check it, such opportunities will come rarely. Set the preference to 0 for one check per URL resource per Firefox Web surfing session, 1 always to use the cache, 2 never to use the cache and 3 (the default) when the HTTP caching rules says it's a good idea to cache.

Disable Scripting Limitations

It's possible for a Web page to implement a denial-of-service attack on the browser user. All you need is a Web page that runs a heap of JavaScript in an infinite busy loop. Firefox can't accept user input when such intensive processing is going on. This integer preference causes script execution to halt if it goes on too long. The units are seconds, and the default is 5:

```
dom.max_script_run_time
```

You might have Firefox set up to do some tricky Web spidering. You might have it acting as a bot or running continuously as an unattended console. In any of those cases, set this preference to -1, and Web page scripts run forever unmolested.

The use of various asynchronous mechanisms, such as setTimeout(), support long scripting timelines in a normal Web page. There's no need for preference changes to support such things.

Disable Favicons and Site Icons

In the Firefox browser, a tab title, Location Bar URL or displayed bookmark can acquire a small image (an icon), which is displayed to its left. Usually it contains a brand mark for the site of the currently displayed Web page. You might not want this to happen. It makes your bookmarks file bigger, and (especially if you're on dial-up) it causes an extra HTTP request when the page first is visited. That request fetches the icon for display. These two Boolean preferences, both with a default of true, can be set to false to disable those fetches and the subsequent icon display:

```
browser.chrome.site_icons
browser.chrome.favicons
```

Set either one to false, and these icons are ignored. Bookmarks get the standard bookmarks icon, and elsewhere no icon at all appears.

You might ask, "Why are there two preferences?" Part of the reason is because these icons can be specified in two ways. You can put a 16×16 pixel Microsoft Windows ICO format icon at this URL: www.example.com/favicon.ico.

That icon will do for all pages on that site and is officially a Favorites Icon or favicon, to use Microsoft's term. Alternately, you can add an icon per page, using a <link> tag and any 16×16 ICO URL, like this:

```
<link rel="SHORTCUT ICON" href="/images/mybrand.ico" />
```

For some historical reason, that per-page use is called a Site Icon.

The other reason for two preferences has to do with parallel development streams in the Mozilla Project, the mess that is bookmark file formats and a shortage of time for trivial cleanup tasks. We're looking under the hood, remember.

Tune Up the Rendering Engine

If you have a drop of graphics programming in your blood, you might spare a kind thought for L. David Baron, Robert O'Callahan and company—the core developers of the Gecko rendering engine inside Firefox. Displaying a Web page is a fiendish compromise among standards, performance and subjective user perceptions. One of the most difficult constraints that Web pages impose is the need for incremental display. Show me the Web page as it arrives, not all at once at the end. This means constantly reflowing the displayed elements, which may be delivered out of order (a problem word processors don't have). Worse, such documents nearly always are network-delivered with unreliable timing.

To see the difficulty of this job, visit an image-intense Web site such as gamespot.com. Over broadband, the loading page jumps around in an agony of layout updates while chunky content is dumped into the browser in no particular order. On dial-up, the process is slower and more familiar, but the amount of layout labour is even larger, because there's more time to adjust each received fragment of page. Figure 2 shows the image-heavy GameSpot site, rendered while the images are still coming in.

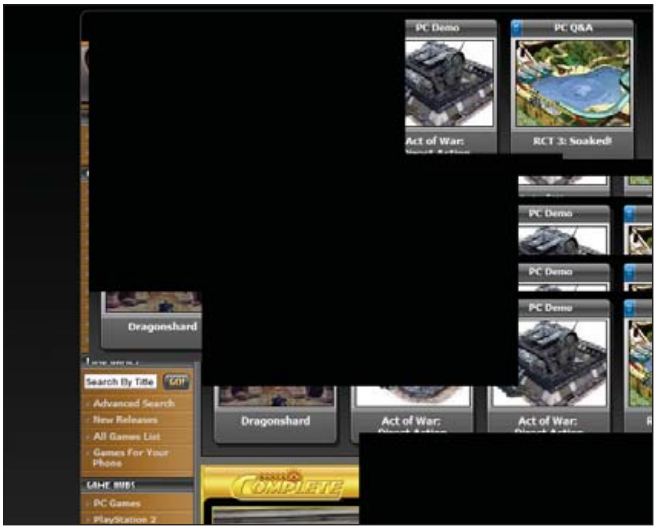


Figure 2. Close-up of a half-received Web site, jumping around as Firefox updates the layout.

Given this kind of problem, you can imagine, therefore, that all kinds of hidden tuning preferences are available—if you know where to find them. This isn't a tuning workshop, so here are two of the more interesting ones.

