■ HOTTEST NEWS ■ LATEST REVIEWS ■ BEST TUTORIALS ■ YOUR PROBLEMS SOLVED

LINUX FORMAT

THE UK'S BEST-SELLING LINUX MAGAZINE!

TURN OVER FOR YOUR FREE LINUX PRO MAGAZINE

STOP SPAM!

PLUS PREVENT VIRUSES • HALT HACKERS • GET PROTECTED

We show you how to secure your system – SAVE your servers and desktops with our definitive guide!

ON THE DISCS:

All the latest anti-spam, anti-virus and firewall software

START PROTECTING YOUR SYSTEM NOW!

CREATIVE COMMONS

Licensing isn't restricted to software – Open licenses can apply to *all* your art & text **p60**

PROGRAM IN SDL

...and save the universe from killer fish with our tutorial **p74**

LIVE DISTROS!

On your coverdisc and on test inside p36

LATEST DISTROS ARRIVE!

SUSE **p20** Fedora Core 2 **p22**

MAKING A NOISE WITH AUDACITY

Our Hottest Pick – find out how to make music the easy way with our four-page beginners' tutorial p66

GIMP 2 ON TEST

Rewritten from the ground up, is it still the best art software? **p18**

KDEVELOP & KDE

New series starts this issue! **p88**

DVD issue also available
Printed in the UK
LXF54 JUNE 2004
£5.99
9 771470 423019 06
(£5.99 outside UK and Republic of Ireland)

www.linuxformat.co.uk

Can Open Source get too good?

The most excellent thing about Linux – and Open Source software generally – is choice. It may be confusing to newbies, but choice really makes a difference in the Open Source world. If you want a mail client, you can choose from half-a-dozen. If your mailserver doesn't support the feature you want, you can bet there's another that does. Choice coupled with openness fosters a climate of invention and creativity that gives all users so many benefits.

But, imagine this hypothetical situation. There becomes a need for a new type of application. Suddenly from nowhere, a project delivers on about 80 per cent of what people might reasonably want from such a piece of software. Other putative projects look elsewhere for a new idea to latch onto, or join up with this new bright star. But over time, the application doesn't fulfil its early promise. It still lacks that final 20 per cent, but progress is slow. It has so successfully stifled competition by being so good, so early on, that nobody even conceives of starting over again. What progress that is made is slow and the developers become complacent – there's no need to hurry feature X, because people aren't going to stop using the software in the meantime.

Just as in the world of proprietary software, the lack of choice can be a dangerous thing. Now, there may be a case for saying that some projects might be exempt from this. You could argue that, were KDE to disappear overnight, the GNOME project would continue at the same cracking pace regardless, and *vice versa*. Perhaps that's true, but I wouldn't like to try it and find out...

Nick Veitch EDITOR

LIVE CD DISTROS – everything you need to run Linux on any computer, any time, anywhere **p36**

GET PROTECTED – don't let viruses, hackers and spam spoil your day **p48**

TROUT WARS – learn games programming in SDL and you could save the world! **p74**

AIMS OF THE MAGAZINE

***Linux Format* is a magazine dedicated to Linux and the Open Source community. We aim:**

- To provide the most accurate, unbiased and up to date information on all things Linux.
- To promote the use of Linux in business and the home, for servers and on the desktop.
- To support the Open Source community by providing a resource of information, and a forum for debate.
- To help all readers get more from their Linux experience by providing insightful and useful tutorials.

MEET *LINUX FORMAT'S* TEAM OF WRITERS...

Andrew Channelle
LXF's newshound and the Beginners' best friend, Andy shows us how to make the most of *Audacity* this month.

David Coulson
Our *Answers* guy is a networking, sysadmin and security guru with lots to tell us about firewalls this issue.

Mike Saunders
Found some Open Source software you like? Tell us what is is, and Mike can include it in *LXF*'s *Hot Picks*.

Jono Bacon
KDE and web developer, writer and musician, he kicks off a *KDevelop* series and explains the Creative Commons.

Paul Hudson
Taking our cover feature literally, he now *protects* himself with a great big sword that he bought off some Internet nutter.

David Cartwright
Veteran journalist and Linux consultant, he knows his stuff when it comes to real-world Linux usage.

Hoyt Duff
LXF lifer and leading Linux author, Hoyt is a master of finding easier and better ways to do things with Linux.

Marco Fioretti
In his quest to run Linux on every PC he encounters, Marco shares his expertise on Linux Live CD distros.

Michael J Hammel
Professional *GIMP* artist who pens (or pencils) our Open Source graphics tour-de-force dives into GIMP-Perl.

Dave Cross
Coder Dave concludes his Perl tutorial series showing us how to use Template Toolkit with plain text and HTML.

CONTACT US

Letters for publication:
lxf.letters@futurenet.co.uk

Subscriptions/back issues:
subs@futurenet.co.uk

Technical help/Ask the Experts:
lxf.answers@futurenet.co.uk

Disc problems:
lxf.support@futurenet.co.uk

General enquiries:
linuxformat@futurenet.co.uk

Website: **www.linuxformat.co.uk**
Or send your letters to:
***LINUX FORMAT*, Future Publishing,**
30 Monmouth Street, Bath, BA1 2BW
Phone: 01225 442244
Fax: 01225 732295

More contact info on p102 & p114

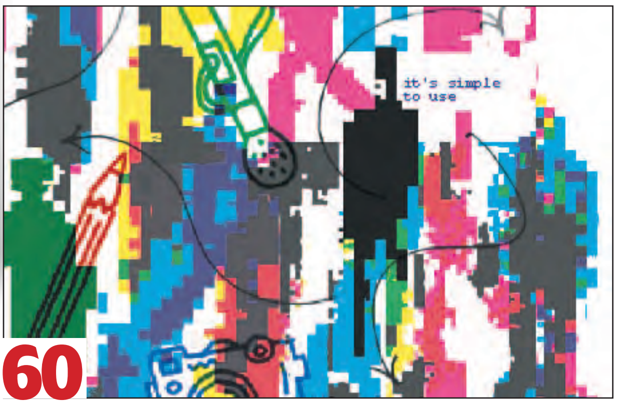
Contents

Welcome to another jam-packed issue of *Linux Format*, your guide to all things Linux!

LXF54 June 2004

COVER HIGHLIGHTS

HOTTEST NEWS ■ LATEST REVIEWS ■ BEST TUTORIALS ■ YOUR PROBLEMS SOLVED
LINUX FORMAT
THE UK'S BEST-SELLING LINUX MAGAZINE!
TURN OVER FOR YOUR FREE LINUX PRO MAGAZINE
STOP SPAM!
PLUS PREVENT VIRUSES • HALT HACKERS • GET PROTECTED
We show you how to secure your system – SAVE your servers and desktops with our definitive guide!
ON THE DISCS: All the latest anti-spam, anti-virus and firewall software
START PROTECTING YOUR SYSTEM NOW!
CREATIVE COMMONS Licensing isn't restricted to software – Open licenses can apply to all your art & text p60
PROGRAM IN SDL ...and save the universe from killer fish with our tutorial p74
LIVE DISTROS! On your coverdisc and on test inside p36
LATEST DISTROS ARRIVE! SUSE p20 Fedora Core 2 p22
MAKING A NOISE WITH AUDACITY Our Hottest Pick – find out how to make music the easy way with our four-page beginners' tutorial p66
GIMP 2 ON TEST Rewritten from the ground up, is it still the best art software? p18
KDEVELOP & KDE New series starts this issue! p88
LXF54 JUNE 2004
www.linuxformat.co.uk



60

'Like eating more healthily, making regular backups is something we all genuinely intend to do at some point'

42 Mike Saunders has some hot tips...

'Tiger is supposed to be faster at starting up and should consume much less memory into the bargain...'

28 Another landmark release for Java?

LINUXPRO

PHP PRO the advantage of professional tools

OPEN JAVA? the arguments for and against

LINUX EXPO warming up for the event of the year

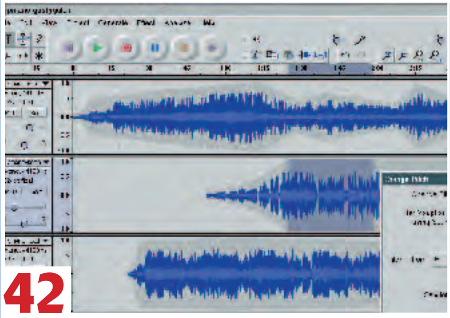
FREE FOR ALL why big business and free software can get along fine...

AND MUCH MORE!

23 All the power of Linux, but with no knowledge required!

06

36



42

88

Coverdiscs

A DVD or 3 CDs packed full of the latest Linux goodies **105**

CDS A, B AND C

KNOPPIX The King of the Live CD distros – a complete Linux system that runs direct from the CD or can be installed on HD

DAMN SMALL LINUX A complete desktop distro in only 50MB

SLAX Slackware based Live distro

STORIX PERSONAL Backing up is easy (and important) to do!

THE GIMP 2 If you don't believe our review, try it out for yourself – don't forget the tutorial on page

AND Hot picks and relevant mag files for the security feature and other tutorials.

DVD

LIVE DISTROS A total of seven different live distros, ready to burn from your DVD. A full, customised new version of Knoppix runs directly from the DVD

MANDRAKE MOVE Yes, that's one of the Live distros – we know you've been waiting for it **PLUS GNOME 2.6** and plenty more!

On the CD and DVD

Please read the coverdisc instructions on page 105 before installing from coverdiscs!

Newsdesk

● Saving the planet with Greenpeace ● Linux Personal Video ● Network installation ● Security ● HP Linux groupworking PCs ● Lindows launches Linspire ● Linux PDA ● Linux in Hollywood

THE US$2 BILLION APOLOGY

Sun and Microsoft – an unholy alliance?

Sun and Microsoft, bitter industry combatants for over 20 years, have decided to bury the hatchet; and speculation is mounting that this unholy alliance may be intending to turn its fire on Linux. The two companies have shelved all ongoing litigation, and in return Microsoft has offered Sun a US$2 billion apology. However, it is thought the deal and its provision for closer co-operation on patent issues will be a forerunner to a new legal offensive by Microsoft to stall adoption of Linux in the lucrative OS market.

On the other hand, some have suggested it may presage a new understanding of the reality of a market in which proprietary software must 'play nice' with Linux on the server and, increasingly, the desktop.

Announcing the deal, Sun Chairman Scott McNealy said this was the beginning of a new era of co-operation between the two companies, one that would preserve rather than restrict customer choice.

"This agreement will be of significant benefit to both Sun and Microsoft customers. It will stimulate new products, delivering great new choices for customers who want to combine server products from multiple vendors and achieve seamless computing in a heterogeneous computing environment. We look forward to this opportunity – it provides a framework for cooperation between Sun and Microsoft going forward."

Bitter enemies put aside their differences in the name of commerce. But how long will the smiles last?

Microsoft's Steve Ballmer said it was not the end of competition, but that there was a recognition that *"cutting-edge R&D and intellectual property protection are the foundation for the growth and success of our industry."*

"This is a positive step forward for both Sun and Microsoft, but the real winners are the customers and developers who rely on our products and innovations," he said.

Pundits have suggested that Microsoft's reason for dealing was the recent legal difficulties with the European Union, while Sun's was a little more prosaic: money, and lots of it. Both, in the words of eWeek's Stephen Vaughan-Nichols can envision Linux and Open Source *"eating their lunch."*

In the past, large-scale IT players such as IBM, HP, Sun and Microsoft have accrued vast patent 'armouries' with the result that – akin to the nuclear arms race during the Cold War – the industry remained at stalemate. Microsoft is rumoured to hold a number of patents which relate directly to the Linux kernel, but the possibility of reciprocal action by the likes of Sun and IBM on patents they hold has prevented the Redmond giant using those particular weapons in the fight against competitors. But with all else failing, and the SCO case not denting the growing popularity of Linux, Microsoft and Sun may feel the time has come to bring out the big guns.

In tandem with this Microsoft news, Sun also confirmed the ascendancy of Jonathan Schwartz to the role of company President. Schwartz, seen as a supporter of Linux within the company, stressed Sun's continued support of the OS and announced an agreement that will see Sun Java Desktop-based PCs on sale in Wal-Mart stores in the US.

Chris Stone, Linus Torvalds and Jack Messman take questions from the floor at Brainshare 2004.

INCREASED LINUX ADOPTION

Novell Brainshare one year on...

Novell's 2004 Developer Conference marks the first anniversary of the company's big push into the Linux sector. In that time, it has purchased two of the most high-profile Linux 'brands', become enmeshed in the SCO/IBM tussle and signalled its intention to build its future on Open Source software.

As expected, there were a number of developments in the camp as this year's Brainshare event got underway.

Despite its ownership of GNOME developer Ximian, Novell Chairman Jack Messman affirmed that SUSE would continue to be a KDE focussed distribution into the near future. The ultimate goal, though, is to migrate Linux to a single desktop, something Messman says the combined 'development muscle' of SUSE and Ximian are more than capable of. Chris Schlager, VP of R&D at SUSE, says it's not a case of merging the two DEs, but taking the best of both worlds to make a unified desktop. The process has already started, he says, with the Ximian Desktop 2 for SUSE project *"but you'll see the first major results of this effort in the next versions of SUSE Linux, which will be released toward the end of the year."*

Linus Torvalds, the godfather of Linux, turned up at the opening of the Brainshare conference to bestow his blessing on the Novell move into FOSS. During a Question and Answer session, Torvalds was asked what the next big thing in Linux would be. Gesturing toward Messman and his depute Chris Stone, he said: *"I only work with the kernel itself. You guys can maybe be the next big thing in Linux on a different scale."* Torvalds also highlighted the dangers that patents pose to the entire industry.

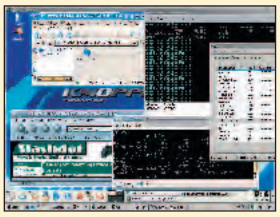
"Software patents, where non-technical issues can be used to stop development, stop people from doing what they want to do and can do – that, to me, is the biggest threat," he said.

In addition to finalising the US$50 million investment by IBM, Novell also unveiled a new partnership with Hewlett Packard to support SUSE Linux right across the enterprise. Martin Fink, HP's Vice President of Linux, said they were the first major technology vendor to offer enterprise customers a single, proven Linux solution from desktop to data centre. *"Our relationship with Novell is based on a joint commitment to innovation, low-cost platforms, and total customer experience,"* he said.

Finally, Chris Stone revealed that Novell has begun the process of migrating all its desktop systems to *OpenOffice.org*, with a full move to Linux pencilled in for 2005.

NEWSBYTES

■ **MandrakeSoft** has begun to offer the Official release of Mandrake 10.0 to Club members. By the time you read this, both boxed products and the download edition should be available. The Community Edition, featured on last month's LXF cover discs, has been updated with slightly newer kernel and, insiders say, a lot of bug fixes. In all, MandrakeSoft suggests that the new development regime – which essentially turns Mandrake's early adopters into beta testers – has been a success, addressing some of the quality assurance issues of the past.



■ **Cooperative Linux** is a new project aimed at making OS choice less problematic. The system, nicknamed CoLinux, allows a full implementation of Linux to run within a Windows NT-based session. Developers say the Open Source application offers some serious advantages over traditional virtualisation software such as *VMware*. It should work (with some small modifications) with any Linux distribution, and the team have had considerable success running Knoppix on the system. http://www.colinux.org/

■ **The Mozilla Organisation** has begun the beta testing process for the next major release, 1.7, which will become the next 'stable' branch of the suite. Changes in this release include facilities to prevent sites blocking the context menu, better theme support for *GTK+* themes, and support for MSN Authentication in the mail client. Performance is said to have improved substantially; start up times are speeded up by eight per cent, and a similar increase in page rendering speed, with a reduction in binary size of five per cent. www.mozilla.org/

■ **Wyse Technology** is adding support for Linux-based appliances from Neoware Systems to its Rapport management software. In addition to Neoware and its own systems, Rapport supports thin client hardware from HP and a wide range of PDAs, scanners and other hand-held devices.

■ **Pixar**, the company behind *Toy Story* and *Finding Nemo*, has released an update to its industry standard *Renderman Pro Server*. The upgrade includes support for both Mac OS X and 64-bit Linux for the first time, as well as versions for IRIX, 32-bit Linux and Windows NT.

Jono Bacon

The founder of UK Linux, KDE developer and all-round nice guy, Jono is also a musician who's tunes have been featured on Slashdot.

COMMENT

'Parasite' vs gift culture

"At a recent computing trade show, I took part in a special Linux debate with a panel of notable figures from Linux-related businesses (Sun, IBM, HP, Red Hat, Novell *etc*) and Brad Tipp from Microsoft. I had seen Brad at a previous show and was intrigued to see him again at this debate.

In one part of the proceedings, Brad Tipp referred to the R&D culture in Linux as being "parasitic." This definition referenced the way that one company may develop a new product or technology and license it under the GPL, but then competitors can adopt it if useful. He made the point that some companies may *never* contribute back R&D, and hence be parasitic in their culmination of their technology. I can imagine that Open concepts are difficult to fully understand and reason from a typically proprietary background; justification of R&D is difficult to define when the result can be utilised instantly by competitors.

Where I think the point is missed is that Open Source is essentially a gift culture. Red Hat has value in the way that it puts its technology together and supports it; this is the definition of a distributor. What is important for all distributors is that the technology as a whole needs to be constantly improved and expanded. This pushes Linux on as a technology platform in which collaboration improves the industry as a whole.

Linux as a business culture is an interesting combination of sharing, peer review, improvement, gritted teeth marketing and brand value. I hardly think this process is in any way parasitic; I would better describe it as 'symbiotic competition.'"

LINUX AND CHARITIES

Saving the planet with a SAN

IBM has provided SAN facilities for Greenpeace.

Greenpeace, the charitably funded environmental action group, recently completely an IT upgrade, opting for Linux in three out of four of its critical systems. IT director Steve Thomson said after a process of evaluation the organisation chose Linux for web, application and general office server, but stayed with Windows for the mail server. *"We looked at the two and Linux just seemed more flexible, probably cheaper and probably more sensible for these particular bits,"* he said.

The idea, he said, was to simplify operations which, in the past, had relied on a combination of UNIX, Windows and Novell systems. The core of the system will be a newly acquired storage area network (SAN) based on IBM's FAStT600 storage array used for archiving, running a *DB2* database, office applications and storing *MS Exchange Server* data.

UK retail giant John Lewis has also signalled a change in IT policy by migrating its core business applications to Linux. The company said the planned migration of the formerly Windows-based application, which took place over the Easter bank holiday, went smoothly and would enable the company to cope with the growing number of transactions. Like Greenpeace, the John Lewis partnership opted for IBM hardware for the system, in this case a new eSeries mainframe.

GREENPEACE
Finding Chemo
Features
Jury trial awarded in Bush vs. Greenpeace
Recent News
More news
Canadian harp seal hunt: largest ever
11:37

Three out of four isn't bad. Greenpeace will be running web, application and storage on Linux.

Linux Web Watch/

Linux Personal Video

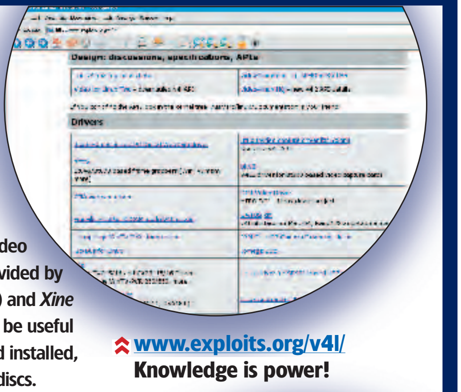
With the BBC under threat of commercialisation, isn't it time you let some Linux apps help your properly organise your personal television viewing?

Deciding that the aging *LXF* bedroom PC should do more, this month we have been investigating tools for turning it into a humble telly.

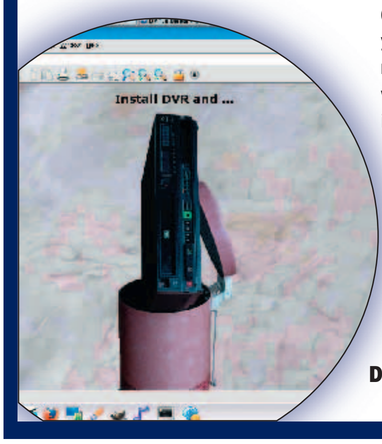
Well, humble is not really appropriate, because with something like DVR (http://dvr.sourceforge.net/) it should be possible to turn a suitably equipped Linux PC into a sophisticated recording and playback device. But first we need a TV card that works. One of the best sites for information BEFORE you hit the stores is the *Video4Linux* project (http://www.exploits.org/v4l/). Here you'll find a rundown of everything needed to watch and record TV and will turn even the newest new user into a savvy buyer.

There are a lot of nascent PVR (personal video recorders) for Linux, but one of the more elegant – on first viewings at least – is *Freevo* (http://freevo.sourceforge.net). *Freevo* promises to turn our PC into a real mediacentre-style workhorse complete with proper EPG (Electronic Program Guide). Video facilities in this project are provided by *MPlayer* (www.mplayerhq.hu) and *Xine* (http://xinehq.de), so it would be useful to have these downloaded and installed, or get them off *LXF52*'s coverdiscs.

Finally, having hours of telly programmes recorded to the hard disk is fine for local viewing, but what happens when we want to watch on the big TV in the lounge? Well, getting video onto disk is not the easiest process, but the effort should begin at the home of the *dvdrtools* application (www.nongnu.org/dvdrtools/), a fork of the more common *cdtools*. But, you've guessed it, *dvdrtools* has been tailored towards DVD burning.



www.exploits.org/v4l/
Knowledge is power!



« http://dvr.sourceforge.net/
Dispose of your VCR responsibly!

» http://freevo.sourceforge.net
MPlayer + *Xine* = Linux PVR!

NEWSBYTES

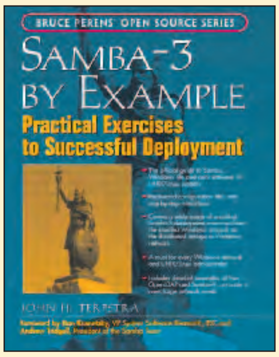
■ The release of **GNOME 2.6** was delayed briefly after an intruder managed to infiltrate the project's main server, which houses the **www.gnome.org** site. Owen Taylor said that beyond the intrusion "No additional damage has been discovered; at the current time we are cautiously hopeful that the compromise was limited in scope."

■ **Hewlett Packard** has launched a new multi-user Mandrake-based PC that offers facilities for four people to work simultaneously on the same machine. The HP441 system is built around a 1.8Ghz Celeron and 256MB RAM, but includes four graphics cards and inputs for a quartet of keyboards and mice. In addition to the core Mandrake system, each user will also have access to KDE, *Mozilla, Evolution* and *OpenOffice.org,* as well as a range of education software.

■ **IndeView** is a new presentation system that can take *KPresenter* or *OpenOffice.org Impress* presentations and make them viewable on PCs where native software is not present. The developers say the GPL is the perfect tool for ensuring presentations are pixel perfect regardless of the host system.

■ **MyHosting.com** is offering a **free Linux hosting plan** to its Windows customers in response to customer demand. *"We still stand behind and support the Windows platform but we must listen to our customers. And they tell us that applications drive their choice of hosting provider, not the Operating System,"* said CEO Tony Yustein.

■ **Prentice Hall** has published a new book covering Samba network administration. *Samba 3 – By Example* (ISBN 0-131-47221-6) is part of the *Bruce Perens Open Source* range, and was written by long-time *Samba* developer John Terpstra. The book is accompanied by a CD-ROM containing samples, scripts and all the tools mentioned in the text. Prentice Hall will also make the 'source code' of the text available as part of the project's official documentation. It should be available on the official *Samba* site by the time you read this.


BRUCE PERENS' OPEN SOURCE SERIES
SAMBA-3 BY EXAMPLE
Practical Exercises to Successful Deployment
JOHN H. TERPSTRA

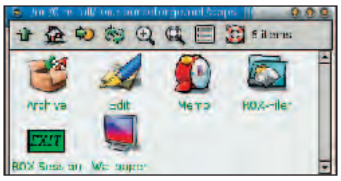
NETWORK FILESYSTEM

The end of installation woes?

Zero Install, an offshoot from the ROX project, is attempting to address the common perception that Linux software is difficult to install. In fact, the project takes the process to the extreme, making software installation not just easy, but unnecessary.

The project is built around a network filesystem from which you run applications, and once an application is 'installed', subsequent launches make use of a cache, ensuring that the application runs as smoothly as if it was installed locally. Moreover, the project leaders suggest users will never suffer a dependency error as software and libraries are fetched dynamically when needed.

The website of this project is at **http://rox.sourceforge.net/phpwiki/index.php/ZeroInstall**, though ironically, installation of the application wrapper itself is not a simple operation!



Installing applications is as simple as double-clicking an icon.

WINDOWS AND LINUX VULNERABILITIES

Distro vendors address the security issue

Distribution vendors Mandrake, Debian, Red Hat and Novell/SUSE have joined forces to rebut the conclusions of a recent survey which found Linux to be less secure than Windows. The report, the vendors suggest, seriously skewed results by not taking into account the seriousness of a threat involved in a reported vulnerability. The four Linux vendors were all involved in the initial research after an approach from analyst group Forrester.

The study would collect data about vulnerabilities and the speed at which they were tackled over a one year period. In their collective statement, the vendors said that problems are assessed according to severity. Windows flaws were regarded as more serious, while Linux vulnerabilities took longer to patch.

"This severity is used to determine the priority at which a fix for a vulnerability is being worked on weighed against other vulnerabilities in our current queue. Our users will know that for critical flaws we can respond within hours. This prioritisation means that lower severity issues will often be delayed to let the more important issues get resolved first."

"Even though the Forrester report claims so, it does not make that distinction when it measures the time elapsed between the public knowledge of a security flaw and the availability of a vendor's fix. For each vendor the report gives just a simple average, the 'All/Distribution days of risk', which gives an inconclusive picture of the reality that users experience. The average erroneously treats all vulnerabilities as equal, regardless of the risk. Not all vulnerabilities have an equal impact on all users."

The full statement is available at **http://lxer.com/module/newswire/view/9986/index.html**

Linux flaws take longer to fix...

Hoyt Duff

One of 800 Hoyts living in the USA, he runs a fishing pier when not shouting affectionately at his many computers.

COMMENT

A Great Thing

"As I co-author the current *Fedora Core Unleashed* book, people make some interesting assumptions about me.

I just met a chap in a computer store sporting a Tux lapel pin and fretting over a purchase. I offered help, and the discussion wandered. He effervesced with his admiration of Linux and had chosen Fedora Core as his distro, and was elated he a had an 'expert' at hand to answer a few questions. But he had questions about GNOME, *Nautilus*, and *Evolution* – apps I don't routinely use, as I am simply used to other alternatives. He was aghast that I would use Fedora Core and *not* use the default apps: if RH is a recognised brand and industry leader, wouldn't it choose – and offer – the best?

I explained that Linux *in toto* offers you the flexibility to choose yourself, and many different reasons to choose. I gravitate to a choice based on selecting the best tool for the job – but not always: I'm also resistant to change and frequently make the task fit the tool rather than the 'proper' way. Often, the choices are so similar that its foolish to invest energy in remaking them.

While Red Hat, Fedora Core and Debian are GNOME-centric; SUSE, Mandrake and Knoppix favour KDE, other WMs and other apps work just as well on those flavours. Why? It's all Linux, no matter which cute mascot is on the desktop.

For me, part of the thrill of learning is exploration, but not everyone wants this. The choice of defaults by vendors insures that, when focused solely on the task, the tools will be up to the job. Distro defaults should be considered merely a convenience. A default usable desktop AND the choice to change it – that's a great thing about Linux."

FIGHTING MICROSOFT'S FUD

Lindows battles with MS are Linspirational

Michael Robertson says the techniques used by Microsoft are a threat to free competition.

After a long-running fight with Microsoft over the generic-ness of the word 'Windows,' Lindows boss Michael Robertson has relented and re-branded his company's OS product as Linspire. Robertson said that, while the legal battle would continue in the USA, it wouldn't be fair – in the light of Microsoft's European legal wins – to restrict the ability of resellers across the world to trade freely.

"First, Microsoft is asking courts to levy massive fines just because users in certain foreign countries can access our website. We can't afford to lose a battle on this topic. Second, we have partnerships with hard disk, motherboard, laptop and desktop companies that want to ship our products to their worldwide market. If they are restricted to certain countries, then it impedes their business and therefore ours."

After failing to secure a legal victory in America, Microsoft went to courts in Sweden, the Netherlands and Denmark, to press its trademark case against Lindows with opposite results. Microsoft asked judges to impose a fine of 100,000 Euros for every day the Lindows website was available to users in the affected countries. At the time, Robertson said the process set a very bad precedent for any enterprise.

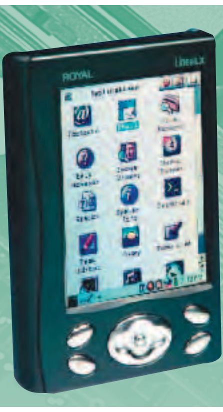
"We may be headed toward a world in which rich companies can shop around, repeatedly searching for a friendly court that is willing to ban content, ideas, products and choices with which they may disagree," he said. He said changing now, when Linux was growing and most users had yet to encounter Lindows, would be easier than if the name had become more well-known.

In honour of the name change, Lindows will be offering free downloads of the newly updated OS using BitTorrent at the company's newly named website, which is now **www.linspire.com**.

Embedded Linux News

● **Intervideo**, which recently launched its Linux-based InstantOn technology for PCs, has expanded the product to include both PVR and DVD Burning facilities. InstantOn is designed to offer a dual-boot option on media-centric computers and, the company claims, can start up in less than 10 seconds, making it suitable for purposes where a full blown PC is not needed. The PVR tools include time-shifting, pausing live TV and recording programmes for later viewing.

● After a very long wait, **Royal** has announced availability of its Linux-based PDA. The $399 device is based on Motorola's 200MHz DragonBall processor equipped with 64MB SDRAM, 32MB Flash memory and Trolltech's Qtopia DE and personal information manager suite. The Linea LX is able to work in landscape or portrait mode and has a quite snazzy snap-on keyboard. **www.royal.com/**



Royal Linea LX: snap-on keyboard not shown.

● The **Simputer** project, which proposed a Linux-based device to bring the communications revolution to India's masses, has launched its first piece of hardware. The collaborative project's first piece of kit combines the features of a music player and organiser, while featuring sophisticated handwriting recognition software. The device also has an on-screen keyboard capable of working in both Hindi and Kannada. For more information, see **www.simputer.org/**

By Pamela Jones, www.groklaw.net

SCO NEWS

Baystar, which invested US$20 million in SCO, has asked for its money back, citing breach of the original agreement. Dion Cornett, an analyst for Decatur Jones Equity Partners-Soleil said: *"I don't think BayStar is going to be very successful in getting their money back. It's very difficult for a private equity investor to force a redemption on a company that doesn't want to redeem. But it makes it very difficult for SCO to raise future financing."*

- We heard from the overworked judge in the Red Hat case – SCO got trounced. This is the second against SCO in court: the judge refused SCO's request to dismiss Red Hat's case.
- The judge, while ruling against SCO on its dismissal request, decided to put the case on temporary hold, asking both sides to keep her posted on the progress of the IBM case. She would like the issue of whether IBM "stole" code to be decided before she wades into the arguments Red Hat raises. The day before Judge Robinson's order, SCO asked for another five-month delay in the IBM case for discovery.
- SCO also asked that IBM's counter-claims be tried in a separate trial, and IBM responded with the suggestion that they will be asking for summary judgement when the discovery process is finished. They expect most, if not all, the issues in the case will be quickly resolved without trial.
- The next big decision looms in the Novell matter. A hearing is scheduled for May 11 on SCO's Motion to Remand (asking the judge to send the case back to local Utah courts) and on Novell's Motion to Dismiss SCO's complaint in the 'slander of title' dispute. To be found guilty of slander of title requires proving that you knew that your claim was false. It is obvious from Novell's documents that it has a position it believes to be true.
- The Novell case, in my opinion, is the true heart of everything involving copyrights. If SCO loses here, it's unlikely they will able to sue end-users for copyright infringement.
- SCO has been unusually quiet this month, indicating that the judge has asked parties not to try their case in public; except in Germany, where VP Gregory Blepp declared that he had proof of copyright infringement; but all he had was the same discredited materials SCO showed at SCOForum. http://www.groklaw.net/article.php?story=20040416151214305#c118401

Linux is making inroads into the mid-range broadcast market.