It's rare to want to tune down Firefox. (You should buy your nice mother a better computer.) It's more likely that you've got lots of CPU and video grunt and want to use it. You probably click the mouse more than 2,000 times a day. Theoretically, you can shave a quarter of a second off your response time—that's an extra coffee break a day—with this integer preference:

```
nglayout.initialpaint.delay
```

Set this to 0 (zero) milliseconds. It's set to 250 by default. When a Web page starts to trickle in to the browser, Firefox normally waits a bit after it has organised the page fragments in memory. It makes sense to bunch up the first few bits of content before attempting to show them. If you've got a quick eye, though, you can make it show what it's got ready straightaway.

Similarly, Firefox buffers up the incoming raw network content before it bothers to break those bytes down into something ready for display. That's another chunking process that saves the CPU but slows the output on a fast machine. Set this integer preference to, say, 5,000 (microseconds), and incoming network bytes are pushed to the display system much more quickly:

```
content.notify.interval
```

Doing so, however, makes Firefox work very hard scheduling updates in response to every drop of content. If you lower this value too much, that extra work merely results in the incoming data buffering farther back in the dataflow—perhaps behind a socket in the kernel—while Firefox thrashes around trying to complete a whole display update for every trivial character of text that appears. Lower the preference a bit at a

time, and watch the CPU with top(1), perhaps.

Penetrate the Mystery of Trusted Codebases

For a long time, Firefox, Mozilla and, before that, Netscape 4.x, supported this hidden Boolean preference:

```
signed.applets.codebase_principle_support
```

Normally, it's set to false—if you want you can set it to true. It's a poorly understood preference, so here's an explanation. First of all, the name is about as relevant as UNIX's /etc—it's so steeped in history that it's basically wrong. There are no applets at work; there's no Java at work. Mozilla has an amicable separation from Java, where Netscape 4.x was deeply wedded to that technology. Mozilla now handles its own security natively, in C/C++ code. It should be called signed.content.codebase_principle_support—one day, maybe.

This preference is used to assist developers who work with digitally signed content. It has no relation to SSL or to PGP/GPG. An example of signed content is a Web site or Web application bundled up into JAR format and digitally signed in that form.

Roughly speaking, two checks are done if digitally signed content arrives in Firefox. First, the digital certificate accompanying the content is checked against Firefox's list of known certificate authorities (CAs). If that much is fine, the maker of that content is considered authentic. Firefox then lets the content request extra privileges, ones that overcome normal browser restrictions, like access to the local disk. Usually that's done through JavaScript.

When those requests are made, Firefox throws up dialogs to the user. This is when the second check is done—it is done manually by the user. If the user agrees, the content can run with security restrictions dropped and your computer is exposed, or at least the currently logged-in Linux account is exposed.

For a developer, these checks are a nuisance. It's extra effort to buy (with real dollars) a suitable certificate for signing the content and set up the infrastructure. That should be necessary only when the site goes live.

Instead of using a real digital certificate to sign the content under development, suppose you use a dummy certificate—one that's not authentic. You can make a dummy certificate with the SignTool tool, available at ftp.mozilla.org/pub/mozilla.org/security/nss/releases. Next, you tell the browser that it's okay to accept such a dummy certificate. That's what the above preference does.

Setting this preference weakens only the first security check. You always have to perform the user-based check—at least Firefox offers to remember what you said after the first time. Setting this preference means that Firefox accepts a dummy certificate from any Web site, so use this only on isolated test equipment.

Read Your E-mail from Firefox

Finally, here's a simple way to set up Thunderbird access from Firefox. Set this Boolean preference to true to enable the mailto: URL scheme—the one that appears in Web page "Contact Me" links:

```
network.protocol-handler.external.mailto
```

An example of a mailto: URL is mailto:nrm@kingtide.com.au. Secondly, set this string preference to the path of the Thunderbird executable or to the path of any suitable executable or shell script:

```
network.protocol-handler.app.mailto
```

Digging out hidden preferences is a bit of a treasure hunt. Many are documented on Firefox-friendly Web pages, but the ultimate authority is the source code. Preference names are simple strings, and it's possible to create your own. Many of the extensions that can be added to Firefox dump extra preferences into the preference system. As long as the extension remembers to check and maintain those preferences, they have the same first-class status as the ones that have meaning for the standard Firefox install.

Remember, you always can save a copy of your prefs.js file before an experimental session with about:config and restore the saved copy if things get too weird. Happy config hacking!

Resources for this article: www.linuxjournal.com/article/8139.

Nigel McFarlane (www.nigelmcfarlane.com) is the Mozilla community's regular and irregular technical commentator focused on education, analysis and a few narrowly scoped bugs. Nigel is the author of *Firefox Hacks* (O'Reilly Media) and *Rapid Application Development With Mozilla* (Prentice Hall PTR).

Building a Bioinformatics Supercomputing Cluster

Bioinformatics tools running in the OSCAR cluster environment turned 17 recycled PCs into a system that improves performance for user queries.

BY JOSH STROSCHEIN, DOUG JENNEWEIN AND JOE REYNOLDSON

Bioinformatics is an increasingly important scientific discipline that involves the analysis of DNA and protein sequences. The Basic Local Alignment Search Tool (BLAST) was developed by the National Center for Biotechnology Information (NCBI) to aid scientists in the analysis of these sequences. A public version of this tool is available on the Web or by download. Because the BLAST Web site is such a popular tool, its performance can be inconsistent at best. The University of South Dakota (USD) Computer Science Bioinformatics Group decided to implement a parallel version of the BLAST tool on a Linux cluster by combining freely available software. The BLAST cluster, composed of old desktop PCs destined for surplus, improves searches by providing up-to-date databases to a smaller audience of researchers.

Our cluster project began with an implementation of the Open Source Cluster Application Resources (OSCAR). OSCAR was developed by the Open Cluster Group to improve cluster computing by providing all the necessary software to create a Linux cluster in one package. OSCAR helps automate the installation, maintenance and even the use of cluster software. A graphical user interface provides a step-by-step installation guide and doubles as a graphical maintenance tool.

WWW BLAST was created by NCBI to offer a Web-based front end for BLAST users and is the Web interface we selected for our BLAST cluster. WWW BLAST can be installed easily on a Linux machine running a Web server such as Apache.

Although WWW BLAST enhances the usability of our cluster, mpiBLAST enhances the performance. mpiBLAST was developed by Los Alamos National Laboratories (LANL) to improve the performance of BLAST by executing queries in parallel. mpiBLAST is based on the Message Passing Interface (MPI), a common software tool for developing parallel programs. mpiBLAST provides all of the software necessary for parallel BLAST queries.

Overview of a Query

A Web-based query form marks the beginning of a BLAST search on our cluster. By default, WWW BLAST does not support batch processing and job scheduling. Fortunately, OpenPBS and Maui are provided by the OSCAR software suite to handle job scheduling and load balancing. With this support, the cluster can handle a larger user audience more easily. OpenPBS is a flexible batch queuing system originally developed for NASA. Maui extends the capabilities of OpenPBS by allowing more extensive job control and scheduling policies.

Once the user submits the query, a Perl script provided by WWW BLAST is invoked. This script creates a unique job based on parameters from the query form. A job is a program or task submitted to OpenPBS for execution. Once the job has been submitted, OpenPBS determines node availability and executes the job based on scheduling policies. This job starts the local area multicomputer (LAM) software, which is a user-level, dæmon-based runtime environment. LAM is available as part of the OSCAR installa-

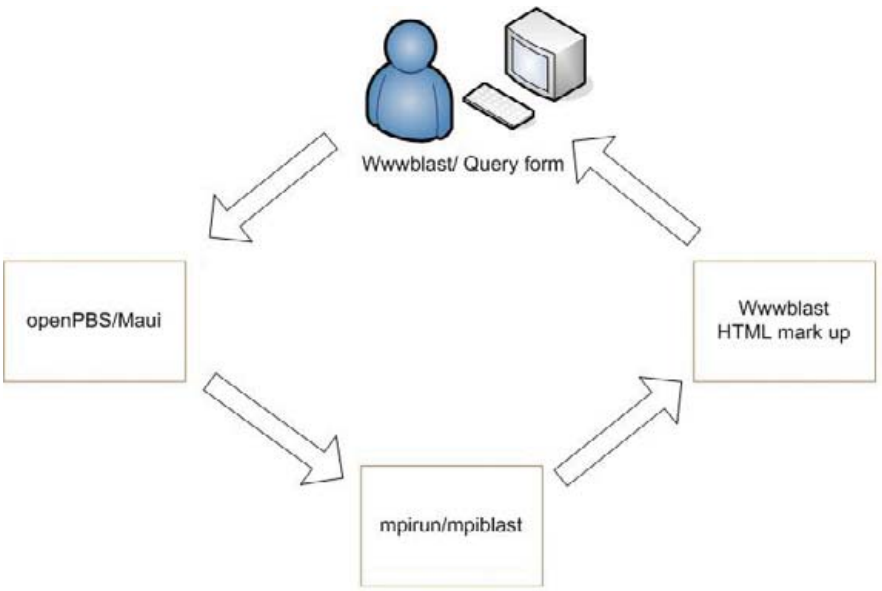


Figure 1. When a query comes in from the Web, WWW BLAST submits a job to OpenPBS. OpenPBS starts the job with mpirun and WWW BLAST formats the results.

tion and provides many of the services required by MPI programs. OpenPBS executes the job by utilizing the mpirun command, which executes the query on each node and gathers the results. WWW BLAST passes these results back to the browser, presenting the user with a human-friendly report (Figure 1).

Implementing cluster technology to perform parallel BLAST searches requires some software reconfiguration. Many of the tools we use work with default installations, but a parallel BLAST cluster requires extra configuration to get things running.