LINUX AT THE MOVIES

Discreet brings Smoke to Linux

Discreet software is attempting to consolidate its position in the film and effects industry with the release of the *Smoke6* editing and finishing system on Intel-based IBM/Linux workstations. Strong demand from the mid-range broadcast market is said to be behind the release. The software was previously available on Silicon Graphics Octane and Tezro hardware. Porting *Smoke6* to Linux was eased by a collaboration with Nvidia on drivers for the Quadro FX 3000G high-end graphics card. Jeff Brown, general manager of workstation product management at Nvidia, said the partnership had brought about a huge leap in professional visual computing.

"We're resetting industry expectations because it proves that a rock-solid, turnkey editing and finishing system can run on a Linux-based workstation," he said.

IBM's Bob Lenard said that the product could not have come at a better time. *"We're seeing post-production and broadcast companies strongly embrace Linux migration because of its speed, networkability and cost-effectiveness,"* he said. Turnkey systems start at US$68,000.

Meanwhile, using the same Sony-Panavision 24p high-definition digital cameras used by George Lucas on his *Star Wars* prequels, *Starship Troopers: Hero of the Federation* will become the first feature film to utilise the Rave/HD capture and editing system for a complete production. Based on an Open Source embedded hardware/software combination, RaveHD is geared toward the high-end film and graphics industries and uses standard protocols to ensure it can easily drop into any studio's workflow.

Embedded Linux brings aliens to life for Starship Troopers 2.

David Cartwright

David Cartwright is an IT consultant who specialises in providing Linux systems and solutions.

COMMENT

Free trade-offs

"It happened to me again this week. Usually it's: *"Oh, you don't want to use Windows, you want to use Linux."* This week it was: *"Oh, you don't want to use Linux, you want to use OpenBSD."*

On my planet, you implement computer systems in such a way as to give the most appropriate trade-off between solving the business problem and spending resources (people, time and money). So you identify the task that needs to be done, you identify the applications that will allow you to do it, you put each application in the context of (a) the technology you have; (b) the funds available; and (c) the training and capital equipment requirement, and you make a decision. Sometimes the decision will lean toward the platforms that you already have, in order to keep the training and equipment costs down; other times, you'll find an application that's so much better than the competition that you don't mind implementing a completely new platform and training/employing people to support it, because the business benefit of the app is so significant.

What you don't do is say: *"XYZ is my pet platform, so we need an application that runs on XYZ"*.

Standardisation of protocols, the ubiquity of IP networks, and integration technologies like LDAP mean that intercommunication of different platforms is generally not an issue. As app vendors write more Linux versions, that's a bonus for people like me who make money out of implementing (among other things) Linux systems for business. But let's turn 2004 into *International Stop Arbitrarily Telling Me What Operating System To Use* Year."

Mailserver

Share your opinions, right wrongs and demand justice by writing to *Linux Format*. Drop us a line at: ***Linux Format*, Future Publishing, 30 Monmouth Street, Bath BA1 2BW** or email: **lxf.letters@futurenet.co.uk**

Flex & Bison

I just wanted to let you know that I am reading and following the very interesting articles on writing compilers. I hope you continue with the series of articles. I commend you on showing how to write *real* code – this is not often done in this sort of magazine and it is a very refreshing thing to see.

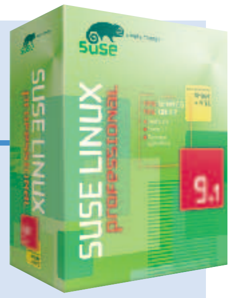
To me, the most useful part has been the tutorial on *Bison/Flex*, but I am following along with interest as you go through loops and conditionals. Please keep it up. This is, for me, one of the best articles in what I consider to be the best Linux magazine.

Two small points that I would like to see addressed in the articles.

- Please get rid of the **strdup()** statements in the parser. Please use the STL string class instead
- Please show how to use *Bison/Flex* within a purely C++ environment rather than reverting to C code.

★Letter of the month

This month's winner receives a boxed copy of **SUSE Linux Professional 9..1**



Difficult display?

I use a Mitsubishi Diamond View LXA520W LCD monitor; it is a few years old but still gives excellent service. When I originally installed SUSE 8, I had some initial problems getting the display 'just right' and thought it was an odd problem that was specific to my particular setup. I now run SUSE 9, and again I had to go through the same unusual problem during my installation of the upgrade.

Recently however, a friend of mine who also runs an LCD monitor on SUSE 9 encountered precisely the same problems as I originally met and solved, and so I thought it would be beneficial for me to share them with *Linux Format* and its readers as I don't think I have ever seen this particular situation documented.

My friend had successfully installed his SUSE 9 and it was behaving perfectly – or so we thought... A few days later, he did an online upgrade (part of which included an update of his video drivers) and to his horror, he found that his cursor had also vanished. Curiously, the cursor was 'still there' – even if invisible, and as he moved his mouse, the icons on the desktop would light up and change as the 'vanished arrow' moved over them. He could even start software programs – just could not see the cursor. Obviously, he knew it was a video card/driver problem but had no idea where to start.

My friend tried everything, but could not restore the cursor and happened to mention it to me just before taking the ultimate step of re-installing. I then asked him whether or not there was a narrow gap (about half a centimetre wide) visible down one side of his desktop where there appeared to be no display, and he replied that indeed there was. I was then almost certain my friend was suffering from the same problem that I had solved a couple of years earlier, and so it proved to be.

Fixing the panel

The solution was for him to open *YaST2* and select Hardware followed by Graphics Card and Monitor. I requested that he make a careful note of the current settings of monitor, resolution, videocard *etc*, because ultimately, they would not be changed. Next, I requested him to go through the procedure of resetting those exact variables with the aim of getting to the test screen where *YaST2* allows the user to check that the video settings are correct. This he did and reached the test screen, which has two sets of 'Arrow' buttons, permitting coarse and fine alterations to the position and size of the screen. This screen also showed the narrow blank strip at the side. He was next required to very carefully use these buttons and adjust the display width so that it the strip disappeared and the test screen covered the monitor screen fully. On his first use of a width adjustment button his exclamation was: *"I've got my cursor back... and the space has gone."* He now saved this new setting, retreated back out of *YaST2* and was delighted to see his cursor present once again, the side blank strip had vanished – and there was no need for any re-installation. The display has worked perfectly ever since.

I have also noted that another detail seems to also tie in with the above: When the video display is not set perfectly (as with my friend), the X window (which appears just before the log-in screen is presented) always has lines and 'disturbances' across its 'odd cross-hatch herringbone pattern,' even if the window manager display on top is quite okay. If the video display is set perfectly, the X window is absolutely stable and appears perfectly. It might be my imagination but I also feel that the window manager (KDE in my case) always appears with better clarity when the X window is perfect.

I have never seen previously published details of this very unusual 'problem' which I would bet occasionally turns up with other LCD monitors in other distributions. I hope it is of value to others.

Dr Tony Young, *Blackbutt, Queensland, Australia*

Thank you very much for this very useful piece of advice. If anyone can explain the 'magic' behind both this problem and its solution, we'd be even more impressed. However, for your efforts so far we're awarding you the star prize this month, a nice new SUSE Professional 9.1 boxed distro.

Compiler writing , PHP & SDL coding: please give us *even more* feedback!

I understand that *Flex* and *Bison* can run to produce a parser class, which can then be instantiated. This would eliminate any need to use global variables and would make it cleaner.
Michael, *via email*

Paul Hudson writes: We're glad you like the tutorial series – it was just an off-the-wall idea at first to see if there was much interest, but we've had quite a positive response back about it and are looking for other similar ideas.

With regards to dropping the use of **strdup()**, I can see where you are coming from, but I've tried to restrict use of the STL to where it's absolutely necessary to stop code bloat. I use the string class in my own work, but **strdup()** is much easier to teach, and also more familiar to people coming from a straight C background.

Building a parser class (a 'pure' parser) isn't something we had in mind, and we'll certainly consider it towards the end of the series – as long as support keeps up, that is!

Algorithm method

I wanted to drop a line to express my keen interest and ongoing support for the compiler-writing tutorial. I am a professional developer (who studied something completely non-computerish at university) who is considering applying to a graduate program in computer science. I have taken, over the past few years, to studying computer science (especially algorithm analysis) in my own time. This tutorial has reaffirmed my enthusiasm for the subject of computing. Thanks to Paul Hudson and the staff at Linux Format!
D. Matthew Burt, *Westerville, USA*

Thanks for taking the time to comment. I'm sure Paul will be using these responses to his tutorials to great effect in upcoming editorial meetings, and no doubt we'll have to pretend to pay attention to more of his strange ideas...

Where's MDKMove?

Usually your *LXF* DVD contains everything in the CD version and more, but for *LXF51,* the CDs had something the DVD was missing – MandrakeMove. You did promise to include MandrakeMove on the DVD for *LXF52*. I've bought the DVD-version of issue 52, but as far as I can tell, MandrakeMove is *not* included.

Could you please tell me if it's there – and if it is, where? If you forgot to include MandrakeMove on this DVD, would you then include it /next/ time (*LXF53*)... or will it just not be made available for those buying the DVD version?
Koppe, *via email*

By a strange quirk of fate, whereby the DVDs and CDs are mastered on different dates, we were able to include Move on the CDs for that issue. You are right that we did say we would try and put it on the DVD for the following issue, but inevitably we found more software that we thought had a right to be included. We've crammed the download version of Mandrake Move on the DVDs this issue along with six other Live CD distros, but for its full functionality, we suggest that you try and get hold of the retail version. It's good value as it includes a USB key drive to store your settings and files on.

Distro-allergic

***LXF* has entertained, informed, bemused, and generally been a tremendous help in migrating from RiscOS to Linux. The CDs have been excellent. I make great use of *Mozilla1.5* (installed from *LXF* coverdisc) and all my wife's business accounts are managed by *OpenOffice.org* spreadsheets. (*OOo* from coverdisc too) I could go on and on but you get the idea; the coverdiscs have saved me hours of dial-up download time.**

So it was a mild disappointment for me to find that *LXF53*'s three discs were almost entirely taken up by Mandrake 10. Having just gone through the stress of upgrading from Red Hat 7 to Fedora Core 1, it will be a while before I change again! I appreciate that you have to do your best to please the whole range of your readers, and some folk apparently like trying out each new distro as it comes along. It is not their fault that the mere thought of repartitioning the drive makes my palms sweat!

Please do not see this as a complaint: it is not. I just wanted to let you have a bit of feedback which, I hope, is a help. As I said, the mag is excellent and does a great job in catering for such a wide range of interests and expertise.
Ken Wilson, *Fishguard, Pembrokeshire*

We appreciate that not everyone is a Mandrake fan, though past issues where we featured it have often been the best-selling. There is a slight problem these days that, whereas once upon a time it was sufficient to have one CD for a full distro, many now require three or more. Might we make a cheeky suggestion and advise you to upgrade to a DVD drive? They're much cheaper now than they were even a year ago, and the DVD gives much better value, because of all the extra software we manage to cram on it, whether or not a full distro is included.

»

"Might we make a cheeky suggestion and advise you to upgrade to a DVD drive? They're much cheaper than a year ago, and the *LXF* DVD is better value..."

READER TIPS

SHARP C860

In response to Embedded Linux News from your April issue. I've had a Sharp C860 for about three weeks now and I can confirm they are a lovely bit of Linux goodness. They have quite literally the best screen available for a PDA (640x480 with exquisite brightness and clarity) as well as other superb hardware characteristics such as 128MB RAM and 64MB of flash memory.

Besides being very good at the normal PDA functions: Address book, Calendar, ToDo, Notes, email, all the other tools and games make this a fantastic laptop alternative: All the connectivity and development tools you expect of Linux matched with mobility and compactness! Surfing the web over a wireless connection is much more viable than another PDA or a BlackBerry device. Text editors and spreadsheet programs import and export to *OpenOffice.org* and Microsoft products just fine, and the breadth of freely available software means that there are plenty of tools and games with which to play.

www.shirtpocket.co.uk sells them in the UK, although due to the weak dollar I bought mine from www.thekompany.com. The C860 is actually a Japanese-only product (and well worth exporting). The soon to be released 6000 models will be available in the US and Europe shortly.

Did I mention I am writing this from my C860 at 39,000 feet?
Michael van Strien, *via email*

Thanks for your field report on the C860. It is rather annoying that Europe is last on the list for these goodies, but as you point out, they are available from importers. We hope you remembered to turn it off before landing!

NEW DISTRO

Want to try a new Linux distro? Here's one that comes highly recommended by an *LXF* reader...

I have recently discovered a fantastic new Linux distro: Arch. It's fast (boots in under 14 seconds), minimalistic, bleeding edge (currently running kernel 2.6.5 and XFree86 4.4), and has the best package management system on the street today.

It is quite simple, though not necessarily easy to use. It's speedy and responsive – handy when doing a quick **Alt+Tab** in difficult circumstances. And of course, it has a community. A brilliant community. You want more? OK – go get 'em: www.archlinux.org. Brilliant. I enjoyed the article on steganography a while back and I would love to see more like it.
Zack Adlington, *Winsford, Cheshire*

MANGLED-DRAKE9.2

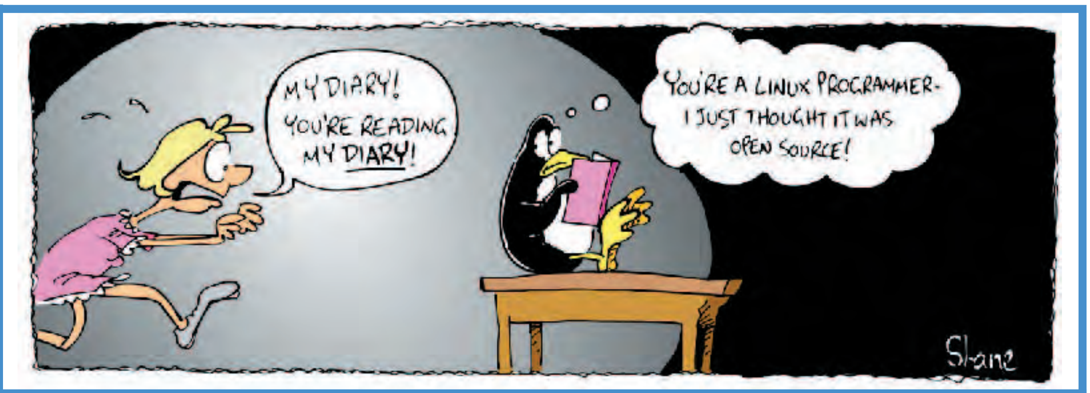
I've been paying close attention to problems people have had with Mandrake 9.2, as Mandrake has been my distro of choice for quite some time. I too performed a clean install of 9.2 from your cover DVD, and like some other readers, I too couldn't access the Internet. It's nothing to do with the serial modem (I've used an external modem for years and see no reason to change now). The problem lies with Mandrake 9.2 – *KPPP* is missing from the distro. I obtained a copy from an older Mandrake 9.0 installation DVD and everything now works fine. MDK had the same problem with its 9.1 download edition as well.

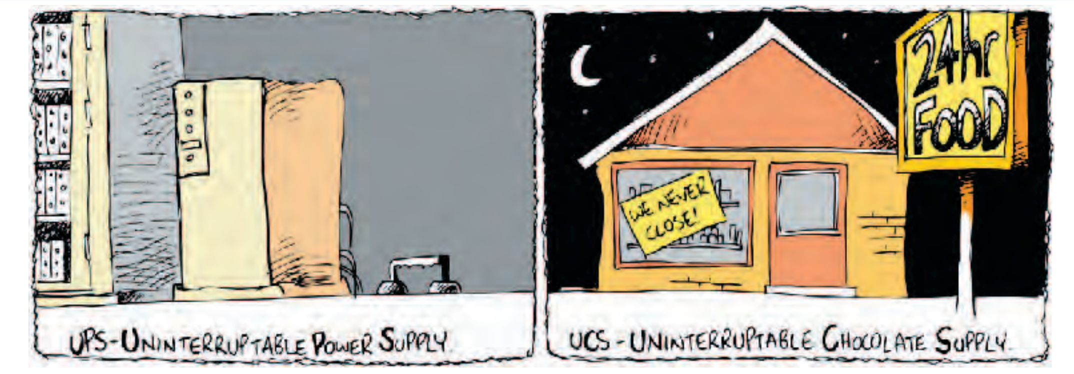
Please pass this information along to your readers – I wish them all the best with our Linux future. Now if only I could get my satellite Internet working... but that's an email for another day when I have finally torn out all of my hair!

Keep up the excellent work – the other UK mags don't even come close, and you've blown all of the US ones out of the water.
John Clayton, *via email*

You are right about the *KPPP* problem. Sadly, it's one thing that's not easy for us to test here, and we can't run a QA process on the whole of Mandrake! It isn't a terribly difficult problem to solve once you know what's gone wrong. We're sorry if it inconvenienced you or other readers.

Helpdex
shane_collinge@yahoo.com


MY DIARY! YOU'RE READING MY DIARY!
YOU'RE A LINUX PROGRAMMER- I JUST THOUGHT IT WAS OPEN SOURCE!
Shane

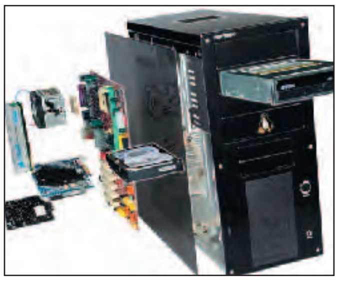

UPS-UNINTERRUPTABLE POWER SUPPLY
24hr FOOD
WE NEVER CLOSE!
UCS - UNINTERRUPTABLE CHOCOLATE SUPPLY

“Our ‘Ultimate Linux box’ feature last issue attracted quite a lot of mail from opinionated *Linux Format* readers!”

« Better CRT

Re: Ultimate Linux PC project in *LXF53*. I followed it with much interest, reasoning that by the time I have to upgrade my computer, these components will be a reasonable price, but still probably the best for the job, since you’ve already tested them together, and with Linux, for us.

My first inkling that all was not well came when you reached the point of choosing a monitor. I give you 3 quotes. *“Pick something poor and end up getting eye strain or headaches for years to come.” “LCDs are the way to go: slim, light, and much more attractive...”; “We chose... primarily because it’s light, has a first-class screen, and, on a side note, we wanted to see whether we could get the monitor pivoting to work with Linux”.*



How exactly to you equate the second two quotes with the first? Unless things have really changed drastically in the last couple of months since I had to replace my old (1990) monitor CRTs still walk all over LCDs for clarity. A first-class LCD only ranks as a second-rate CRT.

Then we come to input device: you went for a wire-free device. Now *everyone* knows that wire-free is slower than wired – you’ve even mentioned it yourselves previously. The difference may not be anything great under normal conditions, but this is supposed to be the ‘Ultimate’ box, and more to the point, it’s supposed to be a gaming machine, and in games, that delay can be the difference between getting the shot off and being on the receiving end.

Incidentally, my sister’s had one of those side-scrolling mouse wheels for a while now: and what’s more, it ain’t Microsoft!

Nevertheless, a most enjoyable feature, and I trust this machine will be offered as a prize in an upcoming competition which you’ll let me win?

Deke, *via email*

Colour a problem

I would make a minor point about your welcome article in *LXF53* on building a PC: a black case is not a wise choice for a DIY build. Anyone who builds a PC (and I built mine) will also be the type who will want to change or add to the configuration in the future, and one day re-use some components in the next upgrade. I am always doing such things. Even though you managed an all-black build here, further black components, as opposed to beige, may be impossible to find (try Travan tape drives) or the choice will be very limited, as you said yourselves about the optical drive.

You said the point of black was to “look the part”, but black mixed with beige will look junk! My advice is to leave black to the appliance PCs in the High Street shops.

Robert Phipps, *Bristol*

Wise points from both readers, though *any* shade of beige doesn’t scream “Ultimate!” to us – but the upgrade path may be easier. Or you could splash out on some black paint... LXF

SUBMISSION ADVICE

WHAT WE WANT:

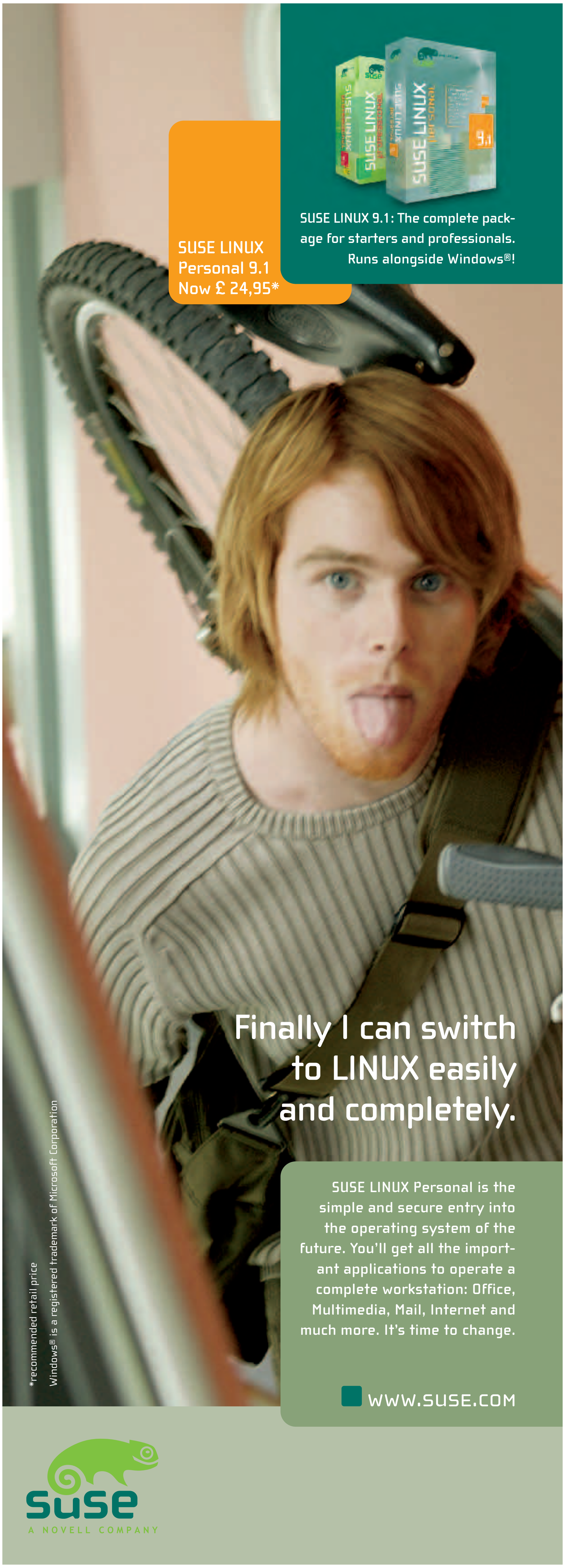
- Letters about the magazine or Linux in general
- Constructive criticism
- Your opinions
- Concise points about relevant subjects

WHAT WE DON’T WANT:

- Technical questions – direct those to our *Answers* pages!
- Random abuse
- Nonsense rants
- 200 pages of meandering diatribe

WRITE TO US AT:

***Linux Format*, Future Publishing, 30 Monmouth Street, Bath BA1 2BW** or email: **lxf.letters@futurenet.co.uk**



OPEN SOURCE ART APP

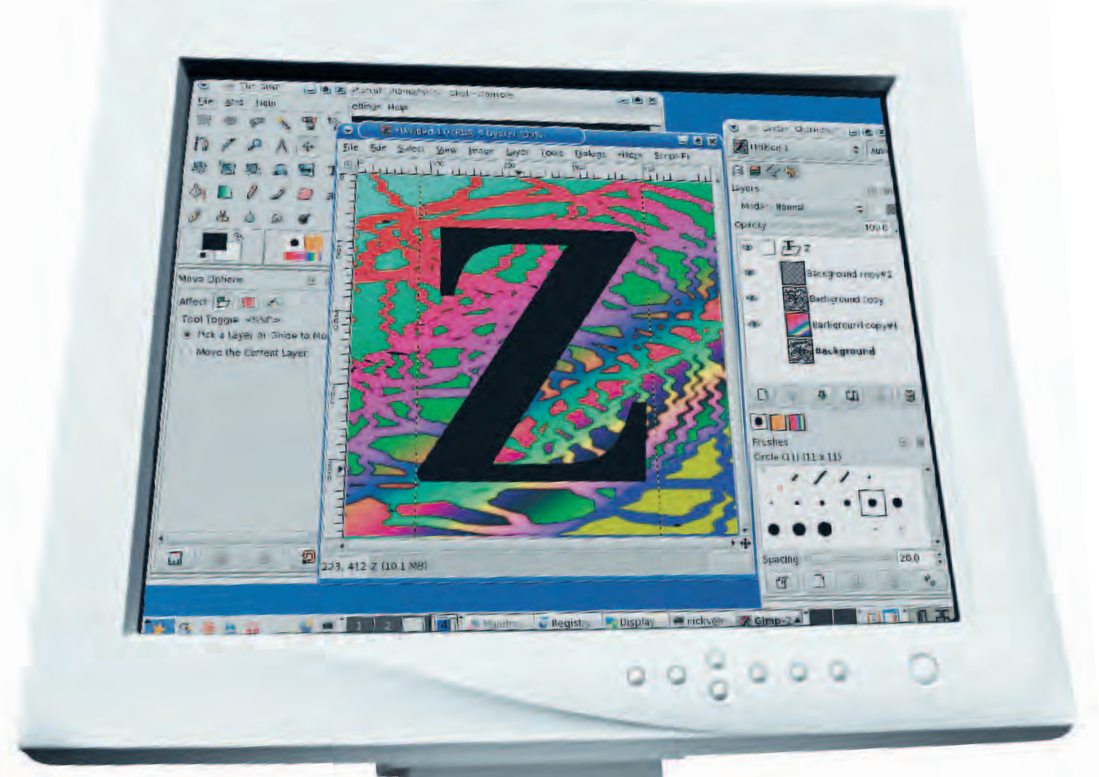
The GIMP 2.0

The artistically challenged Nick Veitch adjusts his beret and paints a picture of the best-known image-editing package for Linux without feeling the urge to cut off his ear.

BUYER INFO

Best-known Linux image editor. Comparable software is thin on the ground – you could try using *Adobe Photoshop* under *Wine*.

- **VERSION** 2.0
- **PRICE** Free
- **WEB** www.gimp.org/


A new interface and new layout, but not too much in the way of new features. This is a building block release.

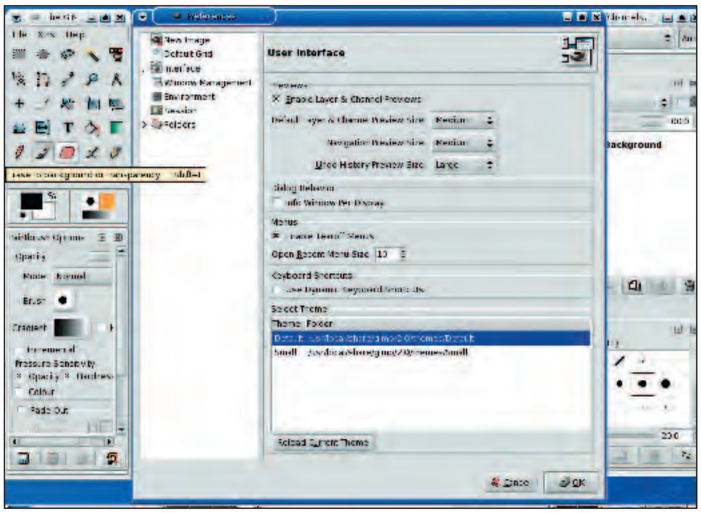
Should you ask any group of Linux users to name an Open Source desktop application, chances are the majority would opt for the *The GIMP*. After all, in its long history, it has spawned projects that many other apps have been built on. But long-standing project that it may be (see the *History* box, below) it has also remained largely unchanged for years. Much was hoped for the 2.0 release, as it has been heralded as an almost complete rewrite. But does it deliver anything to the user?

Cosmetic changes

The most immediate change is that *The GIMP* now makes use of *GTK+ 2*. Although many apps have been using this for some time, the intention of developing it in the first place was to further enable *The GIMP* with a more modern look and feel.

This is a bonus to the end-user. Anyone who wanted to use the software for anything more than a bit of picture retouching soon finds themselves with a multitude of window-spam cluttering up their desktop. With a new system of dockable palettes, there is no reason for more than two or three separate windows to be open at once (unless you want there to be, of course!)

As part of the clean-up operation, the menus have been further rationalised. There is now a menu bar at the top of the image window (where it always should have been), so no



Preferences are handled more sensibly and the entire GUI can be themed, which has prompted some distro vendors to experiment with branding.

A SHORT HISTORY OF...

The GIMP

The *GNU Image Manipulation Program* started its life in 1995. Original authors Spencer Kimball and Peter Mattis wanted to create a comprehensive image editor as part of their Computer Studies training while students at Berkeley. The original structure of the code provided only very basic tools, but instituted a plugin architecture that allowed third-parties to add their own functionality.

The original *GIMP*, version 0.54, used the well-known *Motif* GUI toolkit, which didn't exactly win it many fans, but one rather notable character gave it his blessing – Larry Ewing used this version to first create Tux the penguin!

Peter Mattis decided to write his own GUI toolkit to use with *The GIMP* and started the *GTK* and *GDK* projects. These were originally intended just to be used for *The GIMP*, but of course, took on a life of their own.

Kimball and Mattis released version 0.99 in early 1997, but it was to be their last release – the two students had graduated and got full-time jobs, and for a long while nobody even knew that they weren't working on *The GIMP* anymore. The project was saved by other developers creating a community around the project – there were no clear leaders from this point, but responsibilities for various parts of the project almost delegated themselves to appropriate people.

Development continued, with more features and functionality added over time, until the landmark 1.2 release in 2000. Until now, though there have been many bug fixes, there have been few dramatic changes to *The GIMP's* code.