Using OSCAR to Build a Cluster

Clusters may be made up of a variety of PCs. The 17 nodes we used had 533MHz Intel Celeron processors, 256MB of RAM and 15GB of hard disk space—relatively low-end by today's standards. Using the exact same hardware setup for all of the nodes is not vital for cluster setup, but doing so does reduce the time and effort needed to install and maintain your cluster. Once all of the hardware is ready, you must choose a machine to be the head node. If you are not using identical machines, it would be beneficial to use the most powerful as the head node. Because all of the PCs we used have the exact same hardware configuration, the choice of the head node was arbitrary.

After you have obtained all the necessary PC hardware, you need to choose a Linux distribution. The OSCAR documentation lists all of its supported distributions, and Red Hat 9.0 was our distribution of choice. Installing Red Hat was pretty straightforward; we chose the default options. Because the OSCAR software depends on specific versions of OS packages, you should not install any updates once the installation completes. Of course, this has many security implications, which is why it is important to keep your cluster separated from the Internet by a firewall.

Once Red Hat was installed on the head PC, we downloaded the OSCAR 2.3.1 tarball. See the online Resources for installation documentation. We downloaded OSCAR into root's home directory, because OSCAR needs to be installed as root. Installing the OSCAR software was as easy as running the following commands:

```
tar -xvfz oscar-2.3.1.tar.gz
cd oscar-2.3.1
./configure
make install
```

After the installation completed, we needed to copy all of the Red Hat 9.0 RPMs to /tftpboot/rpm on our head PC. The OSCAR installation needs to install certain packages from this directory during its installation. We used the following command to copy the files:

```
cp /mnt/cdrom/RedHat/RPMS/*.rpm /tftpboot/rpm
```

Once all of the RPMs are copied, the OSCAR installation can begin. OSCAR provides a graphical installation wizard for the installation. Substitute the name of your private network Ethernet adapter; ours was eth1:

```
cd $OSCAR_HOME ./install_cluster eth1
```

Figure 2. The OSCAR installation wizard lets you deploy, configure and test cluster software.

After a few moments, the OSCAR installation wizard begins to load. This wizard provides a graphical user interface and an intuitive eight-step process to complete the cluster setup (Figure 2). The only variation in our installation procedure was to set the default MPI implementation to LAM/MPI instead of MPICH. We chose LAM because it is needed for mpiBLAST to execute properly.

Clicking on step 2, Configure Selected OSCAR Packages, displays a small dialog (Figure 3). From there you can select the Environment Switcher button and choose LAM as the default for the installation (Figure 4).

We followed the remaining steps as described in the OSCAR documentation to build and install a disk image for the nodes. Once all of the nodes were installed and tested, we downloaded and installed mpiBLAST.

Figure 3. Click on Config... to change environment to LAM.

Figure 4. Select LAM for default environment.

Installing mpiBLAST for Parallel Searches

We downloaded mpiBLAST and installed it according to the documentation provided in the README file. We created symbolic links for mpiblast and mpirun in our $PATH, and no further configuration of mpiBLAST was necessary.

Once mpiBLAST was installed, we needed to download a database to search. For mpiBLAST to execute properly, the database needs to be in the FASTA format. NCBI offers an index for all of its databases on the NCBI Web site, and that index lists a FASTA subdirectory containing all of the databases in FASTA format. We downloaded a copy of the nr database to /usr/local/mpiBLAST/db/, an NFS-shared folder set up during the installation of mpiBLAST. mpiBLAST provides the mpiformatdb command, which formats the database into segments; the number of segments depends upon the number of nodes in the cluster. mpiformatdb places the segments it creates into a shared directory. This directory is defined in mpiblast.conf during installation and is utilized by all mpiBLAST programs. Here is an example of formatting the database:

```
# /usr/local/mpiBLAST/bin/mpiformatdb -N 16 -i nr
```

Here, -N specifies the number of database segments—usually the number of nodes in the cluster—and -i specifies the name of the database file to format. In this example, the nr database is formatted into 16 individual segments. mpiformatdb does not copy the segments to the nodes, so a significant amount of overhead is incurred while each node copies its database segment during the first query. Each node copies a segment only once. If the segment is erased from the node, it is copied again during the next query.

To simplify management of the cluster, we wrote a script to download the newest version of a database, format it with mpiformatdb and distribute it to the nodes by executing a simple BLAST query. We scheduled this script with cron to run on a weekly basis. Once we were able to execute BLAST queries in parallel, we added the Web-based front end from WWW BLAST.

Configuring WWWBlastwrap.pl

mpiBLAST provides command-line BLAST searches and includes two files for interaction with a Web-based front end, blast.cgi and WWWBlastwrap.pl. These files are configured to work with WWW BLAST. So our next step was to download WWW BLAST into the /var/www directory, creating the /var/www/blast/ directory. Several configuration changes had to take place for WWW BLAST to submit BLAST searches for parallel execution.

WWW BLAST provides its own directory for databases. Because we are using mpiBLAST to format the databases, we had to point WWW BLAST's db/ directory to mpiBLAST's. We then made the db/ directory in blast/ a symbolic link to the db/ directory for mpiBLAST.