NOW ON WIN32 AND MAC OS X

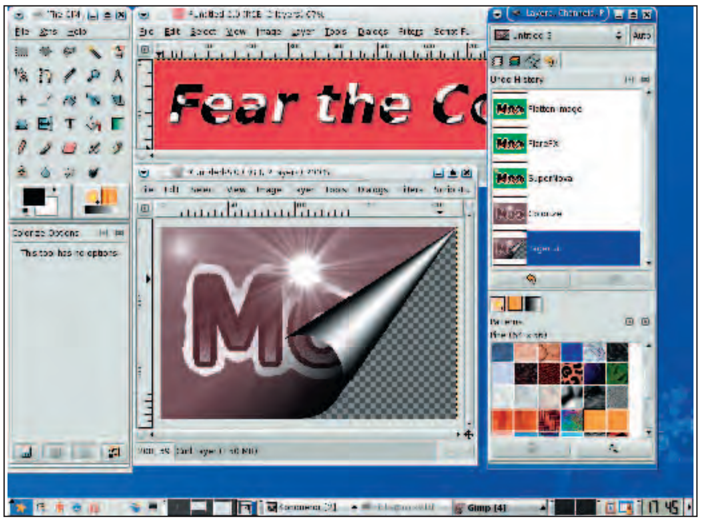
Free art software for proprietary systems

This is the first release of *The GIMP* that has enjoyed a simultaneous release on three major desktop platforms. The exact same software will run on Windows and Apple Mac OS X as well as your Linux desktop!

While we haven't tested the Windows version, it is reported to work fine. On the Mac, you need to install Apple's own X11 extensions before compiling the software or using one of the many prebuilt versions.

It has to be said that the Mac version is often painfully slow, though it wasn't apparent from our cursory exploration whether this was due to *The GIMP* itself (unlikely) or the implementation of *GTK* and/or X11 on OS X. Either way it can only improve, and now a great many more people are exposed to the glory of *The GIMP*.

It obviously doesn't look like a Mac-themed application, and it runs rather slowly, but it works!



The Undo buffer now includes thumbnail views for easy regression – handy when experimenting with filters.

more discovering that you've applied a filter to the wrong image. Further to this, many items have been shuffled around to put them in more sensible places – for example, colour mode transitions are now in the Layers menu, as they are layer-based and not image-based.

Although it is easy to dismiss a lot of the new changes as purely cosmetic, that doesn't reflect the fact that from a usability point of view, cosmetic changes can be very important. Take for instance, the Text tool. This has remained pretty much unchanged for around seven years. Select the Text tool in earlier versions of *The GIMP*, and then click somewhere on the image: up comes the requestor. Here you can fiddle around with the settings, hit OK and the text appears on the image. Then when you realise that it's completely the wrong size, or the font looks terrible and you go back and do it again. And again.

In *GIMP 2.0*, this trial-and-error is a thing of the past. For a start, the docked panel dealing with text now has a Font Preview option in the menu. While it only shows a capital and lower-case A, it does give enough of a hint. Plus, whatever text you type is now previewed in real-time on the actual image – changing the type size or font will let you see what it could look like immediately. So although this is mainly a cosmetic change, it actually makes the business of adding type to images far quicker and less frustrating.

One area of functionality that has seen a major overhaul is the system for handling paths. Paths were always handled in quite a primitive fashion in earlier versions of *The GIMP*, and had to enclose some sort of area. There is no longer any need for all paths to be closed, nor does it need to be entirely composed of the same type of nodes to distinguish between creation, editing and transforming or moving the path. Of more interest to web designers is the fact that SVG images can be imported with the paths intact.

Things that are sorely missed

Probably the biggest, most glaring omission from the software is the handling of different colour models. If it's RGB you are after, fine. If you want grayscale, that's OK too. Want any other sort of colour model? Forget it!

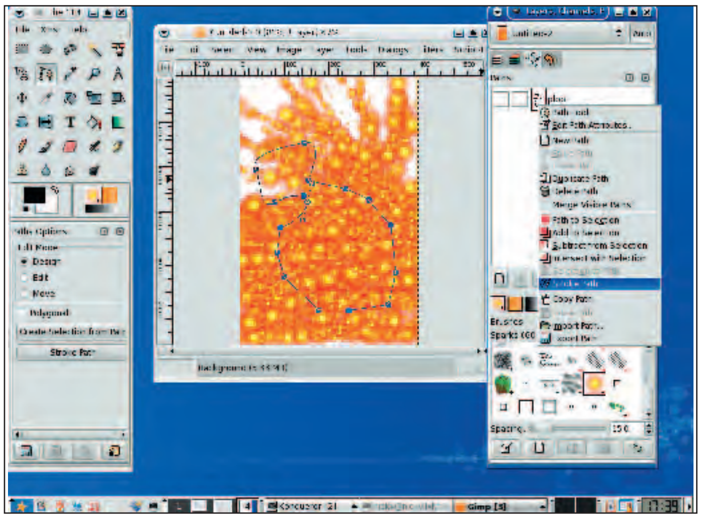
This is a problem for a lot of people who want to use *The GIMP* more productively. CMYK images are the *de facto* standard in the world of print for example, simply because those are the colours of the inks used to print. Yes, you can use RGB images and convert them with whatever you use to print, but you do face the prospect of not really having the vaguest idea of what the output will look like.

Yes, CMYK does make it into *The GIMP 2* as a colour model for selecting colours, but that's as far as it goes. It isn't just the odd print-artist who is inconvenienced by this. A lack of CMYK support also means that *GIMP 2* is still completely unprepared for images that are created in this format that users might try to load in.

Better code

With what is trumpeted as a major release, there may be some understandable disappointment from users to discover that – apart from different icons and an overhauled look-and-feel – there isn't anything substantially different here. True, some long-term issues have been addressed, and it's certainly a winner on the usability stakes, but in terms of features and functionality, its still way behind Corel's *Photopaint*, which briefly appeared on the Linux scene four years ago.

What we are promised is that, with a complete restructuring of the underlying code, the whole application is now more easily extensible. So, this should be the beginning of a golden age of *GIMP* coding, with the features flowing thick and fast from now on. Time will tell. LXF



Enhanced path support makes handling SVG images easier, and opens up new uses for this often overlooked feature.

NEW FEATURES

- New GUI using *GTK+ 2* toolkit
- Interactive text editing
- Enhanced path support
- Menu bar on image window
- Full-screen mode
- Toggle viewable effects
- Tab and dock interface
- Theme support
- Image templates
- New layer modes
- Crop using active selection
- New Color display filters for Gamma correction *etc.*
- EXIF data preserved in JPEGs
- SVG support
- Support for MNG animations
- New channel mixer filter

LINUX FORMAT VERDICT

FEATURES	7/10
PERFORMANCE	8/10
EASE OF USE	8/10
VALUE FOR MONEY	10/10

Obviously still the best Linux art application, but we could do with more up-to-date features next time – and useable CMYK handling for print use.

RATING 8/10

●●●●●●●●○○

LINUX DISTRIBUTION

SUSE Linux 9.1 Professional

***LXF*'s love-affair with SUSE has smouldered for years, but will this release keep the flame burning? Paul Hudson surreptitiously books a hotel room...**

BUYER INFO

The leading boxed home distro gets a refresh. Also consider Mandrake 10 or Fedora Core 2.

- **SUPPLIER** SUSE
- **PRICE** £64.99
- **WEB** www.suse.co.uk/

Linux itself is but a very small part of what we use on our computers on a day-to-day basis. Yes, as the kernel of it all, it is absolutely crucial to the operation of pretty much everything, but it still only forms one little checkbox when comparing distro against distro. Very often, the reasoning is reduced to: *"Well, this version has KDE 3.2, whereas* this *version has KDE 3.2.1, therefore the latter is better"*, which is rarely accurate. Although SUSE plays the numbers game as much as any other distro, it's always SUSE's added value that makes users happy.

The big numbers in this release are kernel 2.6, GNOME 2.4.2 (already outdated by the new GNOME 2.6 that you'll find in Fedora Core 2), KDE 3.2.1, and *Samba 3*. SUSE has quite a long QA process, so it's no surprise that GNOME 2.6 isn't featured.

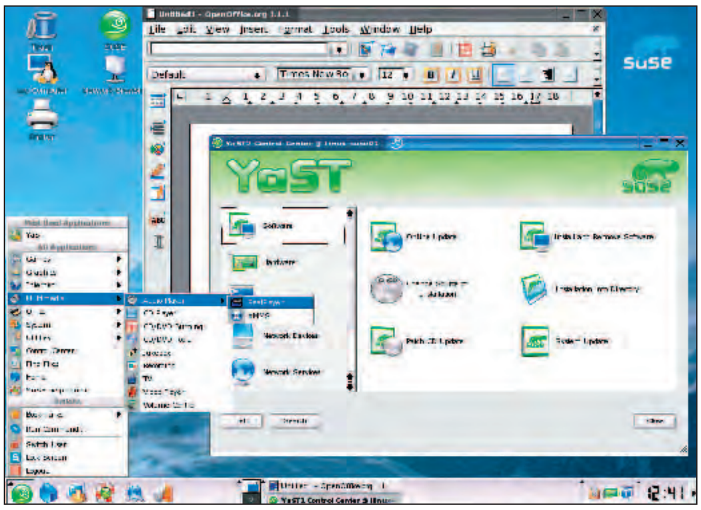
Beyond these base packages, you'll also find demo versions of *TextMaker* (reviewed in *LXF43*) and *PlanMaker* (reviewed in *LXF53*), the usual favourites such as *OpenOffice.org 1.1.1* and *Evolution*, as well as some the new apps bundled with KDE such as *Kopete*, *JuK*, and *KDevelop 3.0*, the latter of which only comes with the Professional release. However, these are heavily tweaked to get more uniformity – *OpenOffice.org* and all the GNOME apps have been given KDE-style skins. In *OOo*'s case, the application is actually linked with *Qt*, so that changing your *Qt* theme also changes the way *OOo* looks. In the GNOME apps, the SUSE developers have just created a Keramik-style *GTK* skin that makes it look like the apps are using *Qt*. This works in reverse too – if you use GNOME, you'll find the KDE apps all use a *GTK*-esque skin.

Value for money

As always, SUSE has provided a solid line-up of applications, and buying the boxed copy comes with the usual SUSE value: for your £64.99 you get five CDs, two double-sided DVDs, and you also get full support for both 32-bit x86 chips and 64-bit AMD64/Intel 64 chips. Previously the AMD64 release of SUSE Professional was a separate purchase, and we're really glad to see SUSE has chosen to combine it into one product. If you already own SUSE 9.0, the upgrade price is usually about 33 per cent lower, which means you get an awful lot for your 40 clams.

THE WINNING NUMBERS

Kernel 2.6
KDE 3.2.1
GNOME 2.4.2
KDevelop 3
Samba 3
OpenOffice.org 1.1.1



***OpenOffice.org* sports a shiny new KDE theme, and even the formerly unattractive *YaST* has been overhauled too.**

If you opt for the personal edition – now priced at just £24.99 – you get two CDs, of which one is a Live CD. Yes, that does mean the entire OS and application suite is contained on the other CD, which is quite cramped. To facilitate this, a number of cutbacks have been made, such as removing GNOME and leaving KDE. This is justified by the fact that most users coming from Windows aren't going to understand why they have a choice between window managers (or even what a window manager is). However, with the Pro version almost three times the price of the Personal version, it's quite a jump for users who want to upgrade.

If you're a fan of the SUSE user manuals, you'll be pleased to find that they're comprehensively revised to take into account the updated software, and are as informative as ever. More often than not, it's the quality of the manuals that people cite as the best thing about SUSE Linux, and it's also why we usually recommend SUSE for beginners.

Is that it?

Without GNOME 2.6, even the new software will lack some pizzazz for some users, which means the only real star functionality is the inclusion of AMD64 and Intel 64 technology in the box. This is the first distro to have made this step, and it *is* excellent value for money, particularly if you have machines with different architectures. The rebranding of applications is nice, though only the first step of several – there's a lot more work to be done before the *Qt* integration is done properly, as opposed to just a skin.

If you're already using SUSE 9.0, there are several reasons to upgrade. The new kernel works just fine, it includes a customised release of XFree86 4.4RC2 (the release before the licence change) that gives all the benefits of the newer release and is rock solid, and KDE 3.2 is simply a joy to use. *YaST*, for its part, has gone from an application we seriously despise to something very usable – even command-line package installation has been redone to make it almost as powerful as Debian's *apt-get*.

So many people feared that Novell purchasing SUSE would signal the end of SUSE's support for KDE, but nothing could be further from the truth – this is the strongest release of SUSE yet, and is unparalleled in its ease-of-use for beginners and veterans alike. LXF

LINUX FORMAT VERDICT

FEATURES	**10/10**
PERFORMANCE	**9/10**
EASE OF USE	**9/10**
VALUE FOR MONEY	**10/10**

Too many incremental improvements over 9.0 to list, but the new-look *YaST* and full 64-bit support out of the box are definitely the most prominent highlights.

RATING 9/10

●●●●●●●●●○

COMMUNITY DISTRO

Fedora Core 2

Have Red Hat and the Fedora Project cracked it? Andy Hudson thinks that they're very close...

BUYER INFO

Community-led distro that forms the core of Red Hat's Enterprise plans. Also consider Mandrake 10.0, SUSE 9.1 Personal Edition, or Debian.

- **SUPPLIER** Fedora
- **PRICE** Free

Same, but different – The Fedora Project's distro now has Optical and Floppy Icons on the desktop under KDE.

OK, I admit it: I've got a bit of 'a thing' about Red Hat in general and Fedora Core in particular, having used Fedora Core 1 (available on *LXF49*'s coverdiscs) since its first test release, and various versions of Red Hat Linux beforehand. Historically, Red Hat has been perceived by some in the Linux community as being 'the Microsoft of the Linux world', but after speaking to a Red Hat staff member recently, it would seem that the company realised that it was in danger of losing its community roots and decided that the best way to redress this was by merging the Red Hat Linux project with the Fedora Project.

Several months after the merger, we now have Fedora Core 2 (FC2). Sticking as closely as possible to the release schedules laid down by the project team, the Fedora Project has managed to get two huge crowd-pullers into FC2: *SELinux* and the 2.6 Kernel. *SELinux* is designed to protect your system at the kernel level to prevent any malicious or overly flawed software from destroying any files or devices. The 2.6 kernel is also a major step forward, and was covered in some depth back in *LXF46*.

Getting it on

The install process is much the same, giving you the option to upgrade existing FC1 installations or install fresh. Out of interest, I did try upgrading my existing FC1 set-up, but had problems when it came to booting the system. This was corrected using a switch in *GRUB*, but I thought it would be appropriate to start from scratch and do a full install. This took about 40 minutes, and included pretty much everything in the package list. A very nice addition (also presented in Red Hat Enterprise Linux 3) is that it auto-creates a *KickStart* file in the root directory, which could prove to be a real boon for system administrators having to install this on multiple computers sharing the same specification.

Boot-up is still the same, and you are greeted with the familiar graphical login screen. A nice security feature is that when you enter your username and hit **Enter** to prompt for the password, you are given the last time you were last seen by the system. Logging into either GNOME or KDE produces no real surprises, as FC2 still has the Bluecurve theme throughout so that things look similar in either desktop environment. Having said that, there is one major difference between the two desktops – in GNOME there is a computer icon reminiscent of Java Desktop System, while in KDE there are separate icons for the floppy drive and optical drives. This is a nice touch, giving new users a bit more familiarity switching from a Microsoft desktop to Fedora Core. By default, FC2 logs into GNOME 2.6 but this can be easily switched to KDE, should you prefer.

The KDE 3.2 and GNOME 2.6 enhancements are all there, including the bouncing 'busy' icon in KDE that's reminiscent of the Dock in Mac OS X. Personally I prefer KDE to GNOME, but to the average user who doesn't necessarily want to fiddle about under the bonnet, there really is no major difference between the two. The usual applications are, of course, all here: *Ximian Evolution, OpenOffice.org 1.1.0, Mozilla 1.6, GAIM 0.75, The GIMP 2.0.* The menu structures are also largely the same for both desktops. However, each category contains all the programs available to that category rather than a select few, with the rest tucked away in a sub-category. This makes for some rather busy menus, but is just a niggle rather than a major problem. What's obvious is that FC2 wants to give you as much choice as possible – there are many programs for productivity, Internet and email access, not to mention a very long list of games!

As this is a community release, there is no packaging available for it, so there are no nice box shots or typed manuals, which could be a small let-down for some new users who are looking to make the switch. However, there have been hints that Red Hat will be returning to the mainstream desktop market in the future, so this will almost definitely be addressed should this happen.

Technology preview

The Fedora Project is a great way of getting your hands on future Red Hat Enterprise Linux (RHEL) technology long before it is available in an official RHEL version. *SELinux* is one of the features that is due to go into RHEL 4, which had only just gone into Alpha in February 2004, and is due for release at some time during 2005. If the Fedora Project manages to stick to its schedules of releasing 2–3 versions per year, then Fedora acolytes should just be on Fedora Core 5 by the time the next RHEL is available.

Kernel 2.6 is just one reason to upgrade – the inclusion of KDE 3.2 and GNOME 2.6 are both powerful incentives in their own right, making the Fedora Project a great distro for both new and advanced users alike. LXF

LINUX FORMAT VERDICT

FEATURES	8/10
PERFORMANCE	9/10
EASE OF USE	8/10
DOCUMENTATION	7/10

A great-looking, easy-to-use, and powerful distro with strong backing from one of the largest Linux companies in the industry.

RATING 8/10

●●●●●●●●○○

ALL-IN-ONE SOLUTION

Ayrsoft icon Server Appliance

If you want an appliance to do everything *without* requiring you to know everything, Paul Hudson shows us a breath of fresh Ayr...

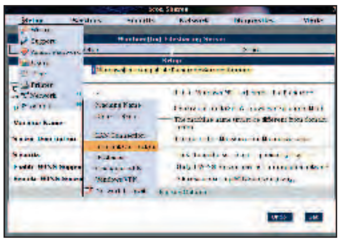
BUYER INFO

All-in-one server with a custom front-end to make it usable by everybody, even non-techies.

- **SUPPLIER** Ayrsoft
- **PRICE** Starting from £999 inc VAT
- **WEB** www.ayrsoft.com/
- **PHONE** 01294 318722

Linux as a server and Linux as an appliance are both trusted deployment situations, and it's fairly common for the two to meet. The big problem with Linux – at least according to those who don't use it day-to-day – is that it's hard to use, and to a point, that's correct. As people want the power and cost advantages of Linux without any of the administration learning-curve, Linux server appliances – products that provide plug-'n'-play functionality for the server world – are becoming increasingly popular.

Ayrsoft's icon Server ships as standard with a compact Shuttle unit that contains an Athlon XP 1800+, 256MB RAM, a 40GB hard disk, and a CD-RW. That plus a licence for the icon software itself comes in at just £999 for under 10 users, which is a real bargain. This rises as you license more users – for fewer than 50 users it's £2799 including the server, and for fewer than 150 it's £5099. Compared to Navaho's TeamCAT device, which is largely similar to this and starts at £2385, the icon is clearly much more competitively priced. Of course, Navaho is renowned for its technical support, whereas you pay extra for support with Ayrsoft.



GUI-based admin is all very well, but where's the command-line?!

Three levels of functionality are available in icon Server, with the most basic level ('Starter') coming with a firewall, CRM tools, and email, and the most advanced level ('icon Server+') coming with intrusion detection, proxying, file and printer sharing, VPN, DHCP, and other enterprise-level servers. Unless you're after a particularly basic system, the chances are you'll want the icon Server+ package, as it gives you the most value for money. To handle the extra load, the Server+ solution comes with a 2.8GHz Pentium 4 with mirrored 80GB hard disks.

Install and config

Although our server came pre-installed with the icon software, we re-installed from scratch in order to see how complex the installation process was. The software itself is a full Linux distro that's essentially a heavily modified Mandrake. Everything is rebranded, however, and much of the easy installer is skipped so that you have the entire system installed with no user input.

Once the system is up, you login the first time as root, which brings up a configuration screen where you can use menus to set up your network information. Although the menus are nice, what this device sorely misses is an option 'Go to console' – we were unable to get to a console no matter what, which is irritating. We can see why Ayrsoft would rather people didn't get into the core of the machine, but it'd be nice as a failsafe! A quick call to Ayrsoft's tech support , and it turns out that there is a way to get the console by way of a back door, which means most users can't get there; but if you talk to tech support, they can guide you through – the best of both worlds, in our opinion.

With the initial configuration out of the way, you can log in remotely through your web browser. This is where the majority of configuration is done – any administrator can control the server from any location and from any device – or so you'd think. Sadly, this isn't the case. Yes, the device can be accessed wherever you please; but no, don't expect it to work anywhere – there's a lot of JavaScript in there to handle the fancy interface, and we couldn't get it to work with *Konqueror*. Fortunately, it *does* work fine with *Mozilla*, but precious few PDAs and mobile phones (the quintessential emergency administration devices) would come with such a heavyweight browser!

Once you get your browser working, actually using the system is easy, if a little problematic. We had various problems getting the proxy server to work, for example – although it's all based around a simple web GUI, very often our actions would be ignored. We called the tech support line, and were through to a technician within ten seconds, who promised us a fix before the end of the day.

Sadly, when we came into the office the next day and loaded up the admin console to see if the update had come through (it checks, downloads, and installs automatically), the screen was filled with messages such as "This software has expired", "Your Product ID is not registered", and other terminal-looking messages. This time, when we called them at 9:45am, no one from technical support was available. Fortunately this just seemed to be down to chance, as calling back a few minutes later we got straight through. When we finally did get through to technical support, we were assured that this kind of problem doesn't happen to customers – we'll let you decide for yourself on that one.

When it works...

If you can get over the initial niggling problems, you'll find this does do all that was promised, and does so quickly and easily. At this time, though, there are enough bugs and irritations in the system that make it less than ideal, and unless you like being a guinea pig, we'd suggest you stay clear for the next couple of months. Once Ayrsoft sorts out these problems, we'd be happy to take another look at this box and give it our seal of approval, but right now we're only 95 per cent convinced. LXF

LINUX FORMAT VERDICT

FEATURES	8/10
PERFORMANCE	7/10
EASE OF USE	8/10
VALUE FOR MONEY	10/10

It's cheap, but sadly in the quality sense almost as much as the pricing sense. We'd wait for the next big release.

RATING **8/10**

●●●●●●●●○○

1U OPTERON SERVER

Systemax Mission 3607

1-way Opterons are almost extinct thanks to the Athlon 64 FX-53, but Paul Hudson shows that there's life in them yet...

BUYER INFO

Single CPU Opteron 248 with 1GB of RAM and a low, low, price. We reviewed the dual-CPU version (Mission 3602) in *LXF52*, but also consider the Armari Opteron that was reviewed back in *LXF45*.

- **SUPPLIER** Systemax
- **PRICE** £1399 + VAT
- **WEB** www.systemaxpc.co.uk/

Although there's usually a big warranty difference between the two, the Athlon 64 FX CPU and Opterons are virtually identical otherwise. The big – and only *real* – advantage in Opteron's favour is that it supports multi-processing, whereas Athlon 64 chips are always limited to working by themselves. Why, then, has Systemax produced a 1-way Opteron – wouldn't it be more economical to base the system on the less-expensive Athlon 64?

This machine is very similar to the machine we reviewed in *LXF52*, with the exception that it has just one CPU. The motherboard, however, is the same in both machines, and this is this server's unique selling point: although only one CPU is fitted for the price, there is space for two, which makes upgrading a snap if you find you need more performance at a later date.

As we're now two months on from the previous review, the price is even lower than before – the dual-CPU version of this server was £2299+VAT when we reviewed it, whereas this one is £1399+VAT. To get the equivalent in performance, you'd need to purchase another Opteron 248 chip, but Misco (**www.misco.co.uk**) has these on sale for just £555+VAT – add the two together and you get £1954+VAT in total, which is a full £345+VAT cheaper than the 3607.

No, there's still no PS/2 port, but other than that we like the way this machine is built. It's fairly quiet, and yet still manages to be cool to the touch after a full day of benchmarks.

Moore's law still holds sway, if the price of this hardware is anything to go by.

System spec

Having 1GB of RAM is about the minimum we're prepared to accept in servers aimed at HPC, although we've a feeling this sort of machine isn't destined for the task unless the extra CPU is purchased. This is further re-enforced with the twin 160GB 7200RPM hard disk drives bundled along, which make this system more of a contender for general data serving. The fact that this comes with dual Gigabit Ethernet suggest a web-based role – we like how the Opterons perform in this area, and the capacious storage and strong network support combine to make this a pretty ideal *Apache* box.

The benchmark scores are almost exactly what we would expect from this hardware, particularly after having benchmarked the 3607. Again, the high-end RAM score makes this a fantastic box for web serving – keep in mind that content-caching solutions will store as much as they can in RAM, so the hard disk score can almost be ignored completely.

At three times our yardstick machine (a 1.8GHz Pentium 4), this has CPU power to spare even with only one CPU. If you're on a particularly tight budget, you could downgrade the chips to the less expensive 244s – at about half the price, they run at 1.8GHz compared to the 2.2GHz of the 248s, so you don't lose much of the performance. If you're comparing this machine against the 3607 in *LXF52*, keep in mind that the 3607 scored 6.07 in the multi-processing test, which is what gave it the higher overall score of 2.93.

Still no Linux

Systemax has still yet to start bundling Linux with its servers, which is beginning to vex us a little. Even a copy of a SUSE 9.0 Professional for AMD64 on DVD would be enough to satisfy this requirement, and that only costs £25 for OEMs – dropping to almost a third of even that low price depending on the number of copies shipped. Even if the company buying it decides they want to replace it with its own distro, we think the addition of such a powerful and flexible OS as SUSE 9.0 would put this server very firmly at the top of its class.

As it has always been a company to go for 'firsts', it's no surprise that Systemax is marketing this server aggressively. The few 1-way Opteron servers we've seen in the past have been there just to pad out a product range, but this offering – particularly when you consider how easy it is to double the CPU performance by fitting another chip – really stands out from the pack. This is a format we'd like to see more of, as it allows companies to scale up as and when they want to.

As for price, what you can you say but "Wow"? Systemax has built up a strong reputation for always keeping its prices low, but this is a pretty unprecedented effort, and is very welcome. If you were looking for a good motive to switch to 64-bit, even if you don't want a 64-bit OS just now, this is the best reason we've seen yet. LXF

BENCHMARKS

CPU	3.1
SINGLE	3.0
RAM	2.17
HD	0.49

OVERALL **2.93**

LINUX FORMAT VERDICT

FEATURES	8/10
PERFORMANCE	10/10
DOCUMENTATION	7/10
VALUE FOR MONEY	10/10

A lean and mean box that provides maximum return on investment and the flexibility to scale up when needed.

RATING **8/10**

●●●●●●●●○○

1U PENTIUM4 SERVER

Gigabyte SR114E

Paul Hudson may have moved to the other side of the *Linux Format* office, but there's still lots of room for servers under his desk...

BUYER INFO

2004 update of Gigabyte SR113, a bargain 1U server. See review text for suggested alternatives.

- **SUPPLIER** Upgrade Options (0800 008 006)
- **PRICE** £820 (£698 ex VAT)
- **WEB** www.upgrade.co.uk/

Back in *LXF40,* we reviewed a remarkably cheap 1U server from Gigabyte that promised a lot – and it delivered too. Since then we've had lots of positive feedback from readers who bought the machine as a result of the review, which isn't surprising – the SR113 we reviewed was a real workhorse and available at a cracking price. Now, just over a year later, Gigabyte has provided us with the model replacement for SR113, aptly named the SR114, which follows much the same mantra but updates the components to the latest specs.

Surprisingly little has changed between the models: the SR113 has a 2.4GHz GHz CPU, 1GB of RAM, and 3x30GB hard drives chained together through a Promise RAID controller. The SR114 comes with a 2.8GHz CPU, 1GB of RAM, and 2x40GB hard disks. The new CPU is about the same price as the 2.4GHz CPU was when we reviewed the SR113, so essentially hasn't changed. The hard drives are actually *worse* in this later model – yes, they are 40GB a pop, but you only get two of them – giving a total capacity of 80GB as opposed to 90GB. Furthermore, you need at least three drives to do RAID striping with parity.

So, why buy this as opposed to the SR113? Well, the key difference is price: this machine is just £820 including VAT, whereas the SR113 we reviewed back in *LXF40* was £872 *excluding* VAT. If you wanted to lay out an extra hundred pounds or thereabouts, you could probably upgrade this thing so that it has larger drives, making it a clear winner in comparison to the previous model.

Pros and cons

We get *lots* of servers for review here at *Linux Format*, of which we tend to print quite a few. Of the many that fly by, there are, as you'll know, three 'lacks' that irk us particularly: lack of PS2 ports, lack of installed OS, and lack of documentation. The first 'lack' is that manufacturers seem to be moving away from the venerable PS2 format. However, this so-called budget server had two PS2 ports as well as four USB ports – connectivity paradise, in our opinion.

The second lack is OS: as per usual, this came with no OS. This is irritating, because it's extra hassle for us having to install Linux, and and also because the SR113 came with Red Hat 7.3 pre-installed. Although Red Hat Enterprise Linux 3 costs more to license than this machine itself, there are many other free distros, and we fail to see why Gigabyte and others can't provide Mandrake or Debian.

Documentation? Surprisingly, this is the best documentation we've seen for a while. Two booklets (one for install guide, one for user guide) come bundled, along with a documentation CD. This inclusion and also the inclusion of PS2 ports would make us question whether this is a budget server at all – we rarely see this sort of thing in servers ten times the price!

Go faster stripes

Thanks to the Opteron moving the performance goalposts for servers across the board, we're not sure quite where the Xeon chip has a market. This machine, in order to save costs, comes with a Pentium 4 chip, which is one of the few parts of this machine that marks it out as 'budget' from a distance. However, our yardstick is also a Pentium 4, so we expect linear performance improvements for this machine.

As you can see in the benchmark results box, both the multi-processing and uni-processing tests scored just over 1.5, which clock-for-clock would make this machine a 2.75GHz chip. Of course, there is a small margin for error, and also this chip has a 533Mhz front-side bus to speed it along, so this is exactly what we'd expect. The RAM score is definitely the headline figure: 1.77 is probably half as a result of the CPU score and half as a result of the FSB. Either way, it's a great score (for a P4) and shows this machine is definitely best put into an environment where it can cache whatever it is serving up.

A hard drive score of 1.14, while being nothing worth shouting about, is slightly above our yardstick and certainly isn't *bad*, by any means. However, not having that third drive does mean you can't use striping with parity for an extra speed boost, and few people would want to stripe *without* parity on a server box.

A strong successor

We liked the SR113 a lot, and it isn't easily knocked off its pedestal of "ultimate budget server". This machine comes close, and if it weren't for the lack of a third drive we'd already be seeing the back of the SR113. On price, this thing comes even cheaper than its predecessor, which, if you care about price and little else, should mean you're all set to whip out the purchase orders.

But, without the option for parity striping, and with the hard drives being too small to support straight mirroring, we don't consider this machine to have quite the same reliability proposition as the SR113. This is a real shame, as everything else is very good indeed, and for such a low price that it makes us wonder where the apparent negative inflation came from...? LXF

Look – PS/2 ports! Some other manufacturers seem to be trying to kill off ths connection method...

BENCHMARKS

CPU	1.54
SINGLE	1.51
RAM	1.77
HD	1.14

OVERALL **1.48**

LINUX FORMAT VERDICT

FEATURES	8/10
PERFORMANCE	7/10
DOCUMENTATION	9/10
VALUE FOR MONEY	10/10

Faster and cheaper than its predecessor, has good docs and great connectivity – but do try to get bigger hard disks!

RATING 8/10

●●●●●●●●○○

PROGRAMMING LANGUAGE

Java 2 Standard Edition 1.5

Fighting off competition from C#, Java comes out with a roar – or so Paul Hudson thinks...

BUYER INFO

Programming language and virtual machine all-in-one. Also consider *Kylix* or plain old C++.