WWW BLAST provides a file called blast.cgi that executes a BLAST query. mpiBLAST provides a replacement blast.cgi that executes a parallel BLAST query by way of

WWWBlastwrap.pl. WWWBlastwrap.pl is a Perl script that creates a query for mpiBLAST to execute. WWWBlastwrap.pl creates this query in the form of another Perl script, populating it with the parameters from the Web form. This script is submitted to OpenPBS. WWWBlastwrap.pl serves several functions, including parsing the parameters of the form, creating a script to be submitted to the cluster through OpenPBS for job queuing and load balancing and returning the BLAST search results in a browser-friendly format.

We needed to make some changes to WWWBlastwrap.pl, however, to allow it to operate correctly in our environment. The first change that we made was to the global variables $scratch_space and $MPIBLASTCONF. These two variables are used throughout the life of the script. $scratch_space holds the absolute path to a directory containing temporary files used during a query. $MPIBLASTCONF holds the absolute path to the directory containing the mpiBLAST configuration file. Both of these directories were set up during the installation of mpiBLAST. We set the two variables as follows:

```
$scratch_space="/usr/local/mpiBLAST/shared/scratch";
$MPIBLASTCONF="/usr/local/mpiBLAST/etc/mpiblast.conf";
```

The next change involved changes to a series of if statements. These statements hard-code the NUMPROC environment variables for the nt, nr and pdb databases. Because the databases need to be preformatted by mpiBLAST, the number of processors used per query is constant. We changed the default number of 20 to 16, which is the number of processors we use:

```
if($data{'DATALIB'} eq "nt"){
    $data{'NUMPROC'} = 16;
}
```

Farther down in the script, the ValidateFormData subroutine is defined. This subroutine ensures that the user has selected a valid database/program combination and produces a 500 server error if a valid combination is not selected. We changed the subroutine to allow the tblastx program to execute queries on the nr database by making the following change:

```
#### BEFORE ####
# Must be applied to a nucleotide database
if($data_ref->{'DATALIB'} ne "nt"){

#### AFTER ####
# Must be applied to a nucleotide database
if($data_ref->{'DATALIB'} ne "nt" ||
   $data_ref->{'DATALIB'} ne "nr"){
```

Later on, the script creates a string of command-line arguments for mpiBLAST and stores them in the variable $c_line. We needed to change the value passed to the -d option, which

tells mpiBLAST which database to search. By default, WWWBlastwrap.pl concatenates the number of processors to the database name and passes the result to the -d option. So if our database was named nr and we had 16 processors, it would pass nr16. Presumably this is done to allow more than one version of a database to be searched, that is, nr16 for a 16-segment database and nr8 for an 8-segment database. You either can name your databases in that manner or modify the script. Because we only ever have one version of a database, we chose to modify the script, removing the number of processors from the database name. The code changes are summarized below:

```
#### BEFORE ####
# Create the command line to pass to mpiBlast my
$c_line = "-d $data_ref->{'DATALIB'}" .
          "$data_ref->{'NUMPROC'} " .
          "-p $data_ref->{'PROGRAM'} " .

#### AFTER ####
# Create the command line to pass to mpiBlast my
$c_line = "-d $data_ref->{'DATALIB'} " .
          "-p $data_ref->{'PROGRAM'} " .
```

When running test queries, we received several `lcl|tmpseq_0: Unable to open BLOSUM62` warnings in the OpenPBS error log. Pointing the environment variable BLASTMAT to the location of the BLAST matrices clears up these warnings, so we made the following change:

```
#### BEFORE ####
print SCRIPTFILE '#PBS -e '.
"$data_ref->{'ERROR_LOG_FILE'}\n\n";
print SCRIPTFILE 'if(-e $ENV{PBS_NODEFILE} ){'."\n";

#### AFTER ####
print SCRIPTFILE '#PBS -e '.
"$data_ref->{'ERROR_LOG_FILE'}\n\n";
print SCRIPTFILE '$ENV{BLASTMAT} = '.
'"/usr/local/ncbi/data";'."\n";
print SCRIPTFILE 'if(-e $ENV{PBS_NODEFILE} ){'."\n";
```

We encountered the final alteration toward the end of the script in the HtmlResults subroutine. The code that directs the user to the results uses a default base URL, which almost certainly is not what you want. Changing the base URL to point to our Web server allowed the client's Web browser to display the results of the BLAST query:

```
#### BEFORE ####
print "Location: https://jojo.lanl.gov/blast/".
"BlastResults/$results_file\n\n";

#### AFTER ####
print "Location: http://domain_name/BlastResults".
"/$results_file\n\n";
```

Conclusions and Results

Our local cluster is able to search an up-to-date database with fewer concurrent users and better overall throughput times than is the NCBI Web site. Simple wall-clock time trials were performed using our cluster and the NCBI Web site. We used eight simple queries consisting of protein and DNA sequences. A timer was started after submitting a query from the Web site and stopped once the results were displayed in the browser window. Trials on the NCBI Web site were performed at various times throughout the span of two weeks. All eight trials were averaged and compared to the cluster's times. The purpose of timing the query from the point of submission until the results are displayed was to observe times that an actual user would incur. On average, the cluster took less time to complete a query.