- **DEVELOPER** Sun
- **PRICE** Free
- **WEB** http://java.sun.com/

Now we give it some thought, it's actually been quite some time since Java was updated, despite an incredibly aggressive onslaught from Microsoft with C#. Since we last reviewed it, *Mono* has come on leaps and bounds in its C# support, and there have been extensive calls for Sun to open-source it once and for all. These requests were rejected about a month ago, and this is the biggest Java release since then. As it has been such a long time in development, it's no surprise that there are some big differences between 1.5 – codenamed *Tiger* – and 1.4.2. The headline changes are Opteron support, autoboxing and unboxing, generics, and shorthand for loop, enumerators, and two new look and feels: Synth and Ocean. Despite all that, the question remains: is *Tiger* strong enough to maul C#?

AMD64 anyone?

With Sun being the leading tier 1 vendor on the move to Opteron, and having already ported its *Java Enterprise System* to the AMD64 platform, it comes as no surprise that Opteron support was one of the first things implemented in *Tiger*. Although we only had chance to test it with SLES 8 (the first platform supported in *Tiger*), we found it to be very stable. Performance-wise, *Tiger* is supposed to be faster at starting up and also should consume less memory, but we didn't find *that* much of a difference in the AMD64 build – 5 per cent at most. This is likely to change soon, though, as more people test it out and Sun is able to get more feedback.

Language tweaks

Four of the headline changes are aimed at making the language faster and easier to use. Although you get a little sick reading the word 'boilerplate' when Sun talks about how much unimportant code programmers write currently that they won't have to write in Tiger, it is true – to an extent. Generics work, like the STL in C++, except they are slightly less of a hack – rather than having to typecast all the time when adding to an array, you can simply tell Java that the array is an array of integers, and you won't have to typecast again.

Another big timesaver is the automatic boxing and unboxing, which is a very common-sense change. Experienced Java programmers will tell you how annoying it is to have to hand-convert ints to Integers when adding to and from collections – an irritation known as 'boxing' – and hand-convert Integers back to ints when reading from a collection – the corresponding irritation known as 'unboxing'. Luckily, Sun has had mercy on all of us and made this automatic, *viz*:

```
int wombat = 1;
Vector vec = new Vector();
vec.addElement(wombat);
```

Previously adding to **vec** would have required creating a new Integer with the value of **wombat** – quite a pointless step, and one no one will be sorry to see disappear with *Tiger*.

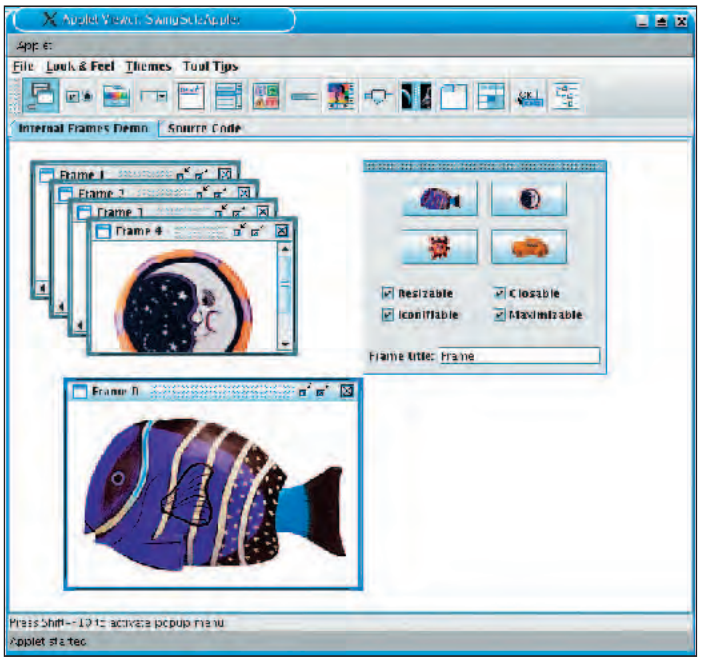
The new for loop is designed for iterating through collections of data in the same way as foreach() works in some other languages. This is designed specifically for looping through arrays, so the bounds are automatically set by the array itself. For example, the old way to do things was this:

```
for (Iterator i =
mycollection.iterator(); i.hasNext(); ) {
 Integer i = (Integer) i.next();
}
```

However, as of *Tiger*, this has slimmed down considerably to become:

```
for (Integer i : mycollection) {
//
}
```

It's substantially easier to read and shorter to type, so again it's a nice timesaver for programmers.



The new Ocean look (pictured) is smarter, slicker, and well overdue. Synth, for its part, is a special customisable look-and-feel that allows you to change the look-and-feel of your application just by changing pictures – the code remains the same.

The last new headline language change is the inclusion of enumerators: a feature many have sorely missed from C. Enumerators allow you to specify sets constants that form a type, such as:

```
public enum Seasons { spring,
summer, autumn, winter }
```

This type can then be used as part of a class, which means that setting it is as simple as

```
myobj->season = summer;
```

There isn't really much of an equivalent to this in earlier versions, which had led many developers to create their own miniature hacks to achieve the same end.

New look-and-feel

At long last, the old Metal look and theme is updated with an attractive new design. In order to retain maximum compatibility, the sizes and shapes of objects haven't changed – only the looks have. Though this isn't ideal, the reason behind it makes perfect sense, and at last Java developers don't have to feel quite so embarrassed about the cross-platform look-and-feel.

All the other changes to the language behind the scenes mean the new look-and-feel is much more than just a cosmetic change – this is a whole new breed of Java that should give Java developers new confidence that C# is going to remain #2 for some time yet. For users, the new GUI design is so much more attractive than the abortive Metal design that it's hardly worth considering, and the speed boosts (particularly noticeable in 32-bit mode right now) will also keep people smiling. LXF

LINUX FORMAT VERDICT

FEATURES	10/10
PERFORMANCE	9/10
EASE OF USE	9/10
DOCUMENTATION	8/10

Java developers need no longer hide in shame when talking about GUIs, and the new programming language semantics make Java feel like a whole new language.

RATING 9/10

●●●●●●●●●○

VIRTUALISATION SOFTWARE

VMware GSX Server 3

LINUX FORMAT TOP STUFF AWARD

Why buy another server when your existing hardware is barely used? Paul Hudson has the solution.

BUYER INFO

High-performance server virtualisation system that lets you run multiple OSes on one system.

- **SUPPLIER** VMware
- **PRICE** Starts from $2500 for a 2-CPU server
- **WEB** www.vmware.com/

Balancing your infrastructure to make maximum use of what you have is no easy task, which is why more and more companies are turning to *VMware* to supply virtualisation software. Having a virtual infrastructure means that you can install all your systems on one server, which itself runs a variety of operating systems and configurations.

This is a complicated task – much more so than just running a virtual machine locally through a product such as *VMware Workstation* – but it's precisely what VMware's *GSX Server 3* claims to do.

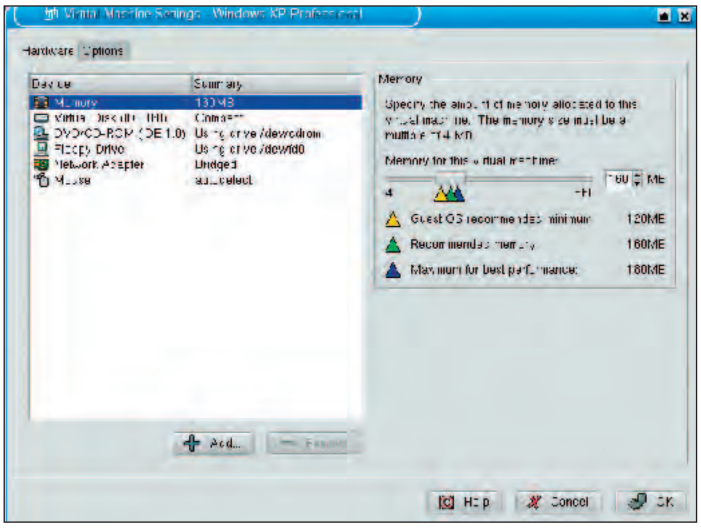
The big upgrade

Compared to *GSX Server 2*, this new release adds three big features: snapshots, scripting, and PXE booting. Behind the scenes there have been numerous tweaks, such as 10-20 per cent faster networking and disk performance, the ability to burn CDs from the guest environment, full user- and kernel-level debugging support, as well as support for kernel 2.6. These 'tweaks' alone are pretty big features, and would probably sell the upgrade by themselves, but wait till you hear what the *real* big new features do

Snapshots allow you to backup a system, while running, at any time you want. You can then restore this over the original system, or use it as a clone and run two identical systems. While the main use for this is likely to be archiving, it also means you need never worry about upgrading an operating system again – clone a logical OS, upgrade the copy while the original is still running, and if everything is good you swap them over.

The new scripting technology allows you to control virtualised OSes remotely. At the most basic level, this means you can start the OS through another program, but more advanced users can script their machines to start up, download some software, run a battery of tests, upload the results to a QA server, then shut down. If you have a number of operating systems to test a single product on, this allows you to do all of them, every night, with ease.

Finally, PXE booting allows the virtual machine to scan the network for possible boot images. Without this, you need to put a CD into the server's optical drive to install each OS, which usually entails a heck of a lot of walking! With PXE, you can store network installation images on one server, and save yourself the hassle.



Virtual Machine Settings - Windows XP Professional

Hardware | Options

Device	Summary
Memory	130 MB
Virtual Disk (IDE 0:0)	Compact
DVD/CD-ROM (IDE 1:0)	Using drive /dev/cdrom
Floppy Drive	Using drive /dev/fd0
Network Adapter	Bridged
Mouse	auto detect

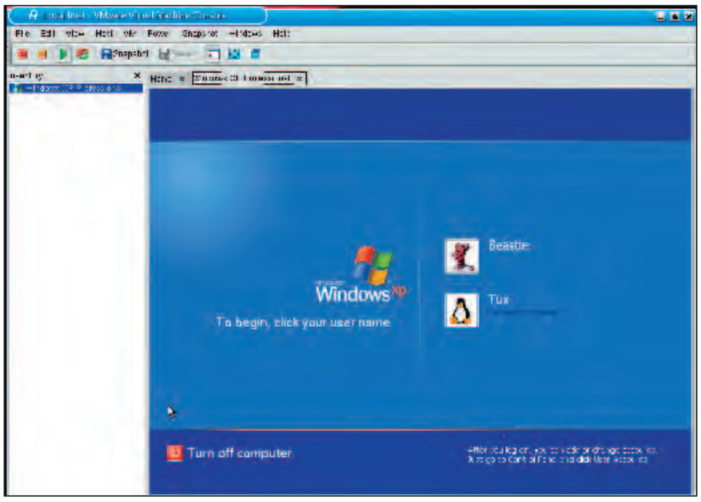
Memory

Specify the amount of memory allocated to this virtual machine. The memory size must be a multiple of 4 MB

Memory for this virtual machine: 160 MB

Guest OS recommended minimum	120MB
Recommended memory	160MB
Maximum for best performance	180MB

You can customise the resources assigned to each virtual machine, down to the megabyte of RAM using the simple set of GUI widgets.



Windows XP in *LXF*?! *VMware*'s main interface has a comprehensive set of menus to keep the most common functions close to hand.

Virtually automatic

Although installation isn't done through a GUI, the text-mode prompts are all clearly explained, with very few questions to answer. The installer automatically scanned our system to figure out the correct packages to install, and it worked first time.

Once installed, the GUI itself is all written in *GTK* and is very easy-to-use. Although there is documentation available, the interface is intuitive, clear, and appealing, which makes life easy even without reading the help files. We did test installs of Windows Server 2003, Windows XP, Debian, Red Hat Enterprise Linux 3, and Mandrake 10, and all worked flawlessly. What's more, the performance was actually quite impressive – at least 75 per cent of the speed the system would have been if it were running natively.

Much of the advanced functionality is transparent, which shows just how well this product has been designed. For example, connecting to remote virtual machines is done using a dialog, but once you're connected it looks and works the same as a local virtual machine – even down to allowing you to use your local CD ROM drive to install software on the remote host. This is all encrypted using *SSL* so that the data sent across the wire is kept safe.

Looking at this, you'd think hosting several operating systems on one OS is *easy*, but that just shows how much work the developers have done to take away the potential stress involved with virtualisation. The best part of all this – and that's saying a lot given the quality of the other features – is that customers of the older version of *GSX Server* get a free upgrade to this release.

This is a very impressive product through and through, and comes at a bargain price when you consider the power and flexibility you get in return. If you have a multi-platform QA commitment, or just want to make your IT infrastructure flexible, this is the easiest way to get what you need. LXF

LINUX FORMAT VERDICT

FEATURES	9/10
PERFORMANCE	10/10
EASE OF USE	10/10
VALUE FOR MONEY	10/10

A powerful product that lives up to its goals without breaking the bank – if you're using the previous version, get in touch with VMware for your free upgrade.

RATING 10/10

●●●●●●●●●●

RTF TEXT APPLICATION

TuxCards 1.1

When that pile of garishly coloured, semi-sticky bits of paper has become just too much to bear, do the sensible thing: get rid of them and make notes on your computer, says Andy Channelle.

BUYER INFO

Hierarchical desktop note organisation, similar to *Knowit* for KDE or *CueCards* on Windows.

- **DEVELOPER** Alexander Theel
- **LICENCE:** GPL
- **WEB** www.tuxcards.de/

There are some applications that insinuate their way into your life without you ever really realising it. Not the big obvious things like email, but something small that fills a little gap in the structure of work before anyone realised there was a gap. *TuxCards* is one such application – on my system, it's gone from curio to essential application in just a few short weeks.

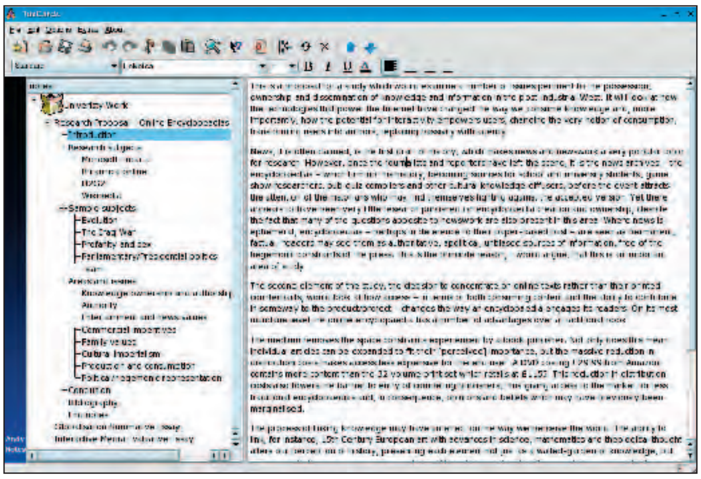
TuxCards is simply a note-taking application that falls in the broad space between a Post-it™-style applet such as *KNotes*, and a basic word processor, grabbing the best features from both while staying agile and bloat-free. Long may it stay this way.

The application is downloadable from the developer's website (see the URL above) and is available as source, SUSE 9.0 RPM, Debian .deb or a very welcome static binary. We tested installations from the RPM and experienced no difficulties when we compiled from source.

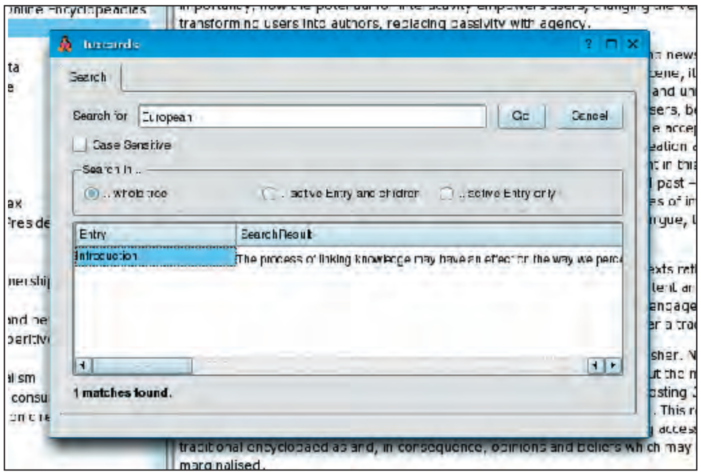
At its heart, *TuxCards* is a no-frills RTF text processor complete with font selection, bulleted and numbered lists, and basic style options. This is coupled with a hierarchical tree browser that allows the user to break individual elements of a project into smaller chunks, thus revealing – and making totally editable – its structure. There are many uses this would be good for, but *LXF* has found it indispensable as a composition tool.

For instance, in the creation of a review or research paper, it is possible to set up each part – introduction, features, conclusion, *etc* – as separate entries within the file. Clicking on any part in the left pane instantly opens up that element for editing. Entries can also be nested or moved allowing complete freedom in relation to the text. Yes, you can do the same in *OpenOffice.org*, but this works on a global scale, and doesn't involve searching and selecting the right text: each element is associated with its heading and can be moved, demoted or promoted at will.

One surprise I encountered was that using *TuxCards* instead of *OOo* encouraged me to think more about the structure of a piece of work before actually getting down to writing it, though it's flexible enough to allow major alteration later on. Seeing the tree in the left pane provides a completely different view of a document, especially when the project is complex and gets beyond a few thousand words. It soon proved more convenient to do most of the writing in *TuxCards*, moving over to *OOo* one section at a time via cut and paste for a final proof, spellcheck and output in an email-friendly format.



Main interface is easily understandable, uncluttered and instantly usable.



tuxcards

Search

Search for: European [Go] [Cancel]

□ Case Sensitive

Search in ..

◉ .. whole tree ○ .. active Entry and children ○ .. active Entry only

Entry	Search Result
Introduction	The process of linking knowledge may have an effect on the way we perc

1 matches found.

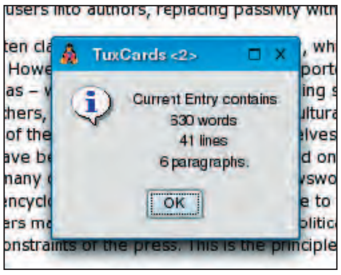
The *TuxCards* Search facility is simple but powerful.

In terms of output, data is stored in an XML file, and while the 'convert to ASCII' feature is not yet available, there is a pretty cool export to HTML option which creates a frameset for the archive, with the document tree set left and the content on the right. We did find this a little flaky in that occasionally pieces of text – especially some longer entries – sometimes didn't survive the porting process. We couldn't find a reason why, but it is something to keep an eye on if you intend to use this feature.

Conclusion

One the whole, this is a very useful bit of software. If you write anything beyond the odd letter or email, it may help you improve the flow of both your ideas and the text itself, and if you have a novel or dissertation to write the ability to structure and refine work plans could be a life saver. Small features, such as the ability to word-count individual entries, drag-and-drop text between entries, and search the entire hierarchy for a specific text string can have a big impact. For example, the Search feature is useful for checking whether you've overused an expression or repeated a point, without you having to comb through the entire text. It is also effortless to use and doesn't hog system resources as a full-blown word processor would.

Beyond the few niggles mentioned already and the lack of standardised keyboard shortcuts, the only really disappointing thing about *TuxCards* is that I didn't discover it sooner! LXF



TuxCards <2>

Current Entry contains
530 words
41 lines
6 paragraphs.

[OK]

Count the words in each entry.

LINUX FORMAT VERDICT

FEATURES	7/10
PERFORMANCE	8/10
EASE OF USE	10/10
DOCUMENTATION	5/10

As simple as it needs to be. If you spend any time writing for work, school or university, *TuxCards* could soon become an indispensable tool.

RATING 8/10

●●●●●●●●○○

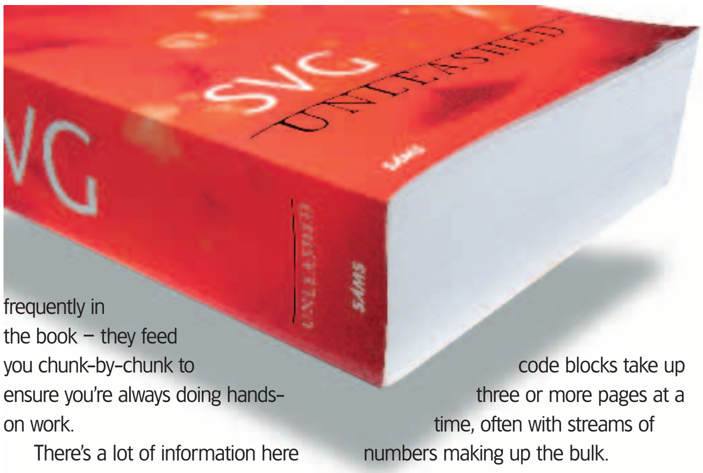
SVG Unleashed

Paul Hudson reads something that's been missing from the market until now...

LINUX FORMAT TOP STUFF AWARD

BUYER INFO

- **AUTHOR** Andrew Wyatt, Chris Lilley, *et al*
- **PUBLISHER** Sams
- **ISBN** 0-672-32429-6
- **PRICE** £25.55
- **PAGES** 1117



File formats, for the most part, aren't complicated. You hit 'File', then 'Save As', and you're already most of the way there. So how come it takes eight people and 1100 pages to describe Scalable Vector Graphics (SVG)?

Sams' *Unleashed* has always been a 'no-holds-barred' series that has gone into every detail about each topic, and this edition is no different: after just a handful of pages you're knee-deep in code and creating starter pictures. *"Code?"* I hear you say, and, yes, that's correct – SVG is an XML-based vector format where shapes, colours, gradients, and animations are all encoded in plain text for maximum portability. As a result, this book kicks off with a comprehensive primer to XML that is just enough to get you going with SVG itself. The topic of more advanced XML is returned to frequently in the book – they feed you chunk-by-chunk to ensure you're always doing hands-on work.

There's a lot of information here about using SVG with other tools – Java features prominently, as does PHP, but you'll also find Perl and XSLT covered in depth, which means this book gives you a lot more punch than just how SVG works. Being a book about graphics, there are screenshots scattered throughout – sadly, Windows-centric – but this ease of reading is largely offset by the chunks of code that is inevitable in a book dedicated to a text-based picture format. At one or two points in the book, the code becomes a little overbearing, and this is most obvious when describing animation – here the code blocks take up three or more pages at a time, often with streams of numbers making up the bulk.

Part of the appeal of SVG is that it is potentially able to displace the proprietary *Macromedia Flash* as a standard vector animation format, and this is given a lot of play here through the ECMAscript (*née* JavaScript) coverage. Although all this has been standardised, the book often takes the opportunity to stray away from the official standard and onto platform-specific functionality – numerous references are made to the *Adobe SVG Viewer* for *Internet Explorer*, which supports a superset of SVG functionality not (yet?) standardised.

The last chapter is particularly interesting, as it is where the authors list various ideas and proposals for future versions of SVG. In other books, the 'looking forward' chapter usually consists of two paragraphs hastily tacked on to tick a few managerial boxes, but not so here. Instead this is a very detailed mix between fact and wish list that makes for compulsive reading, and a great finish to a great book.

Yes, the book does show all its examples in Windows, but the code and content is so wholly cross-platform that we barely noticed the difference. Making a big book of SVG sounds so difficult on paper that the very fact this book is here and is a good read as well should make the authors pleased, but beyond that it's something the market has been sorely missing until now.

LINUX FORMAT VERDICT

Damn heavy, but it needs to be – if you're looking for the comprehensive guide to SVG, our *Top Stuff* award shows we recommend this tome highly.

RATING 10/10

●●●●●●●●●●

Grid Computing: A practical guide to technology and applications

Nick Veitch descends *Tron*-like into the technology of the future...

BUYER INFO

- **AUTHOR** Ahmar Abbas
- **PUBLISHER** Charles River Media
- **ISBN** 1-584-50276-2
- **PRICE** £24.66
- **PAGES** 406

Wherever possible, the book tries to back up the claims for grid computing by giving concrete examples. The subtitle belies a lot of the text in this book. Although the practical aspects of grid computing are covered, a lot of the early material is actually about the genesis of grid computing and the modern theory of how it should work. There are plenty of examples given here, and the reader will develop a real sense of what possible benefits grid computing could have in the real world.

Then it moves on to a discussion in more detail about different types of grids. Desktop grids are covered in some detail – presumably as this is the area that most people are aware of the technology working with, and its the one that gets the most media attention.

The examination of grid computing methods in this and the other environments mentioned (HPC and Clusters) is pretty thorough, and really gets down to the sort of tasks which are possible, and beneficial to run on such systems, as well as other considerations about grid deployment (will desktop users mind /notice the paradigm-shift?).

Superficially, this book promised practicality, for which the reader might expect to read pages of example code and fairly dull explanation. However, in reality, the supplied code is sparse and only the high level aspects of creating grid applications are covered. What this book is though, is a very good overall introduction to grids, how they work and how you might deploy them.

LINUX FORMAT VERDICT

Good overall summary and explanation of the current state of Grid technology, but if you want practical examples, you'll have to look elsewhere.

RATING 7/10

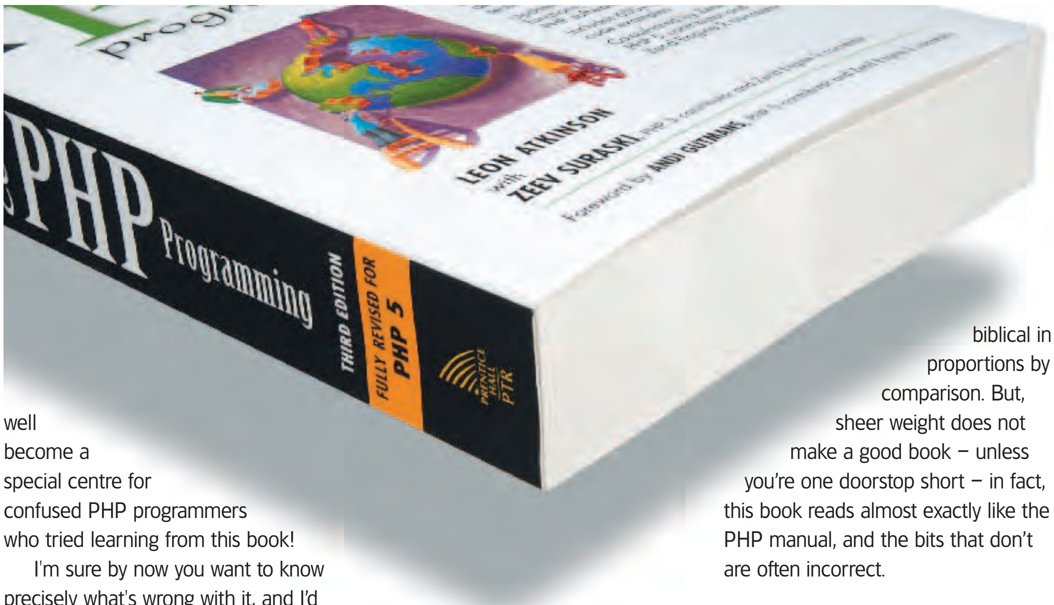
●●●●●●●○○○

Core PHP Programming 3rd Edition

Paul Hudson reviews what is arguably the worst PHP book to date…

BUYER INFO

- **AUTHOR** Leon Atkinson with Zeev Suraski
- **PUBLISHER** Prentice Hall
- **ISBN** 0-130-46346-9
- **PRICE** £35.99
- **PAGES** 1072



By the time technical books get to their third edition, they are usually well-seasoned, well-rounded, and well-liked by their respective communities. After all, if they were a load of tat, any poorly written book would flop in its initial print run and thankfully never be heard from again.

Not so, it seems, with *Core PHP Programming*. Loathe as I am to publicly express any sort of dislike on a PHP-related topic, I was seriously considering sending my free review copy of this book back in disgust. As it is, Bath's Royal United Hospital may well become a special centre for confused PHP programmers who tried learning from this book!

I'm sure by now you want to know precisely what's wrong with it, and I'd be happy to tell you. At 1072 pages, this book looks quite comprehensive on the surface. Even the venerable *PHP and MySQL Web Development* by the Luke & Laura tag-team (Sams, ISBN 0-672-32525-X) is 860-odd pages, so this book is of positively biblical in proportions by comparison. But, sheer weight does not make a good book – unless you're one doorstop short – in fact, this book reads almost exactly like the PHP manual, and the bits that don't are often incorrect.

PHP 5? We wish!

"But Zeev Suraski, lead developer of the PHP project, wrote it!" And yes, that's what made me eager to read the book

in the first place – Zeev is, after all, a master guru of PHP and rightly well-respected in the community. However, I took the time to question Zeev about his involvement in his book, and he was keen to point out that the book is by "Leon Atkinson *with* Zeev Suraski" not "Leon Atkinson *and* Zeev Suraski". If you're thinking the flaws are minor, don't kid yourself – in an effort to "fully cover" PHP 5, the book discusses how to use namespaces, which is a feature that didn't make it into the final build by many months.

Missing inaction

However, what's more important is what the text misses: despite being marked *"fully revised for PHP 5"*, you won't find many of the brand-new PHP 5 features discussed here. *SimpleXML*? Nope. *SQLite*? Not a word of it. *MySQLi*? What's that? There is so little PHP 5 information here, once you ignore the incorrect inclusion of namespaces, that someone, somewhere should take Prentice Hall up on a case of bad advertising.

Now, if you're a *PHP 4* user, the situation is a little different. If copyright laws would allow me to do so, I would scan an example of this to show you. Take pages 588 and 589, for example – just average pages, mind you. On these two pages, you'll find no fewer than ten functions introduced and discussed, of which *only one* has a code sample to go with it. The others have two sentences on average, and that's supposed to be enough. Unless the author thinks "enough" is *"enough to drive readers back to the online manual, which is much more informative"*, it's clear the two of us disagree quite sharply. Even at this point, there's hope, though – what if those two or three sentences are so massively filled with insight that the book is a veritable joy to read? Fat chance. Take the definition of the **session_encode()** function – here's the 'insight' that Atkinson gives us for our £35.99: *"The session_encode function returns a string that contains encoded information about the current session"*. That's it – no more, no code example, and not even an explanation that isn't flat-out obvious from the name of the function.

You wouldn't be alone if at this point you wondered whether the price simply had the decimal point off by one place, although I'd recommend you still to think twice if someone offered this book to you for £3.59!

Just plain wrong

Sadly, and this is perhaps the most frustrating part of all the book (which is saying a lot), very often the handful of meagre sentences Atkinson devotes to each function as he rushes over them are wrong or incomplete.

Take the **pcntl_exec()** function, for example – it executes a program in the current address space, effectively halting the PHP script at the function call and replacing it with the new executable. Does Atkinson mention this? No. Apparently *"The pcntl_exec function executes a program"*, and little more – almost as if he hadn't bothered to try it himself. It should have been obvious to the editors of this book – who really should be hanging their heads in shame – that it would have been better *not* to have covered the **pcntl_exec()** function (as well as dozens of others) and use the extra room to cover the remainder properly. Instead, what we have is a book whose content seems to have been put together by way of checkboxes, with the thinking being that the more topics it manages to cover, the better the finished product must be.