Figure 5. Our cluster, consisting of 17 recycled PCs, improves response times for users' queries.

Resources for this article: www.linuxjournal.com/article/8140.

Josh Stroschein (jstrosch@usd.edu) currently is pursuing his undergraduate degree in Computer Science and Criminal Justice. Josh is working on the cluster project through a grant at USD. He also works for Walton Internet Solutions, based in Vermillion, South Dakota.

Doug Jennewein (djennewe@usd.edu) is a research analyst in Computer Science, and he has been with the USD since 1998. He received his Master's degree in Computer Science from USD in 2004. Doug's main research interest is high-performance computing.

Joe Reynoldson (jreynold@usd.edu) is the research computing manager/instructor for the Computer Science Department, and he has been with USD since 1994. He received his Master's degree in Computer Science from USD in 1997. Joe teaches topics in Perl, systems management and Web development.

LETTERS CONTINUED FROM PAGE 7

ThinkPad T41), and Don has reviewed the HP NX5000. Doc has used Apple PowerBooks in the past. I've felt that trying to use Linux on a laptop was always a "hack". Never really worked right. And, I've tried it more than once with more than one distro. So, which is the way to go? A PowerBook with a Power PC Linux Distro? An EmperorLinux hybrid? DIY and hope for the best?

--
Karl

That's a tough one. We're impressed with how all of HP's hardware was working out of the box, that the system price didn't include a proprietary OS "tax" and that we could get hardware and software phone support with one call. EmperorLinux will sell you a working Linux install on a wide selection of name-brand systems. Don and Jill both have IBM ThinkPads from them, but Don's new Linux load is definitely on the DIY side. LinuxCertified also sells x86 Linux laptops, and Terra Soft will pre-install on Apple hardware, but we haven't reviewed either one.—Ed.

Names, Please

The fact that most of the comments on the new Web site are posted by Anonymous makes reading the threads very difficult, particularly when contrary views are espoused. A good example is the thread following this article: www.linuxjournal.com/article/7719. It's hard to know who is saying what and who is new to the conversation. How about enforcing the entry of a name, even if it isn't a real one?

--
Nick Rout

Xen, Please

I would like to read about virtualization using Xen in *Linux Journal*. Are you considering publishing an article about this issue?

--
Ralf Strandell

Yes, we are.—Ed.

Project Directory Permissions Tip

Good articles on setting up NIS [see Alf Wachsmann's articles in the February, March and April 2005 issues]. There is one additional step that can be done when setting up the project area. After running the commands in the article, set the group sticky bit, which will cause new files to inherit the group ownership from the directory:

```
chmod g+s /projects/X/
```

That eliminates the need to use newgrp so much. But still, users need to be reminded about their umask. In this case, I'd recommend either 007 (no world access) or 002 (no world write access). I also like to set the owner of the /projects/X/ directory to be the point of contact for the project.

--
Jon Miner

We welcome your letters. Please submit "Letters to the Editor" to ljeditor@ssc.com or SSC/Editorial, PO Box 55549, Seattle, WA 98155-0549 USA.

If You Don't Believe in DRM, It Can't Hurt You

"Keep your management off my digital rights" isn't merely a slogan for freedom-lovers. It's a smart IT decision. **BY DON MARTI**

The last time I talked with Martin Fink, HP's Vice President of Linux, the problem on his mind was digital rights management (DRM) and if it could ever be compatible with free software. It's a puzzling question, but Martin, like everyone else in the Linux business, can find better problems to work on.

DRM is any technology that selectively disables features or affordances of a program or device in order to control use of a copy of information by the owner or authorized user of the copy. Think "unrippable" CDs for the home market or, on the office side, e-mail software that lets someone who sends you mail disable your forward or print function.

A coin-operated jukebox is not DRM, and `chmod 600 my-secret-file.txt` on a multiuser system is not DRM. Those technologies exclude only unauthorized users. DRM starts when the technology begins nit-picking about what you can do. For example, "play only on example.com's media player" is DRM. Certainly such a system helps example.com hang on to its customers, but there's no demand for it.

In this crazy business of ours, every once in a while, companies go into a frenzy to sell technology that doesn't work to customers who don't want it. In the 1990s, did customers want overpriced UNIX from bickering vendors or stable-any-day-we-promise Windows NT? Sorry, neither one works for us. Support Linux, please. Or on-line services. AOL or Compuserve? We'll take the Internet, thanks.

When I met Intel VP Donald Whiteside a while back, he summed up the IT industry party line on DRM. IT companies have to do DRM in order to work with the "consumer electronics", movie and record companies who put together media standards. He said computer DVD drives are so locked down because the DVD Copy Control Association would have refused to license the DVD format for computer drives otherwise.

Mr Whiteside is too modest about the IT industry's negotiating position. People started shifting their leisure time from big-budget TV productions to the slow-loading, frustrating Internet long before the big entertainment industry made it there. And the big copyright holders make pie-in-the-sky DRM demands, but a little Internet Movie Database search of actual DVD release dates shows a different story in the real world.

The five top grossing movies for 1998, before the DVD descrambling story broke, took an average of 367 days after first release to come out in DVD format. By 2000, disinfecting DVDs was common knowledge in tech circles, but the top five movies for 2000 actually came out sooner after theatrical release—252 days.

The story is the same for before and after the "DVD X Copy" application for Microsoft Windows—from 190 days in 2002, before it came out, to 160 days during 2003 when it was available. Yes, the movie industry has an infringement problem, and they might even be releasing DVDs sooner than they would want in order to compete with infringing copies. But the DRM features of the DVD itself are a pointless sideshow.

The other hyped-up use for DRM is at the office. Deploy DRM and you can keep employees from forwarding embarrassing e-mail to the media. That sounds like the answer to network-illiterate managers' prayers, but if it's juicy enough to leak, it's juicy enough to write down and retype. Bill Gates of Microsoft, in an interview with gizmodo.com, tried to pitch DRM using the example of an HIV test result, which is literally one bit of information. If you hired someone untrustworthy enough to leak that but unable to remember it, you don't need DRM, you need to fix your hiring process.

When I talk to working IT professionals, the trend is to open up information "behind the firewall" at a company—not lock it down. People aren't worried about how to DRM-ize everything. Instead, I'm seeing enterprise Wikis. "Enterprise Wiki" still sounds funny, but companies with lots of trade secrets are rolling them out. "Edit this Page" adds value, and DRM has the opposite effect.

Even the mighty US army is adopting discussion-friendly social software. Doc Searls sent me a link to Dan Baum's great *New Yorker* article about Companycommand.com and Platoonleader.org, which two army captains started as a side project to exchange advice outside the normal channels. The army promoted them and brought the sites in-house.

What if I'm wrong, DRM really is the Next Big Thing, and the herd of IT vendors is right for the first time in history? Network effects practically guarantee that one DRM system will be a global standard. Picking the winner, though, depends on unpredictable DRM-circumvention efforts by security researchers worldwide.

And when even a PC operating system can be an "essential facility" to be regulated on antitrust grounds, DRM that actually worked would be too much power for governments to let anyone else have. Win the DRM war, and the prize is becoming a regulated industry like the pre-breakup AT&T. Martin Fink doesn't want Linux users to miss the DRM boat. I'll miss that ship of fools any day.

Resources for this article: www.linuxjournal.com/article/8127.

Don Marti is editor in chief of *Linux Journal*.

The Straight Talk People℠
SINCE 1991

ABERDEEN

WHO SAYS YOU CAN'T BE EVERYTHING TO EVERYONE?

ABERDEEN STIRLING 51T

1U 14" Extra-Short Depth

- 14 inch depth allows back-to-back racking
- Intel® Pentium 4® Processor with 800MHz FSB and Up to 2MB Cache
- Intel E7221 Chipset with 64-Bit Support
- Up to 4GB DDR2 533 Memory
- Up to 1 x 400GB SATA Hard Drive
- 260W AC Power Supply w/PFC
- **5-Year Limited Warranty**

Starting at **$999**

ABERDEEN STIRLING 104T

1U Short Depth 4 SATA

- Intel® Pentium 4® Processor with 800MHz FSB and Up to 2MB Cache
- Intel E7221 Chipset with 64-Bit Support
- Up to 4GB DDR2 533 Memory
- Up to 4 x 400GB (1.6TB) Hot-Swap SATA Hard Drives
- 300W AC Power Supply w/PFC
- **5-Year Limited Warranty**

Starting at **$1,225**

ABERDEEN STONEHAVEN A141

1U Dual Opteron™ 4 SATA/SCSI

- Dual AMD Opteron™ Processors w/HyperTransport and 1MB Cache
- AMD 8000 Series Chipset w/64-bit Support
- Up to 16GB DDR-400 Reg. ECC Memory
- Up to 4 x 400GB (1.6TB) Hot-Swap SATA or 4 x 300GB (1.2TB) Hot-Swap SCSI Drives
- 400W AC Power Supply w/PFC
- **5-Year Limited Warranty**

Starting at **$1,935**

ABERDEEN STIRLING 111

1U Dual Xeon™ 4 SATA/SCSI

- Dual Intel® Xeon™ Processors with 800MHz FSB and 1MB Cache
- Intel® E7520 Chipset with 64-Bit Support
- Up to 16GB DDR2-400 Reg. ECC Memory
- Up to 4 x 400GB (1.6TB) Hot-Swap SATA or 4 x 300GB (1.2TB) Hot-Swap SCSI Drives
- 560W Power Supply (Optional Redundant)
- **5-Year Limited Warranty**