If this is really the best we have in the PHP community for books, things are clearly lacking. I don't know about you, but when I read a book I want to think *"Yeah, this guy really knows his stuff"*, not *"Wait a minute, namespaces aren't even in PHP 5, what's this guy been smoking?"*

Perhaps the fourth edition of this book – if it ever comes out – will correct these insulting flaws. However, having read this has put me in such a terrible mood that I've decided I'm going to be better off writing a book on PHP *myself* to try and bring up the general standard. LXF

LINUX FORMAT VERDICT

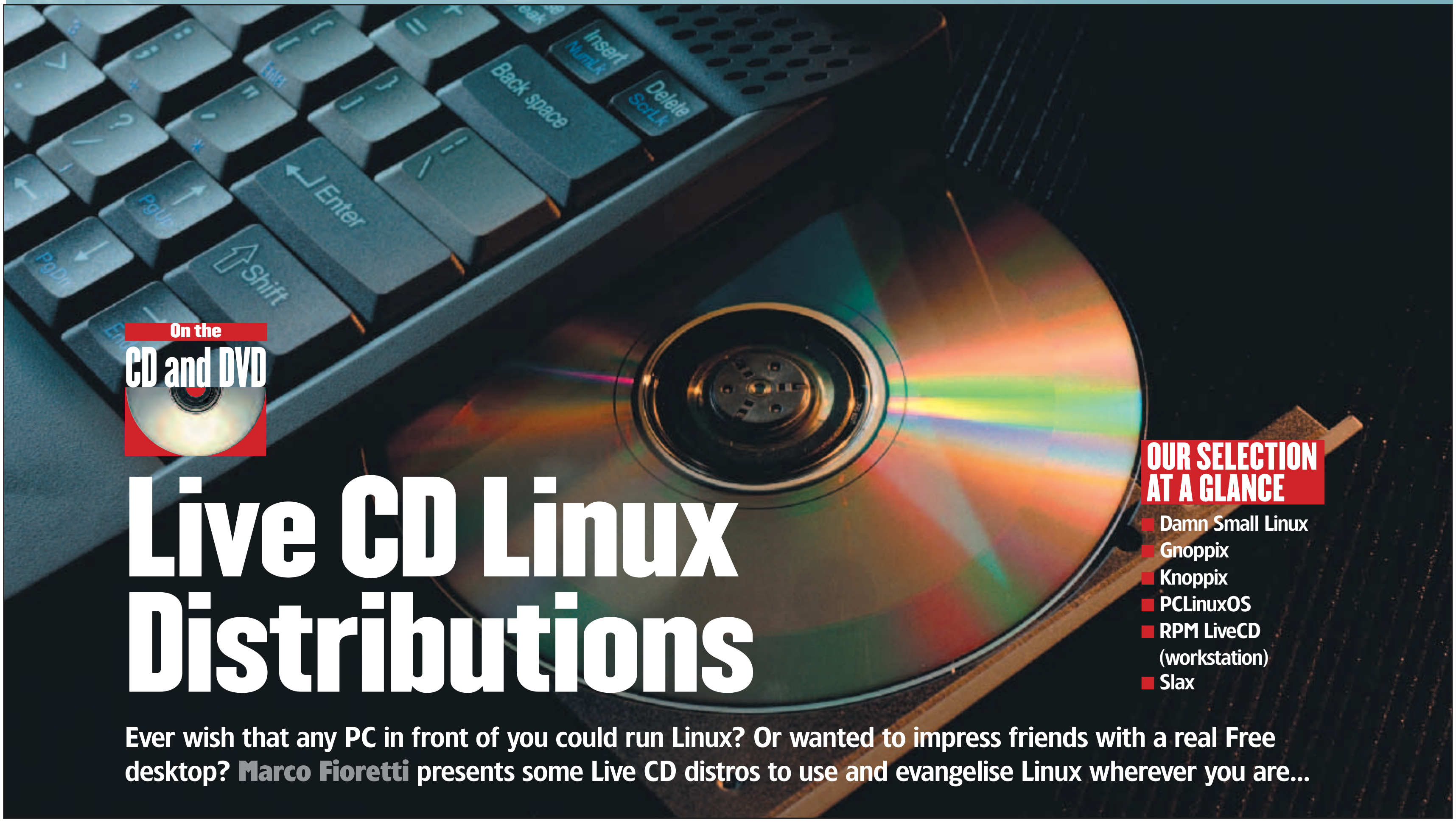
We argued long and hard about whether we were going to give this book review any space in the magazine – finally deciding that everyone must be warned...

RATING 3/10

●●●○○○○○○○

Roundup

Every month we compare tons of software, so you don't have to!



On the CD and DVD

Live CD Linux Distributions

OUR SELECTION AT A GLANCE

- Damn Small Linux
- Gnoppix
- Knoppix
- PCLinuxOS
- RPM LiveCD (workstation)
- Slax

Ever wish that any PC in front of you could run Linux? Or wanted to impress friends with a real Free desktop? Marco Fioretti presents some Live CD distros to use and evangelise Linux wherever you are...

These days, we happy Linux aficionados have no excuses anymore not to run our favourite OS everywhere, even if we didn't bring a computer with us! Even more importantly, your parents, relatives, bosses or teachers are also now officially right out of excuses not to at least give a *real* operating system a try. How? Live Linux distributions are one of the most exciting Free Software trends of this year.

These bootable CD-ROMs contain complete Linux desktops that start and run just like that, without touching or installing anything on the hard drive. The basic concept is nothing new. Specialised servers or rescue systems on floppies, USB sticks or CD-ROMs have been around for a while now. The new spin is the flourishing of a lot of versions geared to full graphical desktops. The Live Linux section of **www.distrowatch.com** shows several distributions for all imaginable uses: firewall, clusters, scientific workstations, portable jukebox, you name it – it's there! The six Live distros featured in this roundup descend from four of the most popular Linux flavours: Debian, Mandrake, Red Hat/Fedora and Slackware. The other common characteristics are that they focus on the desktop, and are volunteer-only projects. At the time of this writing, Mandrake, Lindows and SUSE also all offer Live versions of their products.

The six distributions were tested on two computers: a Linux-only K6-2 CPU (350MHz, 128MB RAM) and a Windows-only 900MHz Duron (256MB RAM).

How do they work?

Before we start, let's take a peek under the bonnet for a moment. The basic architecture of Live CD distros is always the same, and is based on the simple concept of ramdisk. A ramdisk is nothing more than a portion of the available RAM that is formatted and then used as if it were a partition of an actual hard drive. This makes it possible to uncompress and run more software on-the-fly than would otherwise fit on the original medium. It also makes all tools that need to write logs or other stuff feel at home, er... we meant /tmp. The other pieces most commonly used are the bootloader *Isolinux* (**http://syslinux.zytor.com/**), the *ovlfs* kernel modules (**http://sourceforge.net/projects/ovlfs**) and the scripts found at **www.linux-live.org/**: the latter allows you to create a Live CD from just about any Linux distribution. *ovlfs* (the acronym stands for *Overlay File System*) allows all img files to be 'pseudo-writable.' The final piece of the puzzle, *IsoLinux,* can use and boot from disk images of every size.

A Live CD boots by loading the kernel via *Isolinux*: the basic filesystem is then created in an initial ramdisk. This will contain at least device files, and all base system settings. Right after that, binaries and libraries are mounted directly off the CD-ROM, but with tricks that make them appear in the usual locations (for example, /usr/bin instead of /mnt/cdrom/usr/bin). From then on, everything else will work almost as if it were a normal Linux system. One thing to keep in mind is that any Live Linux will run considerably more slowly than if the same system had been installed on the hard drive (especially if there is little RAM). In practice, depending on the design and the available hardware, several programs could be run off the CD or be pre-loaded in another ramdisk.

Damn Small Linux

A powerful desktop, but reserved to Linux experts.

■ **VERSION** 0.5.3.1 ■ **WEB** www.damnsmalllinux.org/

ON THE CD

At 50MB, Damn Small Linux (DSL) is by far the smallest distro in this roundup. This project, started because *"it is just too cool a thing not to do"*, sacrifices user-friendliness if needs be, but always trys to include the best compromise between size and functionality. *DSL* is (indirectly) derived from Knoppix, and can also be installed on your hard drive, a great choice for shoe-horning Linux onto older hardware.

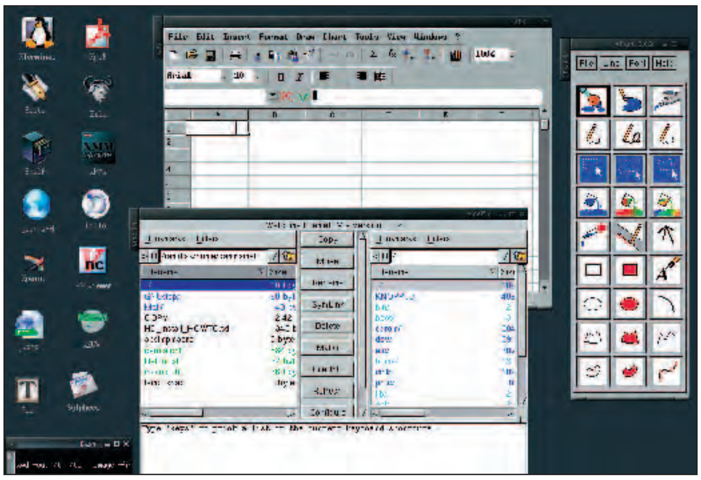
The boot procedure requires some manual input, but is still the fastest of the distros in this roundup: X server (*Xfbdev* or *XVesa* from *KDrive*), mouse type, screen resolution and colour depth must be selected. The user account (**damnsmall**) is enabled to mount and unmount devices.

The desktop is spartan (*Fluxbox* is the window manager), but not bare. By default, icons are missing, but there is an entry in the system menu to enable them. As a matter of fact, DSL is a good compilation of several little-known gems that should be used more widely.

Email and news are managed by *Sylpheed*. The web browsers are *Dillo* (300KB binary, enhanced with *SSL*, tabs and frames) and *Links* (also patched for *SSL*, JavaScript, frames, cookies and HTTP Authentication): more than capable enough to check a bank account or shop securely online.

Ted-GTK and *ABS* take care of word processing and spreadsheets. The first supports RTF and can save in PostScript or PDF format. The export filter and macro language of the latter are compatible with *Microsoft Excel* and Visual Basic. Simple image editing is handed over to *Xzgv* and *XPaint*. The database program is the server-less *SQLite* (**www.sqlite.org**). CD-ROMs can be burned with *Bashburn* (**http://bashburn.sf.net**). The file manager *emelFM* (reviewed in *LXF39*'s *Roundup*) features a built-in command-line and customisable menus. Want fun? *XMMS* is also just one click away.

Hardware detection is not the strongest side of DSL. A USB stick was mounted easily, but attempting to use a printer or an Internet connection (PPP or ADSL) means editing configuration files of programs like *apsfilter* or *wvdial*. Nobody questions the power of the command-line, but for a system that will probably be booted on different/unknown hardware every time, this can be a problem: especially when – to save space – all docs and locales are missing.



DSL proves that real functionality exists even when eye candy is absent.

LINUX FORMAT VERDICT

FEATURES	6/10
EASE OF USE	5/10
DOCUMENTATION	5/10
PERFORMANCE	9/10

Ideal if performance is the highest priority, but day-to-day use will mainly be reserved for expert Linux users, or those really willing to learn a different approach to the desktop.

RATING 6/10

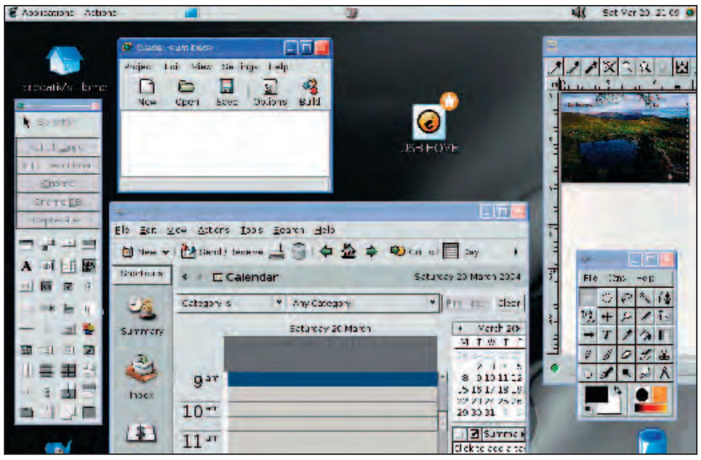
●●●●●●○○○○

Gnoppix

An experimental but promising GNOME for every PC.

■ **VERSION** 0.6.0-rc1 ■ **WEB** www.gnoppix.org/

The name doesn't mean that this is derived from Knoppix: it is a similar product, but based on Debian and the GNOME desktop. The XML-based *Discover* manages hardware detection (**http://platform.progeny.com/discover/**). Configuration happens through the GNOME system tools.



Unlike some other Live Linux systems, Gnoppix defaults to a full GNOME 2.4 desktop environment, with good hardware detection capabilities.

Gnoppix doesn't require a password to log in, which is understandable in a system of this nature. Arabic, Dutch, Urdu, English, French, German, Spanish, and Turkish are supported, but locale management is not perfect: even if a user chooses English, the keyboard is considered German. Luckily, clicking on the Key icon with flags in the GNOME panel changes the keyboard layout.

Evolution and the GNOME database frontend, *Mergeant*, are included. Sadly, the most important aspect of a business Linux desktop could not be utilised. *OpenOffice.org 1.1.0*, started the install procedure, but was unable to finish it. Multimedia apps all installed though: *MPlayer, Xine, XMMS* and *Ogle*. On systems with only one CD/DVD player, however, their use is limited to files on the local hard drive, since the system CD can't be extracted during operation (a limit common to the other distros too). Browsing the menus shows tools as different as a teletext viewer and development environments (*Glade*, the *Anjuta* IDE and others); and several TV cards can be configured, if present. Gnoppix correctly recognised and controlled an UMAX Astra 610 model, providing a shortcut to it from *The GIMP* via *XSane* – in *LXF51*'s Hardware feature, standard install tools failed to do this on the same system! The printer, though detected, was not available to apps, Internet access required root privileges, and the reboot and shutdown commands froze the machine – only the PC reset button restarts the system.

One very promising feature – still in alpha stages – is the possibility to store the home directory and all system configuration data in an ext2 partition on an USB stick. The partition can be encrypted with the AES256 algorithm, and its size is configurable.

LINUX FORMAT VERDICT

FEATURES	6/10
EASE OF USE	8/10
DOCUMENTATION	6/10
PERFORMANCE	5/10

As a relatively new project, Gnoppix is still finding its way, but is already largely useable, and a good way to try out GNOME with tons of potential.

RATING 6/10

●●●●●●○○○○

Knoppix

ON THE CD

A rescue CD par excellence – it's a good distro too!

■ **VERSION** 3.3 ■ **WEB** www.knopper.net/knoppix/index-en.html

The Knoppix Live CD-ROM contains gigabytes of compressed software to do everything from entertainment to office tasks, development and running servers: *KOffice, AbiWord, OpenOffice.org 1.1.0, Gnumeric, Scribus, Mozilla 1.5, Wine,* many ASCII editors; Movies apps *Xine* and *VideoLAN;* KDE-Edu packages and some games round things off nicely.

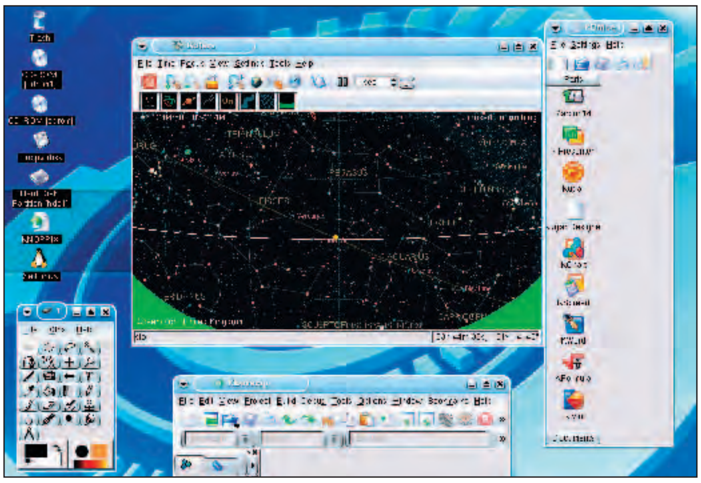
The *KDevelop* IDE can be used, even if the docs are missing: when started, it complains that it cannot find the docs and cannot index them due to the lack of *htdigs* and/or *Glimpse*. Other devel tools include CVS or RCS interfaces and several compilers for C, C++ and Java; Knoppix is also useful for rescue and analysis purposes: backups, network and security analysis are the job of apps like *Amanda, Ethereal, Mondo* and *Partimage*. The system can also be set up as an *ssh, Samba* or *NFS* server.

128MB RAM or more is recommended: the official minimum system requirements are 20MB RAM for text mode, 96MB for KDE. We found Knoppix to be one of the slowest Live CD-ROMs though, especially to launch apps. This is probably caused by the fact that all software must be decompressed from the CD-ROM before starting.

There are two users: **root** and **knoppix**. When you need to act as root (to mount a floppy, for instance), launch a root shell from the system menu and no password will be requested.

The boot phase of Knoppix is also quite long. Hitting **F2** will display the available 'cheatcodes' to change window manager, keyboard, screen resolution, hardware auto-detection, *etc.*

Knoppix's superior hardware detection over all the other live distros seems to be confirmed by our tests, and excuses the longer boot time. Should the procedure freeze during a boot, it is possible to bypass any section of it. All our hardware was recognised, with the exception of one soft modem. Another glitch is the fact that the default print daemon (CUPS) wasn't started by default. ADSL, PPP and ISDN can be set up graphically. All hard disk partitions were automounted without issue and associated to desktop icons.



Education, work, fun and software development... Knoppix really has something for everybody, so is ideal for winning over Linux sceptics.

According to the website, the next version of Knoppix will add such features as: the 2.6 kernel (optional), a new setup for IRDA and Bluetooth peripherals, and support for Internet connections through GPRS.

LINUX FORMAT VERDICT

FEATURES	9/10
EASE OF USE	8/10
DOCUMENTATION	8/10
PERFORMANCE	6/10

The most complete product of this selection also has the best hardware detection – which is why its so widely popular.

RATING	8/10

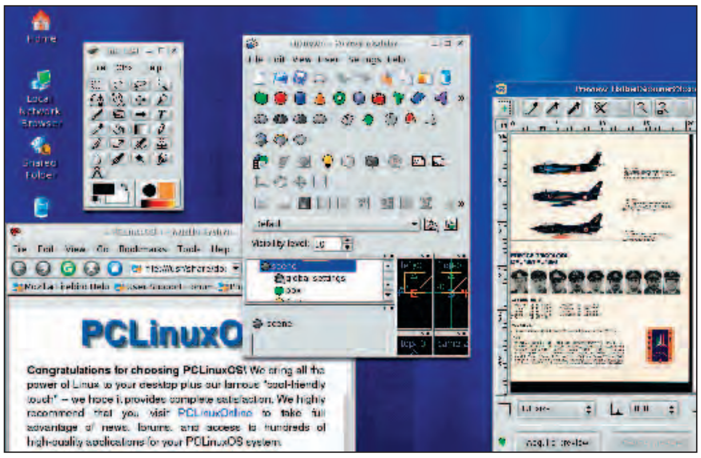
●●●●●●●●○○

PCLinuxOS 2K4

MDK 9.2 derivative with some promising ideas.

■ **VERSION** Preview 5 (1.20.04) ■ **WEB** www.pclinuxonline.com/pclos/

The review version is only in English, but support for other languages is planned. The boot screen is friendly: it directly lists what options are available, and which are the defaults. A nice feature is the possibility to declare if local partitions are to be mounted as writeable (the default) or not. Pressing the **Escape** key during boot shows what happens behind the scenes. Ethernet and sound card were correctly detected, and it was even possible to access the scanner from within *The GIMP* right away. There were some downsides: the printer (Epson) was not useable; drivers for some Winmodems are present but not integrated yet, though the presence of Nvidia drivers will make many users happy. The Windows partition was found and mounted, but the guest account can't mount or access the floppy, only root. For both accounts, passwords are equal to account names.



Congratulations for choosing PCLinuxOS! We bring all the power of Linux to your desktop plus our famous "cool-friendly touch" – we hope it provides complete satisfaction. We highly recommend that you visit PCLinuxOnline to take full advantage of news, forums, and access to hundreds of high-quality applications for your PCLinuxOS system.

Nice icons, very readable fonts : PCLinuxOS is one of the most pleasant systems to work with, although it's quite heavy on the hardware

The network configuration utility is the same as Mandrake, *DrakConnect*; but the package management system is not the same (*urpmi/rpmdrake*) but *apt-get*, through the Synaptic GUI. Setup can also happen through the *KDE Mission Control Center* from Ark Linux.

You can opt for KDE 3.1.4 (default), GNOME 2.4 or Fluxbox desktops, and the choice of apps is quite broad: the heavyweights are *OpenOffice.org 1.1* and *Mozilla 1.6* (with Flash, Java and some other plugins working). We were also pleased to find *RealPlayer 8, Scribus, Firebird, TvTime, GnomeMeeting,* the Apollon file sharing client, a GUI for BitTorrent and the 3D modeller *PovRay*.

The main drawback of PCLinuxOS is its hardware requirements. This was the only Live CD distro which wouldn't even start on the 128MB test system. The minimum is 256MB, and even then it isn't exactly fast. Installation on hard drive is under development, and will require 5-6GB. Another experimental feature is the ability to make a USB drive your /home partition. This option can be tried today by typing at boot:

```
livecd home=usb
```

LINUX FORMAT VERDICT

FEATURES	7/10
EASE OF USE	8/10
DOCUMENTATION	6/10
PERFORMANCE	7/10

Even if performance is comparatively slow, this is one of the nicest desktops here. It is not ready to trust for critical data yet though, but has fantastic potential.

RATING	7/10

●●●●●●●○○○

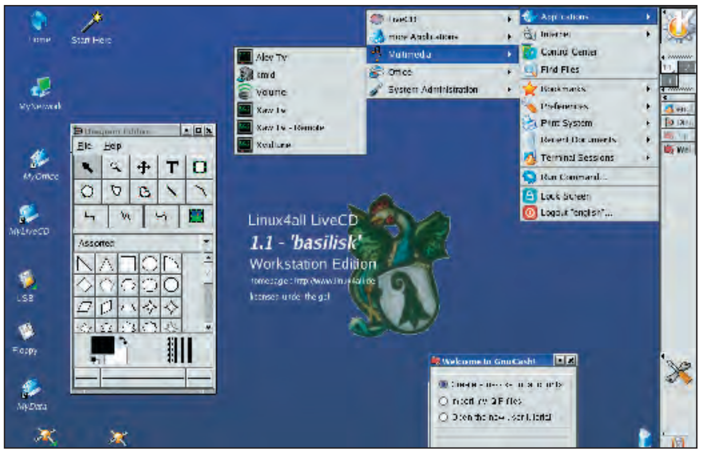
Linux4all: RPM LiveCD

ON THE CD

The most familiar environment for Fedora fans.

■ **VERSION** 1.1 workstation ("Basilisk") ■ **WEB** www.linux4all.de/

The RPM LiveCD was originally conceived as a server-oriented project, with or without graphical environment, but with support for clustering. Consequently, the server live ISO image fully supports OpenMosix and other specialised software of the same nature. While that remains the main focus, there is also a workstation version, based on the original on Fedora Core (version 1 in this case). The web site even hosts a developer kit and a fairly detailed guide to rebuild or customise both versions.



The Linux4all LiveCD workstation contains a nice selection of application for home and small office users.

Basilisk booted without problems, going straight to a graphical login prompt with several language options: Basque, four English varieties and Ukrainian. It can also load initial configuration from floppy, or stop to a console login. The screen offers the choice between KDE and GNOME, but in the current version only the first is useable. Starting GNOME just freezes the system: this is being investigated, and should be fixed in the next version. The user account is named after the chosen language, and the password is **livecd**. The root password (needed to mount removable drives and connect to the Internet via *Kppp*) is **linuxlivecd**.

Hardware detection needs further work: Basilisk didn't see an IDE CD-burner, and printer setup tools seem to be missing from the desktop. Basilisk also shares one problem with Gnoppix: even though we selected US English, the keyboard remained stuck to another language. The solution is still to change the configuration with the mouse, from the KDE panel or *Control Center*.

Almost every category of app for home and small office is represented in Basilisk, even if there's a bit less variety than in other Live Linux systems. Maybe the biggest omission in the 1.1 version is *The GIMP*. *Firebird* is included, though; but *OpenOffice.org 1.1*, while present, lacks the templates for *Impress*. Other functions worth noting are the *sshd* server and the possibility to log through KDE on a *Samba* server. The *Kickstart* tool from Red Hat can be used to save the Basilisk configuration as a basis for a normal install of Fedora Core.

LINUX FORMAT VERDICT

FEATURES	6/10
EASE OF USE	8/10
DOCUMENTATION	5/10
PERFORMANCE	6/10

A worthwhile port of Fedora to the world of Live CD-ROMs: the project is still in its nascent stages though, and not as mature as other competitors.

RATING 6/10

●●●●●●○○○○

Slax

ON THE CD

Live CD of oldest distro still developed: Slackware

■ **VERSION** 4.0.1 ■ **WEB** http://slax.linux-live.org/

Slackware users cherish simplicity and lightness: these qualities are maintained in this version. The ISO image is only 176MB and can run on a 386 CPU. The system starts in console mode – useful for rescue work or setting up some extra parameters before loading the desktop. The overall load time is the fastest after that of Damn Small Linux. It is possible to choose the language among US English, German, Polish, Portuguese and Slovak.

After logging in as root (predefined password, **toor** is shown in the initial screen) three choices are available: **gui** will start KDE, **guifast** fluxbox, **mc** *Midnight Commander*. There is also a fourth option to tune speaker volume.

As usual in Slackware, the desktop environment is KDE straight from the sources, with as few modifications as possible. All the "K" editors (*KWrite, KEdit, Kate*) and *KOffice 1.2* are aboard. The *Kontact* manager integrates *KMail*, calendaring and more: *Kopete* is the default (multi-protocol) IM client.

A source of confusion is the fact that some apps – like *Kugar* and the help files – are listed in the menu, but absent from the disc; corresponding menu entries could have been removed, greyed out, or pointed to a file explaining where to find them online. There are dedicated forums on the Slax website to redress this issue somewhat though.

Audio worked fine without any extra tweaking. System sounds were perfect, *Noatun* could play some MP3 files from the hard disk right away, and videos could be played with *MPlayer*. Screen resolution can be set on-the-fly. The CD-burner *K3b* is included, but sadly, we were unable to make it work – this may be due to incorrect user privileges or program configuration. A more likely reason could be a limit of most Live Linux systems: since they run off the CD-ROM, you can't extract them. Another problem area was printing: the 'Add Printer' menu was simply greyed out.

Slax is a nice and quick demo of how user-friendly a Linux desktop can be, even when space and CPU cycles are scarce. To use it at its best however, it is necessary to already have some Linux experience, preferably with the Slackware way of administering a system.

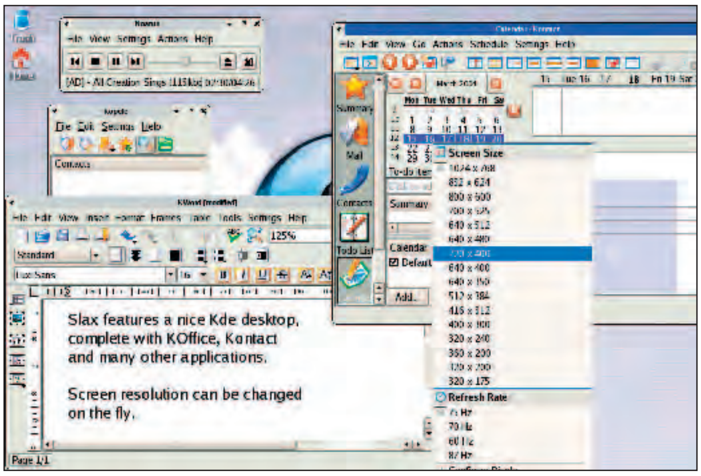
LINUX FORMAT VERDICT

FEATURES	6/10
EASE OF USE	6/10
DOCUMENTATION	6/10
PERFORMANCE	8/10

A light system, working also in console mode. Good as a demo, but not the best solution for Linux newbies however.

RATING 6/10

●●●●●●○○○○


Slax features a nice Kde desktop, complete with KOffice, Kontact and many other applications.

Screen resolution can be changed on the fly.

Slax offers the original KDE desktop.

LIVE CD DISTROS THE VERDICT

As good as they already are, all Live CD Linux distributions targeting the desktop are still relative newcomers to Penguinland, and many share the same limits. To begin with, a system like this can really be a 'Linux anywhere' solution *only* if it has a reliable way to store permanent configuration data and user files. As this roundup shows, several development teams are aware of the problems and are attacking them in the right way. For the record, this is not an issue anymore for the users of the boxed version of MandrakeMove **www.mandrakesoft.com/products/mandrakemove**.

Another feature that should really become ubiquitous as soon as possible is the capability to run Live CD distros – even temporarily – *without* the system CD in the drive; this feature is available in MandrakeMove, although the overall system remains less polished and mature than Knoppix. Allocating precious CD space to a bunch of audio and video players doesn't demonstrate Linux multimedia support if users can't put a DVD in the only drive available.

What about CD burning? With laptops and Small Form Factor computers coming more fashionable every day, boxes with two drives are likely to become more difficult to find over the next couple of years.

Even within these limits, a Live CD Linux distro can still be a lot of fun, increasing the numbers of computer users who have experienced our favourite OS, and save everybody's day in a thousand ways. What about *"My hard disk just crashed, but I can still edit and email back this OpenOffice report in 20 minutes"*? Even if you use only your computer, do yourself a favour and always keep a Live Linux CD in the closest drawer – we already do! Another indirect benefit of this kind of products is that eventually all their optimisation efforts could even be collated to increase performance in the original distributions.

Back to the roundup – Knoppix was the first project to make a complete Linux *desktop* run entirely off a CD-ROM, and this head-start means that its competitors still have varying amounts of ground to make up before they are truly comparable. To go into more detail, the reasons why Knoppix has been declared the winner are its excellent hardware detection, and the widest selection of applications. The former is probably the very first thing a Live Linux CD developer should care about: a system that realistically could be called upon to boot on a different computer every time (maybe for less than an hour) should really minimise the time needed to take control of whatever hardware it finds.

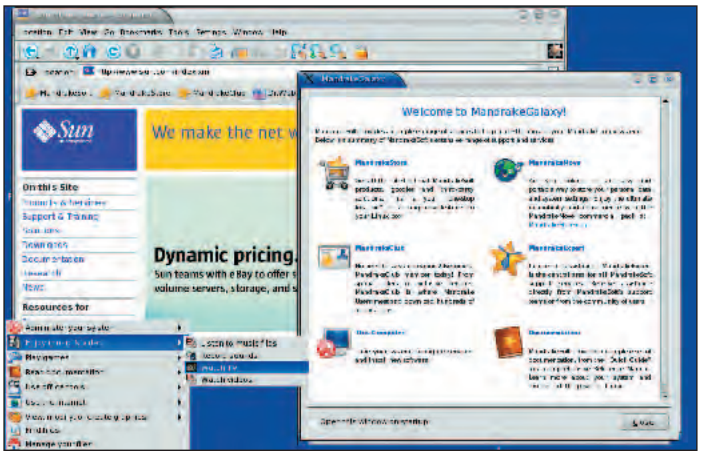
Variety of applications is also critical: it makes it possible to destroy the illusion that *"there is just not enough software for Linux"* in the quickest and most painless way. Oh, and Knoppix's greater stability didn't hurt either.

One more word about the 'Ease of Use' rating: expert Linux users will very often consider the system derived by the same Linux distribution they have been running daily for the last two/three years to be the 'easier' option, rather than addressing the issue more objectively. Our rating here, was given with respect to how complete Linux newbies would react and be able to find their way through each Live CD system. LXF

TABLE OF FEATURES

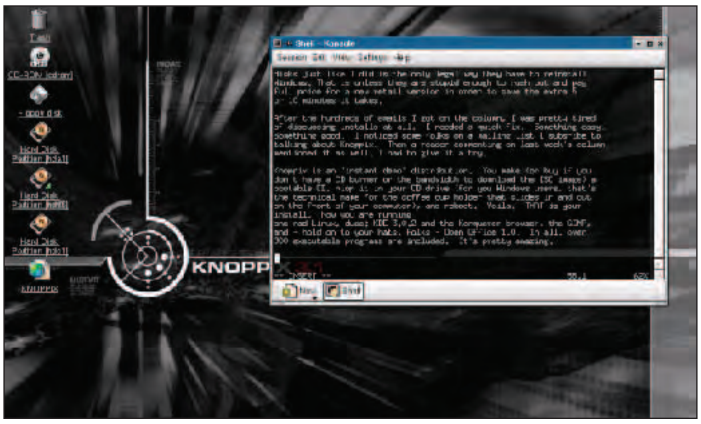
Note: The RAM recommended in the table below is what we feel was needed to run the system, which doesn't necessarily match the figure that is listed as part of the official requirements.

Name	Desktop	ISO approx size	Minimum recommended RAM	Derived from
Damn Small Linux	Fluxbox	50MB	64MB	Debian
Gnoppix	GNOME	660MB	128MB	Debian
Knoppix	KDE	700MB	128MB	Debian
PCLinuxOS	KDE, GNOME	700MB	256MB	Mandrake
RPM LiveCD	KDE, GNOME (planned)	600MB	128MB	Fedora Core
Slax	KDE	180MB	32MB	Slackware



Welcome to MandrakeGalaxy!
Sun
We make the net w
Dynamic pricing.
Sun teams with eBay to offer
volume servers, storage, and s
On this Site
Resources for

This is what MandrakeMove looks like on a computer that's got just Windows XP on its hard drive: indistinguishable from a proper Linux box!



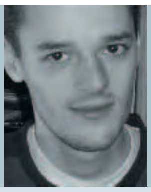
```
[illegible] just like I did is the only legal way they have to reinstall
Windows. That is unless they are stupid enough to rush out and pay
full price for a new retail version in order to save the extra 5
or 10 minutes it takes.

After the hundreds of emails I got on the column, I was pretty tired
of discussing installs at all. I needed a quick fix. Something easy.
Something good. I noticed some folks on a mailing list I subscribe to
talking about Knoppix. Then a reader commenting on last week's column
mentioned it as well. I had to give it a try.

Knoppix is an 'instant demo' distribution. You make (or buy if you
don't have a CD burner or the bandwidth to download the ISO image) a
bootable CD, slap it in your CD drive (for you Windows users, that's
the technical name for the coffee cup holder that slides in and out
on the front of your computer), and reboot. Voila. THAT is your
install. Now you are running
one real Linux, dude: KDE 3.0.2 and the Konqueror browser, the GIMP,
and - hold on to your hats, folks - Open Office 1.0. In all, over
900 executable programs are included. It's pretty amazing.
```

The CD that spawned dozens of Live CD distributions: we have the latest version of the original – Knoppix – for DVD readers this month.

HotPicks

The best new Open Source software on the planet!

On the CD and DVD



Mike Saunders

A coder since Amiga times, Mike's a Linux and BSD guru.

This is the place where we get to profile some of the hottest software around. Each month we trawl through the hundreds of Open Source projects which are released or updated, and select the newest, most inventive and best for your perusal. Most of the *Hot Picks* are available on our coverdiscs, but we've provided web links if you want to make sure you have the very latest version.

If you have any suggestions for software that you think we should cover, email us at linuxformat@futurenet.co.uk with "Hot Picks Request" as the subject-line, or contact us by post through the address on the *Mailserver* pages.

HOTPICKS AT A GLANCE

HOTPICKS AWARD

HOTTEST PICK!

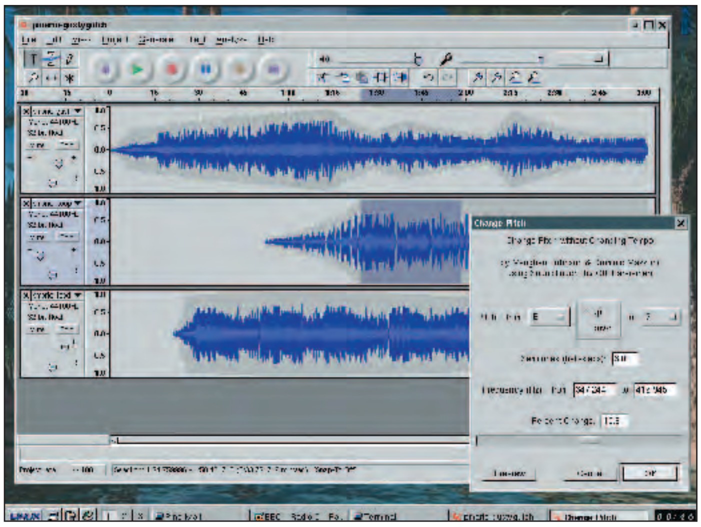
Everything covered in our *Hot Picks* section is unmissable, but every month we'll be singling out one project for outstanding brilliance. Only the very best will be chosen!

AUDIO EDITOR

Audacity

HOTTEST PICK!

■ **VERSION** 1.2.0 ■ **WEB** http://audacity.sourceforge.net/



***Audacity* in action with three tracks being worked on in the background, and the pitch-change box in the foreground.**

Sound-editing programs don't receive the same kind of mainstream attention that Web browsers and office suites enjoy, yet to many people they're unarguably essential apps. *Audacity* is the most celebrated Open Source audio editor: the 1.2.0 release recently shipped with plenty of new goodies, and we've had several suggestions from *LXF* readers to have a fresh look at it.

Thanks to its use of the *wxWidgets* (formerly *wxWindows*) graphical toolkit library, *Audacity* is cross-platform and runs on Linux, Windows and Mac OS X. Many distros include *wxWidgets* as standard – if yours doesn't have it, the *GTK* version supplied on our coverdisc will suffice. For working with other file formats than WAV, you'll also need some other supporting libraries (we've included those too). And if the worst comes to the worst, the static binary should run on most boxes.

Audacity's icon-driven interface uses a mixture of *GTK* buttons and other widgets to create a clean UI. Most buttons have tooltips or status-line hints, and the main control panel is suitably large. Toolbars can be floating, docked or be disabled in the Prefs box, and keybindings can be redefined too.

With a sound file opened, *Audacity* presents an editing pane of the waveform in which various operations can be performed. Additional tracks can be added beneath the current one – this multi-tracking capability is one of *Audacity*'s key features and puts it far ahead of simpler tools, and with the right hardware (a full-duplex sound card) it allows for simultaneous playback and recording.

The usual selection manipulation actions are available, including copy/cut/paste and 25-level undo/redo, along with a handy History box which lists previous operations. Depending on other libraries installed, *Audacity* can import and export to MP3, Ogg Vorbis and WAV, and can also read a few other formats such as AIFF, AU and RAW. MP3 files are converted when they're opened, so it doesn't support direct editing of MP3s *per se*, but the process is pretty much transparent to the end user.

Filters and effects

Audacity sports a healthy range of effects that can be applied to selections or whole projects: fade in/out, noise removal, reversing, bass-boost, echo and plenty more. Particularly smart are the functions to change pitch without changing tempo, or *vice versa* – equally, the ability to hand-draw volume curves with the envelope editor is another useful touch. Naturally, recording from various sources is also implemented, and on the whole *Audacity*'s featureset is very solid indeed.

With so much functionality, though, high-quality documentation is essential and *Audacity* excels in this area too. The online Help guide steps informatively through the basic science of sound, moves on to the process of sampling and explains the use of effects in full – marvellous stuff. The Reference manual is similarly impressive in its coverage – screenshots and diagrams abound, and migrants from other audio editing apps should have very few problems getting to grips with it.

For those who've tried *Audacity* before, new features in this 1.2.0 release include the pitch and tempo effects mentioned above, *LADSPA* plugin support for extra effects and filters, and a plethora of interface tweaks and improvements. Perhaps the most immediately useful new addition is the Preview button in most effect dialogs – this saves time when experimenting with effect settings.

We've been following the progress of *Audacity* for some time now, and we're chuffed to find that the program's performance and stability hasn't degraded as new functionality is added; happily, we experienced no crashes or major glitches during our testing of 1.2.0. *Audacity* deserves to be up there with *The GIMP* and *Mozilla* as one of the Open Source community's flagship applications, proving that loosely-knit communities of coders can produce polished and professional software. If you do any kind of audio editing, it's well worth a look.

MICRO WEB/FTP FILE SERVER

TeePeeDee

■ **VERSION** 0.3.0 ■ **WEB** http://john.fremlin.de/programs/teepeedee/

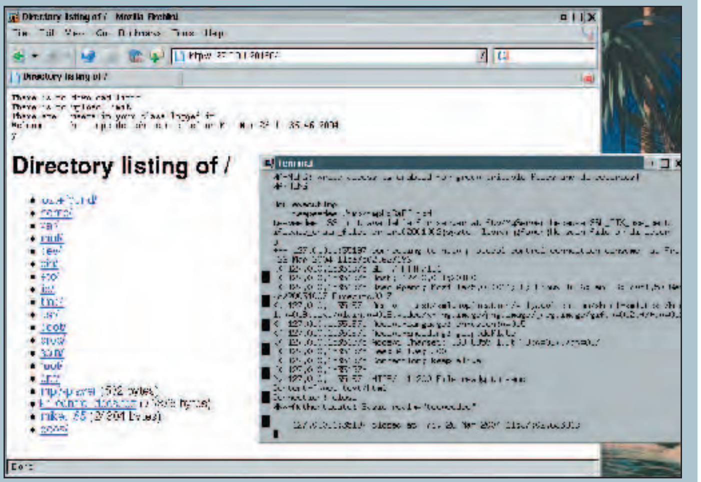
How many ways are there to transfer files over the Net? There's HTTP, FTP, email, IRC DCC, IM clients, a zillion P2P networks, and more. Still, HTTP is the most accessible and easy way to provide access to files (almost every desktop has a browser); so if you need to serve some information to the masses, a Web server is the way to go. However, *Apache* and friends can be overkill for this, and that's where *TeePeeDee* (a compact HTTP and FTP server) comes into play.

Setting up *TeePeeDee* should be problem-free – follow the usual build and install process, and it'll drop the relevant binaries into the filesystem. There are two main programs: *teepeedee*, the server itself, and *teepeedee-share*, a small Python script to control the server and create configurations.

TeePeeDee stores its settings in /etc/teepeedee, although nothing is created there during installation; you have to sort out the necessary files yourself. Fortunately, running `teepeedee-share -c/etc/teepeedee` writes the most critical config files, ready for some hand-hacking – rather than one big file, it's a configuration tree, with small files containing individual options and values.

Once everything's in place, making a directory available by HTTP, HTTPS and FTP is (hopefully!) a simple **teepeedee-share** away. If all's well, this will share the current directory (and any readable sub-directories) over the above protocols – you can test by pointing a browser at **http://127.0.0.1:20180** (the default port). *TeePeeDee* is all about file transfers, so it doesn't support CGI and the like, and as a result it's much lighter on system resources than *Apache* and co.

However, despite the feature chasm between the two, *Apache* performed much better than *TeePeeDee* in some informal benchmarking, but that's not a major issue. This utility is geared towards shifting files between systems when alternative methods aren't available (if you're sharing stuff over the Net, remember to punch the right holes in your firewall so that the ports are accessible). *TeePeeDee* does a fine job when you need an instant Web/FTP server for small-scale or personal use, with minimal hassle.



***Firebird* browsing files while *TeePeeDee* works busily away in an XTerm.**

INFORMATION MANAGER

BasKet

■ **VERSION** 0.4.0 ■ **WEB** http://les83plus.fr.st/sebastien.laout/basket/

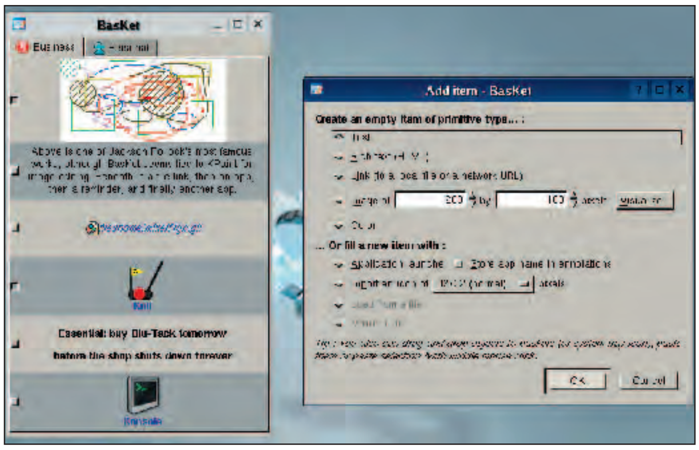
When UNIX was being created many decades ago, one of the fundamental design principles was that 'everything is a file'. It makes a lot of sense from a technical standpoint: you can **grep, cat** and **sed** your way around the system, with devices, pipes and system information all accessible via normal file operations. On the desktop, however, keeping every little bit of information in separate files doesn't always work so well – *BasKet* tries to remedy this.

BasKet was initially intended to be a clone of the Mac OS *Drop Drawers* tool, providing quick access to individual snippets of info; in other words, rather than having lots of little files for phone numbers, images and programs, a single store holds it all. *BasKet* requires KDE 3.x to run and should compile cleanly with no obscure dependencies needed.

Initially, *BasKet* pops up a rather disconsolately sparse-looking window, inviting the user to create new objects therein. Right-clicking on the main pane brings up a menu for adding, editing and deleting items – this in turn pops up a Properties box for managing the object itself. And as with many recent applications, a tabbed interface works wonders for removing onscreen clutter and organising item categories together.

Items can be created as plain (or HTML) text boxes, file links, images, app launchers and more; in addition, they can have meta-data associated with them in the form of tooltipped annotations that appear when the mouse is hovered over. Thanks to drag-and-drop, items can be shuffled about (even between tabs), and when running in KDE external files and links can be dropped in too.

It all works commendably well - separate items can be distinguished by alternating background greyscales, and editing item properties is a doddle with the easy-going dialog boxes. It's all very well executed. Moreover, it turns the desktop from being app or doc-centric to info-centric (buzzword-esque as it may sound), which for newcomers is a concept warranting further exploration. Long-time Linuxers will still appreciate *BasKet*, though, as it does a sterling job of removing clutter and providing a neat working environment.

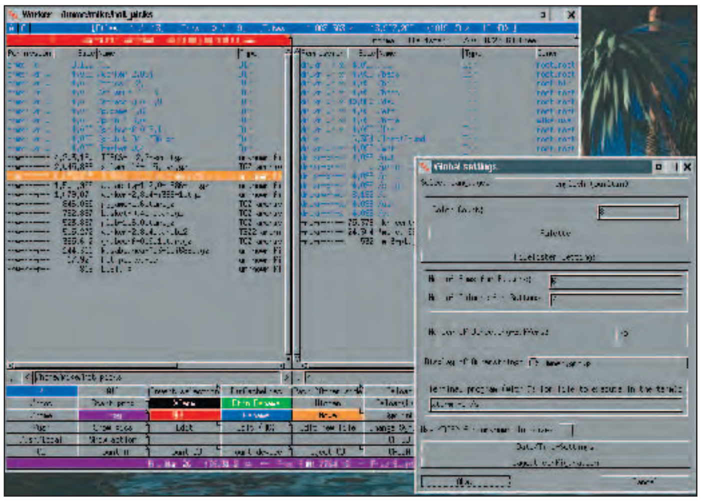

BasKet
Add item - BasKet
Create an empty item of primitive type...:
Or fill a new item with:
Essential: buy Blu-Tack tomorrow before the shop shuts down forever
OK
Cancel

It's best to run *BasKet* under KDE for full drag-and-drop goodness.

FILE MANAGER

Worker

■ **VERSION** 2.8.4 ■ **WEB** www.boomerangsworld.de/worker/



***Worker's* main window, with one of the many config boxes popped up.**

While many experienced Linux fans prefer the versatility of the command-line, file managers are very much a crucial component of modern desktop OSes and the Open Source world has no shortage of them. *Nautilus* and *Konqueror* sit comfortably on the high-end couch, while *ROX* and *gentoo* (not the distro!) provide svelte alternatives. *Worker* is an intriguing X file manager, and was suggested for inclusion in this month's *Hot Picks* by *LXF* reader Jason Lynch.

With very minimal requirements (just X and *GCC*), Worker should compile and run almost anywhere – it uses its own widget toolkit, so nothing else is required to build. Consequently, the interface is somewhat alien at first glance and doesn't match surrounding *GTK* or *Qt* applications, but the upside is a lightning-fast front-end and no dependency niggles.

Right from the start, it's clear that *Worker*'s programmers have been influenced heavily by that seminal Amiga classic, *Directory Opus*. The colourful window furniture and densely packed buttons cram a great deal of information into a small space, although the default colours are rather bold and garish. Beneath the two filesystem navigation panes lies the button bar for common operations, along with a status line; this shows a clock and free memory, or output of an external command.

In its default setup, many of the buttons have two functions, the second being accessible with a right-click (*eg* the Copy button can have a simple dialog or more detailed options). It's another good way to conserve screen space. Just about every aspect of *Worker*'s operation can be tweaked and tuned through the amazingly thorough, if slightly complicated, configuration windows – button functions, file association, keybindings, screen layout, fonts, colours and everything else imaginable. As an example of its flexibility, you can even enter a list of bytes to identify file types by their headers. Amazing!

Worker is probably the most customisable file manager we've ever seen, and as such it's not recommended for newcomers. Advanced users, if they can put up with the peculiar widget set, might adore the near-infinite ability to tweak and modify every little setting, and it's tremendously fast and reliable to boot. Worth investigating.

KDE FTP CLIENT

Kasablanca

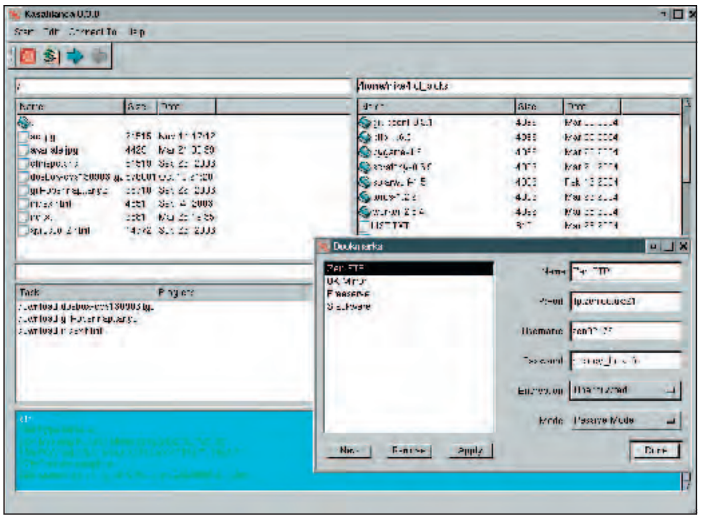
■ **VERSION** 0.3 ■ **WEB** http://kasablanca.berlios.de/

Occasionally, the developers behind KDE and GNOME applications suffer the odd jibe for over-use of Gs and Ks in program names. Some, like *Konqueror*, can be seen as reasonably creative and unique, but others ('*Gnochive*'?) don't show as much inspiration or thought. Fortunately, that has nothing at all to do with the software's quality, although *Kasablanca* is certainly an original and memorable name for this graphical FTP client.

As it's a KDE 3 app, you'll need the relevant libraries and development headers installed to build *Kasablanca* from source. If you encounter any difficulties, it could be worth trying out the RPM on our coverdisc – it's made for SUSE 9, but it installed and ran on our Slackware 9.1 test box perfectly.

Kasablanca's main window is comprised of the usual GUI upholstery found in FTP clients: a two-pane navigation section for moving around remote and local filesystems, along with a progress indication list and FTP command status box at the bottom. It's not overbearingly crammed with buttons to fiddle with, but sadly there's only one resize control, and the colour scheme for the status section makes a Bridget Riley work look positively soothing.

A bookmarks dialog is available for organising sites, which allows for active and passive connection modes and varying levels of encryption (TLS). Annoyingly, this box displays passwords in the clear, so it's not ideal if you want to do FTP work with others in the vicinity. Once connected, downloading and uploading files is straightforward – right-clicking on a file brings up a context menu for immediate down/uploading or queuing for later, with the added ability to delete or rename remote files.



Let's face it: light text green on a turquoise field never, EVER works. Ever.

Kasablanca doesn't include a toggle for explicitly choosing ASCII or binary transfers, which could pose a problem for some users; similarly, there are no options to speak of nor is there any documentation of worth.

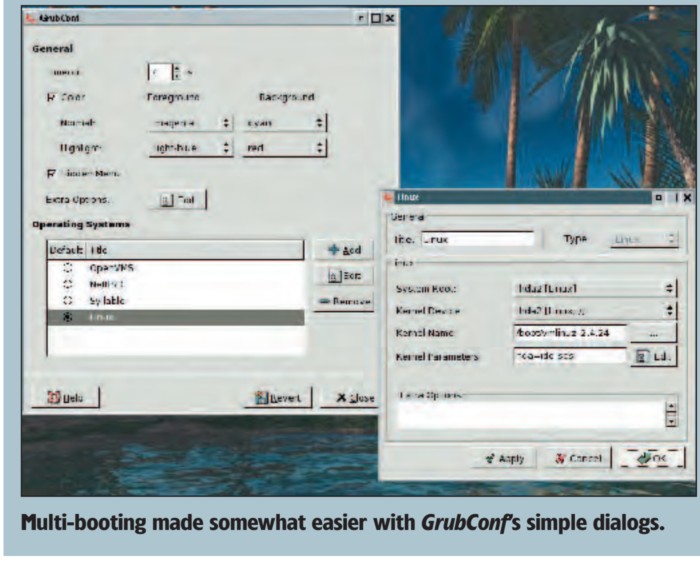
Still, it's early days and *Kasablanca* already has the main features coded in. Undoubtedly it's no rival to GNOME's *über*-slick *gFTP* at present, but for no-nonsense FTP jobs in KDE, it's a fast and frill-free alternative.

BOOTLOADER CONFIG TOOL

GrubConf

■ **VERSION** 0.5.1 ■ **WEB** http://grubconf.sourceforge.net/

In many distributions, *GRUB* has overtaken *LILO* as the default bootloader for Linux – as you can expect, this has lead to heated debate in places (a bootloader flamewar is pretty disturbing). Much of this is down to the wider featureset that *GRUB* offers, although expecting wet-behind-the-ears Linux newcomers to edit the configuration files by hand is a bit much. That's where GrubConf comes in, a small utility which aims to take the hassle out of *GRUB* configuration.



Multi-booting made somewhat easier with *GrubConf*'s simple dialogs.

GrubConf is built on GNOME 2 and only available in source form at the moment (with binary packages planned for a 1.0 release), so the normal **./configure && make && make install** procedure (as root) should install it correctly. During the make install phase you may encounter an error; if so, just copy the src/grubconf binary into /usr/sbin by hand (and if that doesn't work, try the forums on our website).

With the starting window split into two sections, *GrubConf* 's interface provides easy access to the main areas of *GRUB* modification, namely global options and operating system choices. From this window you can assign a splash screen image or change the text-mode colours (a compile time option sets which one of these is available), define the timeout before the default selection is launched, and add extra options in text format.

If you're already using *GRUB*, you should find a list of available OSes on the lower pane; choosing to edit or add to these throws up a detailed dialog box in which you can select the kernel location, root device and so forth. There's an extra window for enabling some kernel options (*eg ide-scsi* for CD-writers on 2.4), and it'd be good to see more of these options in subsequent releases.

GrubConf isn't a spectacular must-have utility, and many distros include their own bootloader configuration tools. Nonetheless, it's simple to use, fits in well with GNOME and is equipped with some fairly good documentation as well. Most experienced Linuxers will prefer to edit the config files by hand – but for novice Open Source convertees, it does the job quickly and pleasantly.

BACKUP SOFTWARE

ESR Backup

■ **VERSION** 0.98b4 ■ **WEB** www.shawnscott.net/esr/

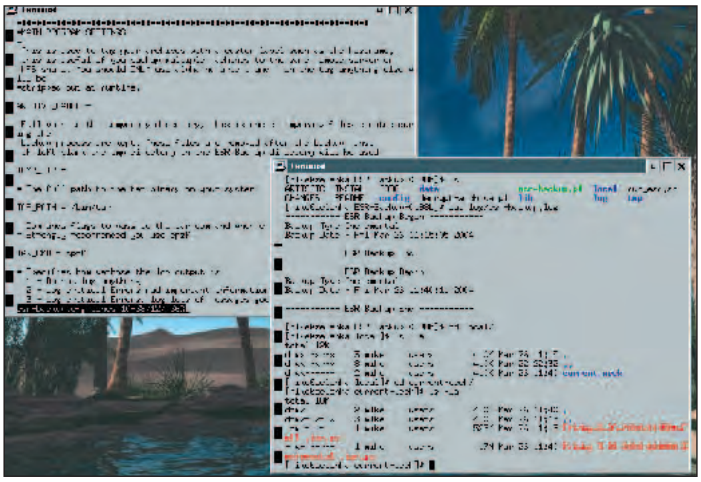
Like eating more healthily or spending more time with the family, making regular backups is one of those things we all genuinely intend to do at some point. In the meantime, most of us wait until a massive hard disc crash before thinking about our backup strategies. *ESR Backup* doesn't make archive copies of outspoken Open Source advocate Eric S Raymond (see *LXF52*), but instead stands for *'Easy Secure Remote'* Backup.

ESR Backup is written in Perl and requires a few additional modules, depending on the features enabled. To back up a /home directory on single-user desktop machines, the tool can be installed anywhere; for multi-user systems it needs to be installed and run as root. The excellent docs detail the configuration file in full, along with creating a crontab entry to run it in the background every day. (In most distros you'll have to change the location given for tar.)

In its supplied configuration, *ESR Backup* will archive the files and directories specified in config/file-list.cfg and, by default, output the results in local/ in gzipped tarballs. Full backups are made at the beginning of the week, while incremental backups (*ie* only files that have been added or changed) are created daily in between. Snapshots are taken of the current and previous weeks – so you always have at least two levels of backup.

That'd be fine on its own, but *ESR Backup*'s nifty features come in the form of automagic FTP uploading and encryption. With the *Net::FTP* Perl module and appropriate settings in the config file, the program will place backups on a specified FTP server – essential if a machine is compromised or the hard drive(s) become corrupted. Security is handled by 448-bit Blowfish encryption, and requires the *Crypt::CBC* and *Crypt::Blowfish* modules.

With security problems and hard drive failures not likely to disappear any time soon, a solid backup system is essential and *ESR Backup* works very well indeed. It's documented clearly, easy to set up and on the whole provides a sensible and manageable backup solution for desktops and small servers.



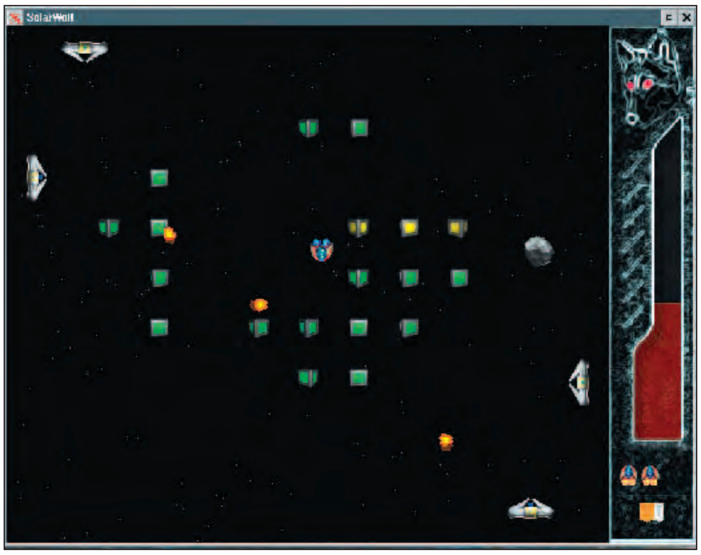
Not much eye candy to show in *ESR* – just the config file and other bits.

SPACE ACTION GAME

SolarWolf

■ **VERSION** 1.5 ■ **WEB** www.pygame.org/shredwheat/solarwolf/

Many early video games were set in space or based on a space theme (*SpaceWar*, *Space Invaders et al*), possibly because a black background with white dots was far easier to recreate than fecund, rolling Terran landscapes. And yet, space has solidly remained a popular setting for modern games, and *LXF* reader Pete Shinners asked us to cover his *SolarWolf* creation in *Hot Picks*.



Complete the level before the red bar drains out, and you'll skip the next.

Installing from source could require some fiddling, as *SolarWolf* is written in Python and requires the *Pygame* library (which in turn relies on various SDL dependencies). Egad. Some distros include *Pygame*, but if you have no luck with that, the source tarballs on our coverdisc should help. The last resort is the single binary file (solarwolf-1.5c.x86.run) – set it as executable, cross your fingers and run it.

SolarWolf's game mechanics are based on those of *SolarFox*, an Atari 2600 title of days gone by (and one of the main coder's childhood favourites). Luckily for us though, blocky graphics and grating monophonic background ditties have been thrown out in favour of snazzy raytraced sprites (generated in *Povray*) and thumping soundtracks. As we're starting to see more often in *Hot Picks* game coverage, hobbyist coders are more than capable of engaging presentation and polished design, rivalling large companies' triple-A efforts.

Split up into single-screen stages, the goal of *SolarWolf* is deceptively simple – collect the coloured boxes littered around the playing area while avoiding enemy fire. This demands accuracy, planning and extreme keyboard dexterity, and the game's depth is improved considerably by the presence of power ups, bonuses and extra hazards to avoid. The craft moves constantly – in a similar fashion to *Pac Man* – and a boost key is available to speed things up.

Despite the simplistic concept, Pete has paid great attention to the player in *SolarWolf*; hint bubbles pop up to describe new items or potential problems, and the collision detection is absolutely spot-on. It's not going to drag you away from work for days on end, then, but it's well coded, lavishly presented and enjoyable for the odd burst of enjoyment here and there.

RACING SIMULATOR

TORCS

■ **VERSION** 1.2.2 ■ **WEB** http://torcs.sourceforge.net/

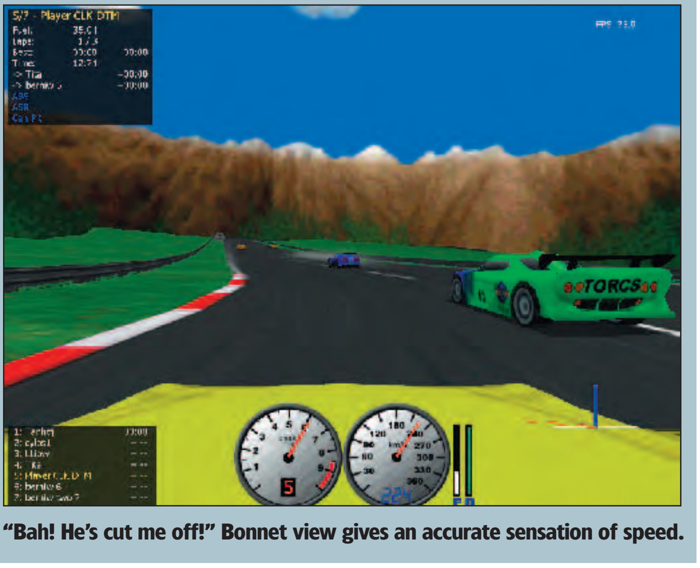
According to a recent survey, 99.98% of the population agreed that driving cars, motorbikes, jetskis, combine harvesters and trams at breakneck speeds is FUN. Hence the massive popularity of Formula 1; millions of people long to scream down a road at 200mph, four inches above the ground, with only a tire wall separating luxury life and spectacular death. For those of us on more limited budgets though, racing romps like *TORCS* provide all the thrills *sans* spills.