Starting at **$2,025**

ABERDEEN STONEHAVEN A261

2U Dual Opteron™ 6 SATA/SCSI

- Dual AMD Opteron™ Processors w/HyperTransport and 1MB Cache
- AMD 8000 Series Chipset w/64-bit Support
- Up to 16GB DDR-400 Reg. ECC Memory
- Up to 6 x 400GB (2.4TB) Hot-Swap SATA or 6 x 300GB (1.8TB) Hot-Swap SCSI Drives
- 460W Hot-Swap Redundant Power Supply
- **5-Year Limited Warranty**

Starting at **$2,875**

ABERDEEN STIRLING X312

3U 12 SATA Storage Server

- Dual Intel® Xeon™ Processors with 800MHz FSB and 1MB Cache
- Intel® E7520 Chipset with 64-Bit Support
- Up to 16GB DDR2-400 Reg. ECC Memory
- Up to 12 x 400GB (4.8TB) Hot-Swap SATA Hard Drives
- 650W 2+1 Redundant Hot-Swap Power Supply
- **5-Year Limited Warranty**

Starting at **$3,945**

ABERDEEN STIRLING X524

5U 24 SATA Storage Server

- Dual Intel® Xeon™ Processors with 800MHz FSB and 1MB Cache
- Intel® E7520 Chipset with 64-Bit Support
- Up to 16GB DDR2-400 Reg. ECC Memory
- Up to 24 x 400GB (9.6TB) Hot-Swap SATA Hard Drives
- 950W 3+1 Redundant Hot-Swap Power Supply
- **5-Year Limited Warranty**

Starting at **$5,925**

ABERDEEN STONEHAVEN A124

1U Quad Opteron™ HPC

- Quad AMD Opteron™ 800 Series Processors
- AMD 8000 Series Chipset w/64-bit Support
- Up to 32GB DDR-400 Reg. ECC Memory
- Up to 2 x 300GB (600GB) SCSI Hard Drives
- 500W Power Supply
- Ultra Cool with Superb Air Flow
- **5-Year Limited Warranty**

Quads Starting at **$8,465**

ABERDEEN ABERNAS 106

1U 640GB NAS

- Intel® Pentium 4® Processor with 800MHz FSB and 1MB Cache
- 512MB Low Latency DDR-400 Memory
- 4 x 160GB (640GB) Hot-Swap SATA Hard Drives (Available with 1.0TB or 1.6TB)
- 300W AC Power Supply w/PFC
- Microsoft Windows® Storage Server 2003
- **5-Year Limited Warranty**

Starting at **$2,495**

ABERDEEN ABERNAS 120

1U 1TB Hardware-RAID NAS

- Separate Internal Operating System Drive
- DVD Recovery of OS
- 3ware Hardware-RAID with Hot-Spare
- 512MB Low Latency DDR-400 Memory
- 4 x 250GB (1TB) Hot-Swap SATA Hard Drives (Up to 1.6TB)
- Microsoft Windows® Storage Server 2003
- **5-Year Limited Warranty**

Starting at **$3,595**

ABERDEEN XDAS 253

2U and 3U 8/12/16 Drive SATA-to-SCSI RAID Direct Attached Scalable Storage

- Daisy Chain Units for Scalability
- OS and Host Independent
- Up to 16 x 400GB (6.4TB) Hot-Swap SATA Hard Drives
- **5-Year Limited Warranty**

Starting at **$5,375**

ABERDEEN ABERSAN I300

3U 2TB iSCSI SAN Storage

- Intel® Xeon™ Processor at 3.0GHz with 800MHz FSB and 1MB Cache (Dual Processor Option)
- 512MB DDR400 SDRAM (Up to 4GB)
- 2GB PC2-3200 ECC-Reg. DDR2 (Up to 16GB)
- 8 x 250GB 7200 RPM Hot-Swap SATA
- 760W Triple Redundant Hot-Swap Power Supply
- **5-Year Limited Warranty**

Starting at **$9,995**

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

Hear Yourself Think Again!

WhisperStation™

Originally designed for a group of power hungry, demanding engineers in the automotive industry, WhisperStation™ incorporates dual 64-bit AMD Opteron™ or Intel® EM64T™ processors, ultra-quiet fans and power supplies, plus internal sound-proofing that produce a powerful, but silent, computational platform. The WhisperStation™ comes standard with 2 GB high speed memory, an NVIDIA FX1300 PCI Express graphics adapter, and 20" LCD display. It can be configured to your exact specifications with either Linux or Windows, and specialized applications including Mercury's AmiraMOL,™ PathScale's EKO Compiler Suite or the Intel Performance Tools. RAID is also available. WhisperStation™ will also make a system administrator very happy, when used as a master node for a Microway cluster! Visit www.microway.com for more technical information.

Experience the "Sound of Silence".
Call our tech sales team at 508-746-7341 and design your WhisperStation™ today.

AMD 64 Opteron

intel inside XEON

Microway®
Technology you can count on℠