The Open Racing Car Simulator is, as the name suggests, obviously a Mah Jong game. Our jocularity about unoriginal titles aside, *TORCS* is one of the best-known Open Source games, although it's a monster to download – along with the base packages you'll need extras for cars, tracks and computer opposition, along with *PLIB* to compile. We've provided all the source, along with the available binaries, on our coverdisc.

Once up and running, it's worth checking out the in-game graphics and display options before you hit the track – *TORCS* is a demanding 3Der and decreasing the visibility depth will improve the frame rate (at the expense of dreaded scenery pop-up). A number of game modes are available, including practice, quick race with opposition and full championship, and (depending on add-ons) there's a decent range of tracks and CPU competitors.

Laudably, *TORCS*' coders have implemented realistic AI in the CPU racers, so you'll occasionally see them spin off or knock into one another. This makes it infinitely more engaging than the *Aero Gauge*-esque 'let's all drive neatly in a line' robot opponents, an affliction that still dogs racing sims today. Course designs vary from intentionally dull (oval) to more inventive and challenging (dirt jumps), while the cars react well – an analogue joystick is best to control your vehicle though.

A few aspects of *TORCS* could be improved, most notably the unconvincing physics. Head-on collisions should be accompanied by a giant smash and flipped cars and flying limbs, not a soft thud and instant deceleration. Big ticks go to the track surface and some scenery, but red crosses for the sky. So it's not quite a *Gran Turismo*-beater yet, but it's entertaining and very playable nevertheless. LXF



"Bah! He's cut me off!" Bonnet view gives an accurate sensation of speed.

STOP SPAM, VIRUSES & HACKERS

On the CD and DVD

Protecting your identity, your data and your resources has never been more important...

cover feature

Security solutions are really about saving time, money and effort. A server that has been maliciously hacked may cause some unpleasantness to your business, not least of which is the effort required to rebuild it and get it back on its feet again. On a seemingly lesser note, unwanted bulk emails may seem to be a nuisance, but they could be robbing you of more than 50 per cent of your storage space, never mind the hassle they cause to the recipients. Even on a single-user level, these external threats can cause catastrophic loss of data, or ultimately even the loss of your online identity.

There are of course solutions, and applying them properly is the key to not making the cure worse than the disease. Over the coming pages we'll be looking at key threats to the modern network or desktop Linux box and outlining the best defence against those who mean you harm.

STOP VIRUSES

Linux and viruses

Let's face it: we Linux users should all admit to feeling at least a *little* bit smug whenever we hear the news that the latest virus is already causing havoc to Windows users worldwide. The popular view is that while Windows is as secure as a tiger cage made out of bacon, Linux and Unixes in general are written in such a way that viruses and worms stand little chance of getting a foothold, never mind actually causing damage to the system.

But how well does that viewpoint stand up to close inspection? Surely a system is only as secure as the people using it are smart? The real answer is that yes, for the most part Linux is indeed safe from viruses thanks to its architectural inheritance; but that doesn't mean you have any cause to lower your guard, particularly when it comes to working on the Internet.

Over the next four pages, we'll be looking at the architectural differences between Linux and Windows that place them light-years apart in security, the areas in which Linux is potentially vulnerable to viruses, and how you can protect yourself against virus threats now and in the future.

The Invulnerable Penguin

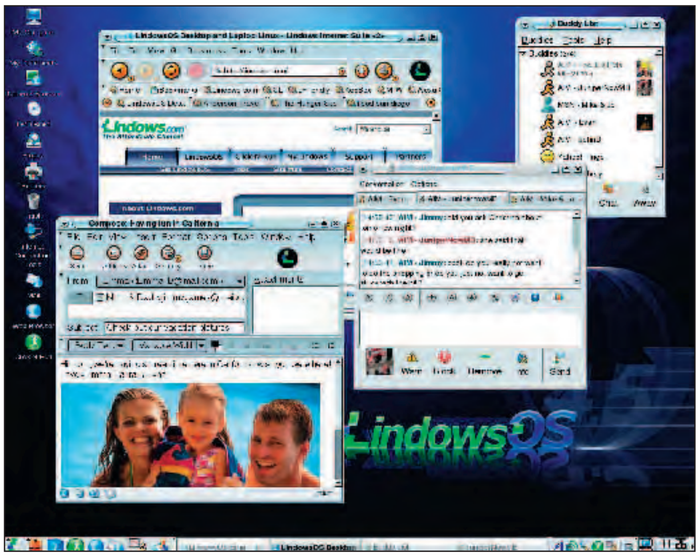
As Unix, the architectural parent of Linux, has been around since 1969, it has had 35 years to mature as a platform. During that time, the various components of it have been worked on by practically all of the major universities in the world, and all but one of the largest IT companies. The odd company there is, of course, Microsoft, where the primary competitor to Unix is produced. Windows is largely based on an architecture created back in 1993, launched as Windows NT 3.1. But in spite of having had some 12 years to mature, the Windows security record is tattered and worn, which leaves many wondering what it is about Linux that makes it quite so secure.

The two primary means of delivery for viruses are social engineering and buggy software. The former, made famous by Kevin Mitnick's book, *The Art of Deception* (Wiley, ISBN 0-471-23712-4), is where users are essentially deceived into launching a program or taking an action that allows a virus onto their computer. Viruses installed through exploiting a bug in software is pretty much out of the hands of end-users, and solution to this relies on distro vendors tracking, fixing and distributing security updates.

Of these two means of delivery, history has shown us that the first is much more common – the leading virus distribution program to date (Microsoft Windows) has seen many viruses that spread by email and encourage users to open executable files. This example of social engineering is essentially cross-platform in that there's no reason Linux users couldn't receive executable mail attachments, and there are dozens of other ways that Linux users could be tricked into running malware – so why *is* Linux such a hostile environment for viruses?

Lack of control

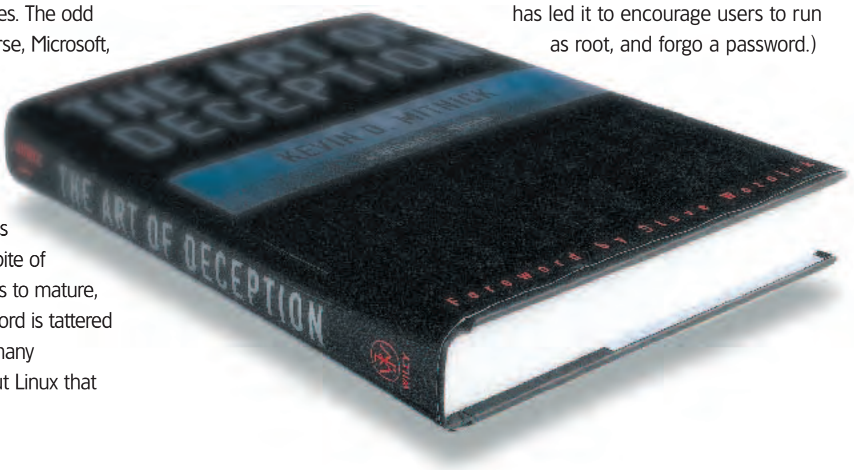
In order to spread, a virus must be able to infect local files with its payload. The most common target for this, by a vast majority, are executable files, because once an executable file is infected it will spread itself whenever a user runs it as part of their day-to-day life. In Windows, even up to Windows XP, users run as 'Administrator' (root) by default, which means they are able to access, edit, and delete any files on the system. On Linux, the opposite is true – people are encouraged to work as an unprivileged user most of the time, and only switch to root to perform specific, system management tasks. (The exception to this is Lindows, where the drive to mimic Microsoft has led it to encourage users to run as root, and forgo a password.)



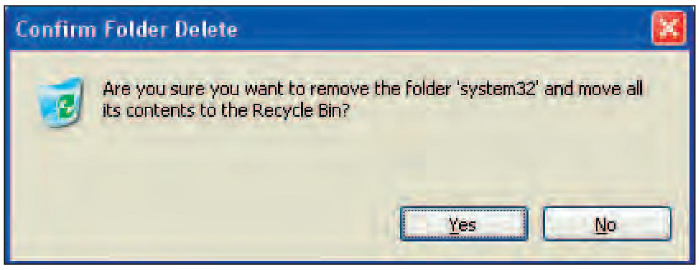
Most Linux distributions encourage you to stay clear of root unless its necessary, with the exception being Lindows – if you're a Lindows user, be very, very careful what you click!

> "To mess up a Linux box, you need to work at it; to mess up your Windows box, you just need to work on it"
> Scott Granneman, SecurityFocus

»



In *The Art of Deception*, Mitnick made it clear that social engineering was easy, scaring thousands of IT managers to at least implement *some* precautions in their businesses.



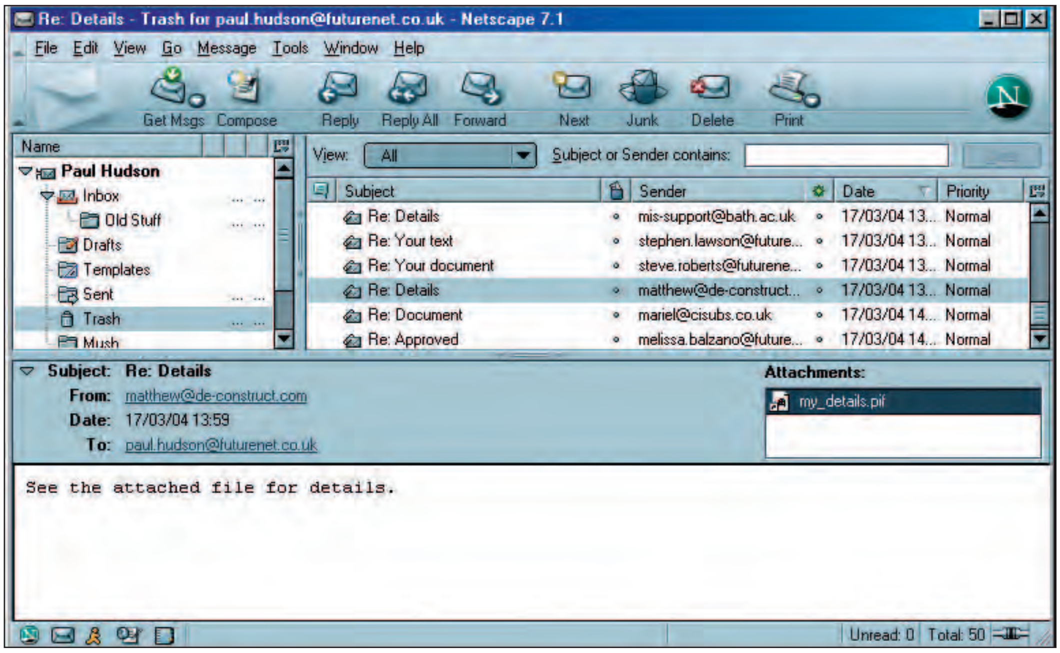
Are you sure you want to remove the folder 'system32' and move all its contents to the Recycle Bin?

Users (and therefore viruses) can cause catastrophic damage in Windows with very little effort.

While running as an unprivileged user, if you were to try executing this command: **rm -rf /** (delete everything on the root partition) you would find that you are denied access to everything but the contents of your home directory. This is because, as standard, Linux only gives you permission to write to files where it is necessary – you can't change files and directories that you don't own, such as **/bin/ls** or **/usr/src/linux**. As a result, if you do receive a virus-infected file it will fail to infect the vast majority of executable files on your system – the worst you can do is infect the files in your home directory.

> **"If you execute an infected file while running as root, you're as susceptible to a virus infection as a Windows user."**

Of course, the exception to this is when you're running as root: if you execute an infected file whilst running as root you are as susceptible to full system virus infection as a Windows user. This is mitigated by the fact that one of the things drilled into users from the very beginning of their switch to Linux is that running as root is a no-no. Typically the reason is given as "so you don't do something you might regret", but an additional benefit of this scheme is that it limits viruses.



Subject	Sender	Date	Priority
Re: Details	mis-support@bath.ac.uk	17/03/04 13...	Normal
Re: Your text	stephen.lawson@future...	17/03/04 13...	Normal
Re: Your document	steve.roberts@futurene...	17/03/04 13...	Normal
Re: Details	matthew@de-construct...	17/03/04 13...	Normal
Re: Document	mariel@cisubs.co.uk	17/03/04 14...	Normal
Re: Approved	melissa.balzano@future...	17/03/04 14...	Normal

Subject: Re: Details
From: matthew@de-construct.com
Date: 17/03/04 13:59
To: paul.hudson@futurenet.co.uk

Attachments: my_details.pif

```
See the attached file for details.
```

Famous last words: *"Ooh, I wonder what a .pif file does..."*

Lack of targets

We have shown that only infected binaries that are run by the root user are damaging to the global system, but what happens when non-privileged users run an infected binary? The answer is that it wreaks havoc inside their home directory, infecting all the binaries it finds. However, to assess how damaging this is, we need to ascertain what kind of user would run an infected file. Consider these three questions:

1. **If someone emailed you a binary file, would you run it?**
2. **If you downloaded a binary file from an untrusted location, would you run it?**
3. **If a person gave you a CD/floppy disk with binaries on, would you run it?**

If you answered "yes" to any of those questions, then you are at risk to virus attack. Having said that, most experienced users get their software from one of four places:

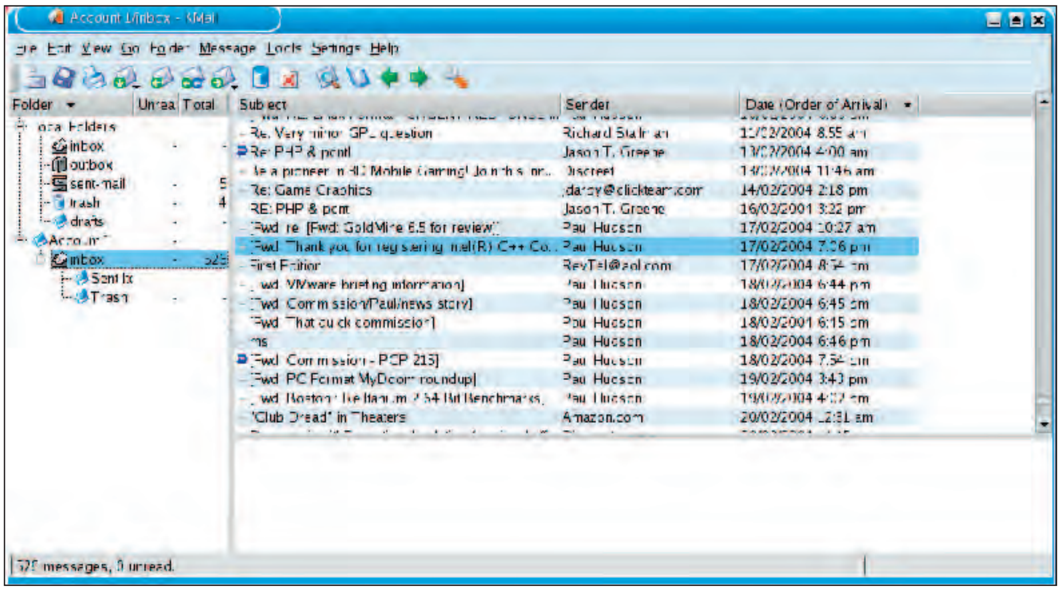
- **By installing it from the CDs/FTP site of their Linux distribution**
- **By using *RPMFind* or similar**
- **By installing it direct from the Linux Format coverdisc**
- **By compiling and installing source code written by a trusted party**

As a result, the majority of people – arguably the only people – who *would* run a binary from an email or a random floppy disk are inexperienced Linux users: newbies. Now, we did a quick scan of our home directories to see what was in there, and the average number of executables found in there was four. Of those four, all were programs that had been compiled from source, installed using **make install**, then forgotten about. These forgotten compiles were not programs being executed, and wouldn't even have been there if we actually cleaned out our home directories once in a while.

Now, if *Team LXF* has just a handful of binaries in their home directories, despite the gargantuan amount of software we try out each and every month, how many do you think the average newbie is going to have? None, that's how many. Chances are they have a few documents, lots of settings, some game saves, maybe some pictures, and very little else. You see, perhaps the defining thing about newbies – the *only* type of user likely to run dodgy binaries, right? – is that they rarely (if ever) stray outside the confines of the programs they are provided with, and almost never have their own programs in their home directory. So, even if they run a virus-infected file, the most it will do is scan their home directory for targets, fail to find any, and die silently.

Lack of local path

Although we have shown that the group of users most likely to run virus-infected files aren't actually likely to have any binaries in their home directory that can serve as a host for the virus to spread, it's still possible that someone might receive a virus file and accidentally run it. For example, getting a user to run an executable file by accident would easy if a malicious users supplied an



Subject	Sender	Date (Order of Arrival)
Re: Very minor GPL question	Richard Stallman	12/02/2004 8:55 am
Re: PHP & pcntl	Jason T. Greene	13/02/2004 4:00 am
Be a pioneer in 3D Mobile Gaming! Join this ...	Discreet	13/02/2004 11:46 am
Re: Game Graphics	darcy@clickteam.com	14/02/2004 2:18 pm
RE: PHP & pcntl	Jason T. Greene	16/02/2004 3:22 pm
[Fwd: re: [Fwd: GoldMine 6.5 for review]]	Paul Hudson	17/02/2004 10:27 am
[Fwd: Thank you for registering Intel(R) C++ Co...	Paul Hudson	17/02/2004 7:26 pm
First Edition	RevTel@aol.com	17/02/2004 8:34 pm
[Fwd: VMware briefing information]	Paul Hudson	18/02/2004 6:44 pm
[Fwd: Commission/Paul/news story]	Paul Hudson	18/02/2004 6:45 pm
[Fwd: That quick commission]	Paul Hudson	18/02/2004 6:15 pm
ms	Paul Hudson	18/02/2004 6:46 pm
[Fwd: Commission - PCP 213]	Paul Hudson	18/02/2004 7:54 pm
[Fwd: PC Format MyDoom roundup]	Paul Hudson	19/02/2004 3:43 pm
[Fwd: Boston: The Itanium 2 64 Bit Benchmarks]	Paul Hudson	19/02/2004 4:22 pm
"Club Dread" in Theaters	Amazon.com	20/02/2004 12:31 am

525 messages, 0 unread.

Although it's popular, *KMail* is just one of many mail clients that are used on Linux.

infected binary called *ls* inside a tarball containing an otherwise innocent set of files. Then, when a user extracted the tarball, they would **cd** into the directory containing the files, type **ls** to get a directory listing, and BANG! The virus would run and the home directory would be infected – or would it?

If the above scenario were possible, then Linux wouldn't be nearly as secure as people have claimed in the past. The good news is that the scenario is impossible, because Linux never places the local directory in the path for binaries to be run – typing **ls** would run /bin/ls – even if there is an *ls* in the current directory. To run a local *ls*, you'd need to be explicit by typing **./ls**, which nullifies any chance that someone smuggled a program in without the user noticing.

Lack of auto-execute

Two elements of the Windows operating system that are distinctly conducive to virus attacks are that:

a any file ending with .exe, .com, .bat, .pif, .vbs, or .scr is considered executable; and
b by default, placing a CD or DVD in the drive can launch one or more programs without asking you for permission.

These features are ostensibly for ease-of-use, but they also do a lot for ease of virus-spreading. If a user receives a binary file by email and they open the attachment on a Linux computer, it will treat it as if it is text and prompt for the program to use to read the file. On a Windows operating system, the executable file will just run, because Windows regards the file extension as sufficient reason enough to execute it as opposed to opening it with another program.

The direct result of this is that a user who receives a virus-infected file on Linux needs to save it locally, notify the system that the file is executable, then run it. While it's still possible to run the virus, Linux triples the number of steps necessary to run it, and the second step has the user explicitly say, *"I want this file to be executed"*. Again, in Windows operating systems, the default setting is *"Hide extensions for known file types"*, which makes foo.txt.exe look

»

IGNORANCE IS BLISS

A short history of Linux viruses

Although the first real Linux virus was called Staog, it was never actually found in the wild back when it was discovered in mid-1996: the first virus that was actually noticed was called Bliss. Bliss itself was first posted to Usenet on alt.comp.virus, comp.os.linux.misc, and comp.security.unix on September 29th 1996 and infected every binary it could write to with itself.

On February 5 of the following year, with Bliss having faded away from the community consciousness practically unnoticed, McAfee issued a press release that contained, amongst other things, the following piece of text:

SANTA CLARA, CALIF. (February 5, 1997) – McAfee (Nasdaq: MCAF), the world's leading vendor of anti-virus software, today announced that its virus researchers have discovered the first computer virus capable of infecting the Linux operating system...

The virus, which is called Bliss, is significant because many in the Unix industry have previously believed that viruses were not a concern to Unix operating system users... "Bliss is a destructive virus which overwrites Linux executables with its own code," said Jimmy Kuo, McAfee's director of anti-virus research.

"Although several incidents of Bliss infection have already been reported, the virus is not currently widespread. We encourage concerned Linux users to download a free working evaluation copy of our VirusScan for LINUX, which can be used to detect the virus."

If we were to believe McAfee, Bliss would have been the first Linux virus, it would have been discovered by them irrespective of it being posted online to security newsgroups four months earlier, it would have caught the Unix industry wholly unaware – who would have thought that running untrusted code might damage your system? – and we'd need to be using McAfee's software to save ourselves.

As McAfee's press release hit headlines around the world, the author of Bliss stepped forward and produced a newer version with various bugs fixed, a description of how it works, and also what tests he'd done on it. Most damningly, he also told the world that:

a Bliss is not intentionally destructive.
b he compiled Bliss with debugging code enabled so that others could see how it worked.
c he wrote it as a proof-of-concept.

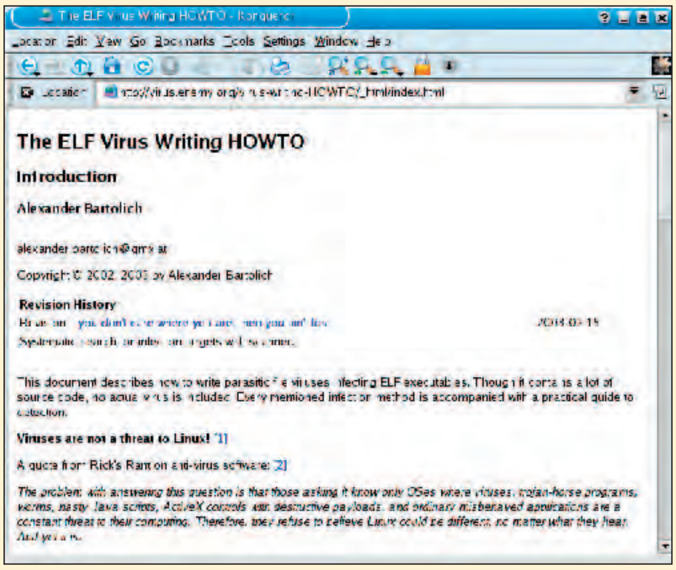
If all those points combined didn't make it completely plain to McAfee's expert team that Bliss wasn't at all threatening, the virus also kept a detailed log of its actions in /tmp/.bliss, and also had a friendly --bliss-uninfect-files-please option that efficiently does exactly what its name suggests.

Despite both Bliss and Staog being proof-of-concept and confined to the laboratory, when one reporter recently listed information on infamous Linux viruses in the wild, they formed two of the six cited, with another one (Lindose) also being proof-of-concept. If this is the worst of the worst, let's hope for McAfee's sake that it has an alternative business plan...

THE VIRUS-WRITING HOWTO...

Linux is all about learning

The Open Source community being how it is, it should come as no surprise to hear that there's a HOWTO document already written on Linux viruses that explains how to write programs that hijack the ELF format Linux uses for its binaries. The document isn't malicious at all, however – you cannot use it to actually produce a virus, and the bulk of the material is a detailed discussion on file formats. Check it out for yourself at http://virus.enemy.org/virus-writing-HOWTO/_html/index.html


The ELF Virus Writing HOWTO - Konqueror
Location Edit View Go Bookmarks Tools Settings Window Help
Location
http://virus.enemy.org/virus-writing-HOWTO/_html/index.html

The ELF Virus Writing HOWTO

Introduction

Alexander Bartolich

alexander.bartolich@gmx.at

Copyright © 2002, 2003 by Alexander Bartolich

Revision History

[illegible] 2003-02-15

[illegible]

This document describes how to write parasitic file viruses infecting ELF executables. Though it contains a lot of source code, no actual virus is included. Every mentioned infection method is accompanied with a practical guide to detection.

Viruses are not a threat to Linux! [1]

A quote from Rick's Rant on anti-virus software: [2]

The problem with answering this question is that those asking it know only OSes where viruses, trojan-horse programs, worms, nasty Java scripts, ActiveX controls with destructive payloads, and ordinary misbehaved applications are a constant threat to their computing. Therefore, they refuse to believe Linux could be different, no matter what they hear. And yet it is.

Virus-writing HOWTO: not for the faint-hearted, and thankfully benign.

like foo.txt – all an attacker has to do is give the executable file the same icon as *Notepad*, and many users would be fooled into double-clicking it thinking they were just opening a file. By requiring users to explicitly require a file be executable, this type of attack is foiled without recourse to system-hogging monitoring software.

Lack of monoculture

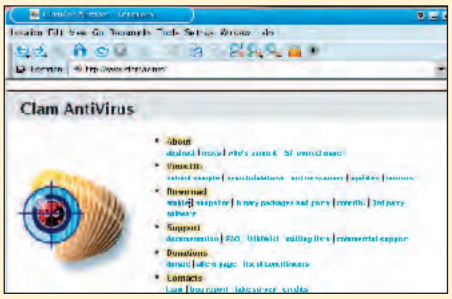
A big argument used against Linux by its detractors is that a given distribution is likely to come with at least two office suites, three web browsers, four media players, and five text editors – that users get confused by it all. Our reply of *"Linux is all about choice, people don't have to install or use anything they don't want to"* is usually enough to muffle the critics' cries, but here's an argument that will silence them outright: ***having more than one choice for everything gives Linux strength in diversity.***

Consider a world where the vast majority of users used *KMail* as their mail client – what if a serious flaw was discovered where just opening an email could allow a malicious user to run a program? That would be a catastrophe for millions of users, because everyone would be unprotected until a patch was produced. In the world of Windows, this is sadly a reality – at least six times in the last five years (with the most recent being in March 2004) users of *Microsoft Outlook* and often *Outlook Express* have been exposed to the possibility of attackers running arbitrary code on their system just through reading an email.

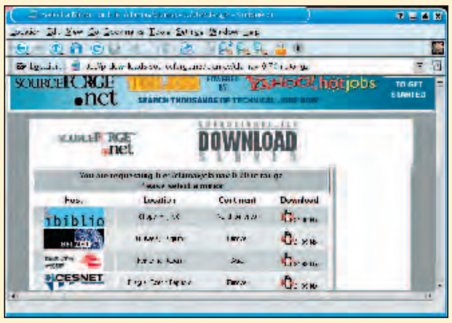
As these two programs hold almost 100 per cent of the market on Windows, virus writers have an easy job: target *Outlook*, and you hit most of the world. In Linux, the market is much more fragmented: some like *KMail*, a few use *Evolution*, others prefer *Mozilla Mail/Thunderbird*, and still others couldn't survive without *Pine*. What, then, does a virus writer target? Unless there is a fundamental flaw in a library that all four programs use, a virus that takes advantage of a hole in one will fall flat on its face in the other three and the however many other mail clients.

INSTALLING CLAM ANTIVIRUS STEP-BY-STEP

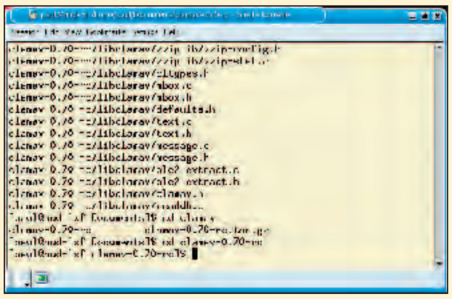
And it won't cost you a single clam...



1 Go to the ClamAV homepage at www.clamav.net and have a quick look over the latest announcements to make sure everything is good to go. There is a version of *ClamAV* on the *LXF* coverdiscs this month, but it may not be the most recent iteration.



2 If the version that we've supplied on the *LXF* coverdiscs is outdated, download the latest stable build in .tar.gz format. You may also want to grab the *MD5* sums to make sure you have a trusted build.



```
clamav-0.70-rc/libclamav/zziplib/zzip-config.h
clamav-0.70-rc/libclamav/zziplib/zzip-stdint.h
clamav-0.70-rc/libclamav/cltypes.h
clamav-0.70-rc/libclamav/mbox.c
clamav-0.70-rc/libclamav/mbox.h
clamav-0.70-rc/libclamav/defaults.h
clamav-0.70-rc/libclamav/text.c
clamav-0.70-rc/libclamav/text.h
clamav-0.70-rc/libclamav/message.c
clamav-0.70-rc/libclamav/message.h
clamav-0.70-rc/libclamav/ole2_extract.c
clamav-0.70-rc/libclamav/ole2_extract.h
clamav-0.70-rc/libclamav/clamav.h
clamav-0.70-rc/libclamav/[illegible]
[illegible] Documents]$ ls clamav
clamav-0.70-rc            clamav-0.70-rc.tar.gz
[illegible] Documents]$ cd clamav-0.70-rc
[illegible] clamav-0.70-rc]$
```

3 Open a console, and execute tar xvzfp <filename> to extract to a new directory. Change to root, and type adduser clamav then passwd clamav and give the new user a password. Finally, go through the familiar ./configure && make && make install process.

CROSS-PLATFORM VIRUSES

The Lindose virus – sometimes known as Winux – is a proof-of-concept virus that infects both Windows and Linux systems. As with most other viruses, Lindose replicates itself to all binary files to which it has write access, and does the same on Windows. As technologically advanced as this might seem on first description, scratching the surface tells a different story. Lindose actually didn't work on Linux systems when it first arrived into the system, as it arrives as a Windows executable. When run for the first time, it infects all the Windows and Linux executables it finds, and, if the user reboots into Linux, the Linux executables may *then* spread the virus.

So, while it can infect Linux machines, it does rely on you having Windows installed and having run the virus therein. This is a weak attack vector, which explains why Lindose failed to make it into the wild.

Are you safe?

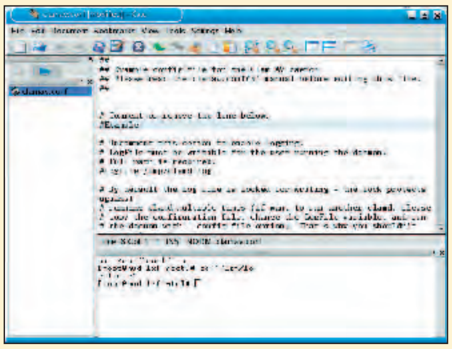
Hopefully by now you'll agree that getting a virus onto a Linux system usually takes some work, and so for the large part we're quite immune to the malware threat. Is that going to change? Well,do one-legged ducks swim in a circle? As the number of Linux users increases, so the number of viruses written for the platform will increase. There is a theory that the reason Windows has so many viruses written for it is because of its popularity, but that's not what we're saying. Instead, as more and more people come to Linux, the chance of social engineering tactics reaching people who don't know better will go up. Let's face it, there are millions of Windows users who open files called ILOVEYOU.exe even though they have read warning after warning against doing so – these same people are eventually going to be using Linux, and we had best do what we can to prepare now – while we can.

Even though we are relatively safe right now, and as long as you continue to be smart you're likely to be safe forever, there really is no excuse for not running a virus scanner on your system. Don't think you're safe just because you get all your binaries from your distribution's FTP site or because you compile everything from source – in the past year, both the GNOME and GNU sites have been hacked, and the fact that no damage seems to have been done is probably down to good luck more than anything else.

Free solutions – such as *ClamAV*, introduced below – are able to provide you the protection you need to keep your system safe, and do so at such negligible cost of system resources that you won't even notice it's there. The number of viruses on Linux is never likely to equal the current level seen on Windows, but that's no reason to leave yourself open. It's important to keep in mind that an elementary virus might delete all the files in your home directory, which is arguably worse than one that infects binaries and is run as root – you can re-install the operating system following the event of infection, but documents, once deleted by a virus, are lost forever.

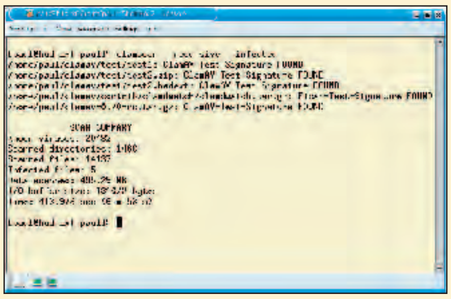
> **"As long as you continue to be smart, you're likely to be safe forever but there's no excuse not to run a virus scanner."**

»



```
##
## Example config file for the Clam AV daemon
## Please read the clamav.conf(5) manual before editing this file.
##


# Comment or remove the line below.
#Example

# Uncomment this option to enable logging.
# LogFile must be writable for the user running the daemon.
# Full path is required.
#LogFile /tmp/clamd.log

# By default the log file is locked for writing - the lock protects against
# running clamd multiple times (if want to run another clamd, please
# copy the configuration file, change the LogFile variable, and run
# the daemon with --config-file option). That's why you shouldn't
```

4 Edit /usr/local/etc/clamav.conf and add a # before the word Example. Without doing this you will not be able to run a virus scan.



```
[localhost ~]$ clamscan --recursive --infected
/home/paul/clamav/test/test1: ClamAV Test Signature FOUND
/home/paul/clamav/test/test2.zip: ClamAV Test Signature FOUND
/home/paul/clamav/test/test2.badext: ClamAV Test Signature FOUND
[illegible] ClamAV-Test-Signature FOUND
/home/paul/clamav-0.70-rc.tar.gz: ClamAV-Test-Signature FOUND

        SCAN SUMMARY
Known viruses: 20782
Scanned directories: 1480
Scanned files: 14137
Infected files: 5
Data scanned: 495.25 MB
I/O buffer size: 131072 bytes
Time: 413.974 sec (6 m 53 s)

[localhost ~]$
```

5 Change to the directory you want to scan, and type clamscan --recursive --infected to do a recursive directory scan only printing out information when infected files are found.

TOMASZ KOJM

On why you should use protection...

"Don't forget we're software developers, not businessmen," is how *ClamAV* founder Tomasz Kojm has described the role ClamAV in the past; yet his anti-virus application is now the leading Linux/Unix anti-virus solution. Is this just modesty, or social engineering in action? *ClamAV*'s Open Source approach provides the kind of competitive advantages for cost-conscious companies and institutions that we highlight in this month's *Linux Pro* cover feature.

LINUX FORMAT: ***Would you say that most Linux system administrators underestimate the virus potential?***
TOMASZ KOJM: The situation is changing for the better, and every day more and more administrators decide to add a virus protection to their systems. For many years, there was no really free anti-virus scanner, and most people were convinced they have to spend a large amount of money for a commercial license. It's now possible to implement a quite effective anti-virus protection for server without spending a penny and admins just start taking advantage of it.

LXF: ***Apart from issues concerning the different numbers of users, why would you say that Linux is more safe from viruses than Windows? What are the challenges that face virus writers targeting Linux?***
TK: Virus writers make use of the fact that most users of Windows are accustomed to work at a highly privileged level that allows to manipulate and modify all critical files in the system. In Linux, it is a fundamental principle that one should only use the root account for administrative actions and never for a regular work. As long as this rule is obeyed, viruses can't do any harm globally.

Another big difficulty is a number of Linux distributions and supported architectures, and natural differences between them. It's nothing unusual that two files at the same location are binary-incompatible on two different Linux boxes.

The diversity of Linux also makes it more resistant to Internet worms. Self-distributing worms take advantage of holes in popular network software. In most cases, it's not possible to create a generic exploit for a vulnerable service in Linux. It's caused by a fact that the same software may have been compiled with different options or even work on completely different architectures. That's why in contrast to Windows (which usually uses the same precompiled software) creation of a Linux worm is a complicated task.

LXF: ***Do you believe that the number of viruses targeting Linux will increase in relation to the ever-increasing number of Linux users?***
TK: It's not a big challenge to write a virus. However as I already mentioned, Linux viruses have slight chances for effective distribution. In my opinion the real problem is backdoors. Sometimes, popular sites fall victim to crackers, who place modified software packages that later can allow them to unnoticeably log into a host running them. That's why it's very important to verify a digital signature – or at least an *MD5* checksum of a package – *before* installing it. Extremely dangerous are kernel-level backdoors. New versions no longer require loadable module support, because they can be loaded via /dev/kmem. In contrast to classic viruses, they cannot be easily detected by anti-virus software. Having full control over a system and many possibilities to hide itself, a kernel rootkit can stay unnoticed for many months.

"In Linux, it is a fundamental principle that one should only use the root account for administration, and never regular work."

LXF: ***What is the primary method of delivery for Linux viruses? How important is social engineering?***
TK: If one wants to infect a Linux system with a virus or trojan horse, one must be able to gain the required privileges or coax a system administrator into running an infected file. On the other hand, if the sysadmin is irresponsible enough to run untrusted software, there's no way (and no tool) to prevent them from making some other kinds of system destruction. Because Linux users share mainly source and not binary files, viruses have a very small chance to self-distribute effectively. Even though viruses exist that can infect ELF executable files, if you catch a Linux virus, it's more than likely it was delivered to you with a premeditated intent, rather than just random infection.

LXF: ***Would you say that distributions such as Lindows – where users are always root – are specific virus targets?***
TK: Regular work with the superuser privileges is a terribly bad habit. Systems like Lindows (which doesn't create a normal user account during installation) and BeOS (single-user) follow the pitiful access model of Windows 9x. That opens a wide gate for malware, because it may modify or infect every file in a system and should be definitely avoided.

LXF: ***Would you say that cross-platform viruses are a possibility in the future?***
TK: Firstly, cross-platform viruses (such as Java.StrangeBrew) have appeared already, but they are mostly a proof-of-concept and not especially dangerous. Fortunately, both Java and .NET/Mono contain built-in protections that make virus distribution in network environment far more difficult (*eg* very popular Java applets cannot modify any files). As long as *OpenOffice.org* doesn't decide to support VBA macros at some point in the future, we can sleep peacefully!

LXF: ***Would you say that all Linux users need a virus scanner?***
TK: No, I wouldn't. If you only use your user account on a Linux desktop, there's no big reason to stress the system with a virus scanner (especially with an on-access one). However, a scanner doesn't hurt and it may prove quite useful at least in mail filtering (if a provider does not do that already) because recent worms are almost as irritating as spam. Of course if a user has to deal with Windows executables, a virus scanner is advisable.

LXF: ***Can an Open Source scanner compete effectively against a commercial system?***
TK: ClamAV is still a relatively young project, but its development is very rapid – new features are introduced every few days. It still lacks some advanced scanning methods (*eg* heuristics) which can be found in commercial scanners, but its strong point is a very good reaction time on new outbreaks, and its advanced virus database distribution system. Easy installation and administration makes *ClamAV* at least a good addition to a commercial scanner.

STOP SPAM

What UBE you be?

While not actually often a security threat in the purest sense of the word, spam probably causes more lost working hours worldwide than every virus attack and system outage there has ever been. Depending on who you ask, around 50-60 per cent of the mail now being delivered to Inboxes around the globe is unwanted rubbish, urging the recipient to stock up on various performance or body-enhancing drugs, get a 'professional' logo for their website or participate in various virtual or physical activities. More insidiously, while viruses can be very dangerous, they often don't have any co-ordinated effort behind them. Unsolicited bulk email on the other hand, is big business, with some of the companies involved clearing profits of over £1 million a month

The virtual equivalent of the junk mail that gets stuffed through your letterbox every day is more than just a nuisance, it's an epidemic. With companies laying on extra storage and power to mail servers just to deal with the increased volume of mail, its possible the most evil form of marketing ever, causing the recipients to bear the cost of its transmission. While the spammers get even richer.

Unfortunately, there is as yet no magic bullet to rid the world of spam. Software you may have seen advertised does the job of clearing up the aftermath, but incredibly, the amount of junk mail troubling the internet, clogging up servers and making your Inbox take twice as long to download is actually on the increase. So what can we do about it? There are three main ways of fighting spam:

1 Leave it to experts

One popular option for corporates is to outsource the problem to someone else. For a lot of organisations this makes sense. For a small fee, an external company will act as a relay for your mail, marking up spam as it sees it, and either delivering it suitably marked or just deleting it.

There are several advantages to this approach. For a start, such companies handling large volumes of mail for their clients, are more exposed to all the wonderful, rich varieties of spam in the first place. Therefore, their highly tuned Bayesian engines, and whatever other tricks they might use (such as tracking open relays, maintaining blacklists etc) should be a lot more effective.

There are several companies offering such services, probably the best known of which is Messagelabs (**www.messagelabs.co.uk**), who will tailor the service to meet your needs.

2 MTA filtering

Once inside your physical network, the first line of defence against spam is the MTA. It is the responsibility of this mailserver to transport messages either to individual mailboxes, or through IMAP or POP accounts to remote email clients. The MTA is the postman of the mail network and is the best place to start the fight back.

Also, note that just because your ISP handles your mail, doesn't mean you can't set up your own local mailserver too. They can fairly easily be augmented or configured to fetch mail from a POP3 address somewhere else and deliver it to local users. This means that as well as whatever anti-spam capabilities your ISP has installed, you can include your own as well.

One of the simplest steps that can cut down on a huge quantity of spam is quite simply to reject mail that comes from known spammers. *The Mail Abuse Prevention System*, amongst others, is a real-time blacklist of addresses (server addresses that is) where spam is known to come from. This can be because they are unprotected mailservers with an open relay being exploited by spammers, or servers or ISPs that spammers are using themselves. Either way, once on the blacklist, the mail will be refused by the MTA. There are plenty of blacklist services available.

To enable the standard mail-abuse blacklists on a sendmail server, first of all, make sure you are using the latest version of (8.12 is the basis for these instructions). Make sure you have also installed the *M4* configuration package for *sendmail*, as it makes configuration a lot easier.

Edit the file sendmail.mc - this may be in various locations (/usr/share/sendmail-cf/, /etc/mail/) depending on your distribution.

About 40 lines into the file, you'll find a group of lines that are commented out (they begin with **dnl**). Uncommenting these lines will enable

»

SPIM

No, it's not a misprint. Spim is a form of spam that has found its way into the world of Instant Messaging. Spim isn't as widespread or as resource-wasting as spam – yet – but it's disruptive enough to result in some instant messaging solutions becoming a lot less useful.

More insidiously, spim steps into a grey area between spam and viruses. The recent 'Osama Found' spim was able to fool hapless `AOL users into downloading extra software, and then propagated itself via the 'buddy-list'. So it's not only an unwelcome time-waster, it can be dangerous.

SPAM CONS

Phishing is the new ex-Nigerian leader's widow. There is a seriously criminal side to some spam. Such ludicrous ruses as the old so-called 419 scam: *"I need you to clear funds through Europe"* may be too well known to work now, but the email fraudsters have moved on. Recently in the UK there was a spate of fraudulent mails apparently coming from any of the well-known banks. The mails told people to visit a website to update their details. Here the technique varied. Some tried a website with a URL very subtly different from the real one, with a very real-looking front page. Others actually opened the real page, but piggy-backed their own look-alike as a pop-up. Enter your sensitive information and the fraudsters now have your bank account.

While people may be suspicious of giving out sensitive information (and at this point it may be worth reminding some that your legitimate bank will NEVER send you an email asking you to resupply all your details) for their bank account, they might be less suspicious of other accounts. The same trick was recently tried with eBay accounts. The direct level of information gathered may not have been as useful, but still valuable to the online identity thief.

> "High on the list of priorities is putting spammers behind bars – most of their activities are illegal in many parts..."

SPAM TIPS

The easiest way to deal with spam is to avoid getting any in the first place. This is easier said than done, but here are a few tips:

- **DO set up an additional email address for accessing mailing lists. These are frequently harvested by spammers.**
- **DO obscure your email address if you post it online.**
- **DO think twice about configuring a bounce-back message. This rarely gets you removed from lists anymore, and the reply-to headers are usually fake anyway.**
- **DON'T open any obvious spam, particularly if you have HTML mail enabled. Often the HTML includes some link code that passes a token to a remote server, verifying that the mail was read.**
- **DON'T bother with the Unsubscribe button some spam includes – use it, and the spammers will then know that your email address really exists and obligingly send you even more spam.**

SPAM JARGON

Bayesian – statistical analysis method often used for identifying patterns in text, named after Rev. Thomas Bayes.

MTA – Mail Transport Agent, or mailserver – the software that actually routes and sends mail packets across the Internet to its destination. Common Unix/Linux MTAs include *Sendmail, Exim, Postfix* and *qmail*.

Phishing – the practice of sending junk mail disguised to appear to come from another individual or company, particularly as a basis for fraud

Spam – commonly used term denoting unwanted bulk emails.

Blacklists – lists of addresses or domains which are suspected of transmitting or relaying spam.

UBE – Unsolicited Bulk Email, a more precise definition of spam. Also sometimes called Unsolicited Commercial Email (UCE).

the RBL lists from mail-abuse. To add a further list, *eg* the spamhaus one, add the following directive:

```
FEATURE(dnsbl, `relays.mail-
abuse.org', `Open spam relay - see
http://www.mail-abuse.org/rss/')dnl
FEATURE(dnsbl, `sbl.spamhaus.org',
`Spamhaus blackhole list
www.spamhaus.org')dnl
```

Then it is necessary to regenerate the config file and restart *sendmail*:

```
m4  sendmail.mc > sendmail.cf
cp sendmail.cf /etc/mail/
/etc/init.d/sendmail restart
```

If you have any errors here, it's likely that **m4** can't find the relevant configurations. Try using *Webmin* to edit the file if you have it installed, as it will automatically locate the relevant files.

Blocking known offenders is all very well, but in reality, spammers tend to move ISPs often, or find new open relays, or other methods to hijack legitimate mailservers. Blacklists can be part of the solution, but if you rely on them alone, you'll still get plenty of spam. Still at the MTA level, we can now start looking at the mail itself to find out whether it is worth transmitting.

The most widely used Open Source app for this is *SpamAssassin* – it uses a variety of techniques to identify unwanted mail. These include heuristic analyses as well as header analysis, blacklists and information from the Vipul's Razor spam database. Written in Perl, it can be easily integrated with a variety of MTAs, or as a standalone linked into *procmail* or another client-side mailreader. To continue our example of *SpamAssassin* using *Sendmail*, probably the easiest way to do this is by setting up a 'procmail recipe'. The procmail files are used for postprocessing user mail accounts, and is an easy place to insert some spam filtering. To do this we need to create or edit the file /etc/procmail.rc.

This is the global file for all users of the system.:

```
#filter through SpamAssassin
:0 fw
* <250000
| /usr/bin/spamassassin
```

There are a number of disadvantages to this: every time mail is received, the *SpamAssassin* program is run. In the case of a busy server, it would be better to run *SpamAssassin* as a daemon and call the client to check files. If you run *spamd* as a service on startup, the last line can be changed to:

```
| /usr/bin/spamc -f
```

This still means that the mail is actually going through the MTA before being identified as spam. There are various solutions for integrating SpamAssassin into Sendmail, and other MTAs. The best source for this information is the SpamAssassin wiki, http://wiki.apache.org/spamassassin/.

3 Client Filtering

Filtering on the client can actually often be the most effective because the client email address is usually harvested or obtained and started off in a specific groups of spam lists – *eg* "Professional Logo design". This means the majority of the spam an individual receives is often related and follows a particular pattern. This means it is also easier to write filters for.

Higher-level filtering tools are almost always set too leniently, simply because it is often worse to remove a few genuine mails than to let through a few dozen bits of spam. The client-side filter can thus act as a general cleanup, and at the very least give some control over the reaction to spam back to the individuals affected. Some MTAs are merely set to mark the spam, not actually remove it: in which case, your mail client will need to filter out the junk itself.

Mozilla (and its spin-offs), *Evolution* and *Opera* all have mature anti-spam capabilities. Usually just turning on the spam filtering will rid you of about half of unwanted messages. They all use a recognition engine that tries to identify future junk based on the sort of mail you have specifically marked. Other mail clients may have the ability to use external filters. The new version of *KMail* (with KDE 3.2) for example, has a Spam wizard, that can be set up to use *Bogofilter* or *SpamAssassin* as a filter. A quick and simple tool for filtering like this is *qsf*. The software is on your coverdiscs and at **www.ivarch.com.**

The *qfs* package can be taught to learn, which at least makes it more likely to pick up spam. It works on *mbox* folders and so can be used by almost any mail client. There is an excellent tutorial on setting up the necessary filters at **www.softwaredesign.co.uk/Information.SpamFilters.html**

Spam futures

While all these methods are useful, a dribble of spam will *still* reach you, no matter how well you configure your filters. Think of all the resources being wasted trying to remove the spam – a better solution needs to be found.

High on the list of priorities is to start putting spammers behind bars. Most of their activities are illegal in many parts of the world, though typical of any sort of litigation, the legal cases will drag on at glacial pace. But more fundamental action must be taken if email – a cornerstone of the e-revolution – is to remain useful. Of course, there are as many 'best ways' of doing this as users to ask, but it seems likely that some form of authentication protocol will be at the root of it. Various systems are being trialled that reject mail where the headers and network origin do not match, but it will be a long time before this problem is solved satisfactorily.

WHO ARE THE SPAMMERS?

Over 90 per cent of the world's UBE problem is believed to rest with just 200 individuals and organisations. These are predominantly 'marketing' agencies or companies interested in promoting more unusual products or services. A fairly detailed list is maintained by 'The Spamhaus Project' (www.spamhas.org). Their 'Register of Known Spam Organisations' (ROSKO) list only contains those who have been denied service by a minimum of three different ISPs for AUP violations. 160 of those currently on the list are based in the USA, while the largest number of the remainder call Canada home.

STOP HACKERS: FIREWALLS

Hacker or Cracker?

Whether you're a home user who uses the Internet for surfing and email, or a corporation with an Internet presence for e-commerce, business, or accounting – ensuring that information, documents and access to systems is controlled remains a priority. While the Linux kernel supports firewalling natively, unless you're a seasoned network administrator or someone with plenty of time on their hands to figure out how it all works, it is usually not the most practical way to secure your network.

We're going to examine the firewall options available for both homes and businesses, and the varying merits of a hardware-based firewall solution versus one distributed only as software. Of course, not all options are ideal for everyone's circumstances or needs, and each option presents its own pitfalls and advantages.

Hardware vs Software

Firewalls come in two flavours. Software firewalls can be installed on existing systems, either in addition to current software or a nice new clean install of a Linux-based distribution. Alternatively, we can obtain a hardware firewall, which is really just an embedded device running software similar to a software firewall. Reality dictates that there is no functional difference between a hardware and software firewall. A hardware firewall is no more or less secure than its software counterpart loaded onto an old Pentium box; and certainly the idea that a hardware firewall is somehow a 'deeper' process that can't fail is something of a crazy idea. Independent of the hardware – whether it be a Cisco router, an embedded Linux system from Linksys, or another embedded operating system from any other firewall manufacturer – there still has to be software running on it to route the traffic, perform filtering and NAT, or simply to do the basic stuff like let you configure it. Any of these software elements could have a flaw opening the device up to exploitation or attack.

Of course, one advantage of a hardware-based firewall is that a lazy administrator can't go and install a slew of unneeded software on the device, which is obviously a major source of security problems. On the downside, with a hardware firewall, we can't install other software on the device and have to hope that the supplier has included everything we're going to need, otherwise we're stuck with it. As the firmware included on hardware firewalls is updated from time to time, it is possible to add capabilities to a hardware firewall, however it is not quite as simple as simply installing a package containing a new IDS tool, a SOCKS proxy server, or additional firewall capabilities.

A software firewall is either built from a custom Linux distribution, such as *Smoothwall* or *IPCop*, or by modifying a standard installation. If we only have one system, then we can run a firewall on that box, even if it's a workstation or a web server. A software firewall generally provides more functionality than its hardware counterpart, and can be easier customised for specific environments, rather than going with a cookie cutter configuration. A software firewall also generally runs perfectly on standard hardware, rather than a neat purpose-built embedded platform, so recycling an older PC as your own dedicated firewall box becomes a straightforward option.

Routing

Routing is an important capability of many firewalls, but what can we make Linux do?

Routing a packet back and forth between two networks – possibly performing NAT along the way – is about the extent of what most firewalls will ever need to do. However, there are many other things that can be done through the routing system in the kernel, and through other traffic management tools.

The most obvious routing change is to make use of a routing protocol to distribute routing information between firewalls and other gateways. Depending upon our upstream connection to the Internet, we may have to talk BGP to our upstreams in order to have traffic routed back to us correctly. Internally, we use a selection of protocols to distribute our routing information. The most well known protocol is RIP, which is a very simple 'hops' based method of deciding the least cost path between any two hosts. Even if it's one hop over a slow serial connection, versus two hops over Ethernet, it will pick the serial connection to route our packets unless it goes down for some reason.

OSPF is a better method of handling routing, as we can set the cost of a particular interface based upon the speed of the link, and if that is the way we want traffic to be routed. OSPF also updates due to a link failure significantly quicker than RIP's standard ninety seconds, which can seem like a lifetime when the network should be passing traffic. When designing a network incorporating redundant systems, a reliable routing protocol such as OSPF can avoid manually having to modify routes or physically moving cables when a router or a server fails.

Kernel & *iproute2*

We can also manage multiple active connections, either internally or externally, using the Linux kernel. Using the *iproute2* tool, which provides full control of the routing capabilities of the kernel, as opposed to the regular **route** command that doesn't do much of anything. *iproute2* allows us to create multiple routing tables depending upon the source of the packet, as well as specifying multiple gateways for a route. By incorporating the *dead gateway* kernel patch, we can route packets out multiple upstreams and in the event that one fails, everything will simply go out through the alternative path. As we can't distribute an IP block across multiple providers without using BGP,

GLOSSARY

IGP (Interior Gateway Protocol)
For exchanging routing information between gateways (hosts with routers) within an autonomous network (for example, a system of corporate LANs). Routing information can then be used by Internet Protocol (IP) or other network protocols to specify how to route transmissions.

EGP (Exterior Gateway Protocol)
Commonly used to exchange routing table information between two neighbour gateway hosts (each with its own router) in a network of autonomous systems; particularly on the Internet. Each router polls its neighbour at intervals between 120 to 480 seconds and the neighbour responds by sending its complete routing table.

RIP (Routing Information Protocol)
Widely used IGP for managing router information within self-contained networks, eg a corporate LAN or group of LANs. RIP is an effective solution for small homogeneous networks; but on larger, more complicated networks, RIP's transmission of the entire routing table every 30 seconds causes unnecessary extra traffic in the network. The major alternative to RIP is OSPF. Manufacturers tend to include RIP support within a router designed primarily for OSPF, as the latter is the legacy protocol.

OSPF (Open Shortest Path First)
An older IGP that is used in many corporate networks, OSPF is used within larger autonomous system networks in preference to RIP. Unlike RIP, which sends the entire routing table, a host using OSPF sends only the part that has changed, and only when a change has taken place.

BGP (Border Gateway Protocol)
An EGP designed for exchanging routing information between gateway hosts (that each have their own router) via TCP in a network of autonomous systems, BGP is often used between gateway hosts on the Internet. The routing table contains a list of known routers, the addresses they can reach, and data associated with paths to each router. Hosts

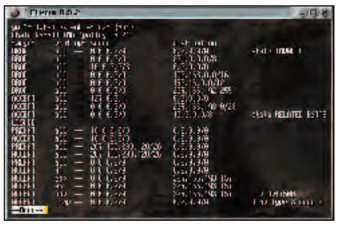
»

GLOSSARY

« using BGP send just updated router table information only when one host has detected a change.

DNS (Domain name system)
A domain name is a meaningful and easy-to-remember "handle" for an Internet address – DNS is the way that Internet domain names are located and translated into IP addresses. Lists of domain names and IP addresses are distributed throughout the Internet in a hierarchy of authority.

Dynamic DNS
The next-best thing to having a static IP – when the Internet was conceived, its architects didn't foresee the need for unlimited numbers of IP addresses – consequently, there are not enough IP numbers to go around. As a get-around, many ISPs limit the number of static IP addresses they allocate, and economise on their remaining numbers of IP addresses by temporarily assigning an IP address to a requesting Dynamic Host Configuration Protocol (DHCP) computer from a pool of IP addresses. The temporary IP address is called a dynamic IP address.



***iptables* is split into a number of tables and rules, each of which perform a different function at various stages of the routing process.**

***netfilter* is the packet-filtering hook system in the 2.4 and 2.6 kernels, through which *iptables* interfaces with the core kernel.**

« it does mean that outbound connections will continue to function. Using dynamic DNS functions, or through the use of tunnels, we can continue to function in a degraded state until the other connection is re-established. Of course, if two links come from the same provider, then many of the issues with routing IP addresses upstream can be negated by having the appropriate configuration at our upstream.

iptables

***iptables* is the core packet-filtering technology under Linux, but how does it work?**

Behind any firewall that runs on Linux – whether it is a software built system, such as *Smoothwall* or *IPCop*, or a hardware device running embedded Linux – the underlying system which filters packets, performs network address translation and keeps track of what nasty traffic we block is *iptables*. *iptables* plugs into the *netfilter* hooks in the Linux kernel to push packets through the ruleset which have configured for our firewall.

iptables splits the rules into three separate tables depending upon the function of the rule. The *filter* table is the most obvious, and lets us permit, block or log packets entering or leaving our network. Within the *filter* table, there are three separate chains, in which the specific rules are contained. Packets entering the firewall, leaving the firewall and those generated locally are filtered separately.

The *nat* table is used when we need to perform network address translation. The most common use of NAT is when we want to hide a private non-routable network range behind our firewall because we only have a single public IP address. However, NAT can also be used to allow traffic from the outside public Internet connect to a service running on a local system on a non-routable block. This can have many advantages, as we can have a single outside address with ports NATed towards multiple inside systems. As with the *filter* table, the *nat* table has separate chains of rules for packets entering, those leaving, and anything generated locally.

Mangle

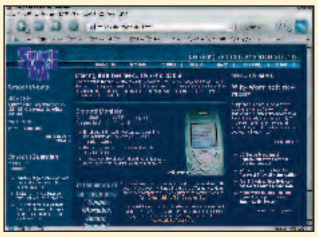
The most obscure table in *iptables* is the *mangle* table, which allows us to modify specific elements of an IPv4 packet, including the type of service (ToS) field, as well as assigning a specific mark value for a packet for later filtering or routing.

Marking packets is particularly useful, as the mark will persist following a routing decision, so we can specifically perform NAT on a packet based upon the interface the packet arrived on, which can't be done through the standard capabilities of the *nat* table. The *mangle* table is split into five separate rule chains, each operating at a different stage of the routing process and depending if the packet is being routed through the firewall itself, or routed to or from it directly. Being able to fully utilise the *mangle* table requires some understanding of how the various fields within a packet can change the functionality of the firewall.

***FreeS/WAN* is the Linux implementation of *IPSec*, creating secure IP tunnels between two networks.**

SMOOTHWALL

It's a Linux-based firewall distribution, but why is it so popular?



***Smoothwall* is a Linux-based firewall distribution that can be installed on almost any hardware we can find.**

Linux will happily run on hardware older than the hills, making it an ideal platform for a device that doesn't require many resources, including a firewall. Unless we're dealing with tens of Mbits of traffic, we certainly don't need bleeding edge hardware or even something released in the last couple of years. Even something like an old P90, or even a 486, will be able to handle the firewalling needs of a home or small business, with cable, DSL or dial-up Internet access. We can go and install a standard Linux distribution, be it Red Hat, Fedora, Debian or SUSE, then customise it to fit the needs of our firewall. The first obvious disadvantage of this is that we need to have enough knowledge to get the base Linux distribution up and running before we can do something useful with it. Beyond this, we need to know how to setup a firewall, make the network interfaces work, and even worse, secure what is left of the core installation.

Easy for novices

At this stage, the more hardened Linux users among us are picking through our junk piles searching for an old box to play with. On the other hand, beginners often don't even consider trying to get an old system working to go through the effort of trying to go through the process of making a firewall out of it. *Smoothwall* takes away this anxiety by providing a simple distribution that can be installed by even the most novice Linux user, or even someone who doesn't know what Linux is.

Smoothwall is a complete Linux-based firewall, offering packet filtering, network address translation, caching proxy and DNS, VPN and tunnelling capabilities, along with an easy to use web-based interface taking the command line fear out of the equation. It literally takes a few clicks to get *Smoothwall* installed, and once configured with the appropriate Internet connection, we can be up and running in less than an hour, just as if it were an appliance bought off the shelf. Obviously, as this is running off a complete system with a decent amount of persistent storage, we can run lots of other network services on the system, rather than having to off load them onto something else. We can configure a caching proxy service through the web interface, through which we can limit access to sites as well as being able to log all web activity of our users.

Smoothwall comes in both free and commercial distributions, depending upon the capabilities we need, and more information is available at http://smoothwall.net/

***Snort* monitors packets passing through our network for unwanted activity and can either log the information or set firewall rules automatically.**

Lots of information on the specifics of *iptables* and the functionality of each individual chain and table can be found at **http://netfilter.org**

IPSec with Linux

However much the behaviour of our users may make us want to, we can't block everyone out of our network, so how do we enable authorised access securely?

Securing connections into our network is a vital role of a firewall, and Linux has a number of implementations of the popular *IPSec* protocol. *IPSec* is a cross-platform standard for transporting IP securely across a public network, such as the Internet. This allows us to create VPNs into our network, permitting outside users to access resources on the network as if they were directly connected. Of course, it's important to ensure that the authentication is secured, as well as the data being passed back and forth. With *IPSec*, we can create client VPNs, where a roaming user or someone from home can connect into our network, as well as creating a site-to-site tunnel where we can connect two separate networks which are connected via the Internet. This makes it ideal for branch offices or people working from home, where private leased circuits are impractical due to location or cost. Running *IPSec* across DSL or cable modems is generally far easier and cheaper than providing a dedicated circuit, and offers the obvious advantage of being able to run multiple tunnels to multiple sites across a single link.

IPSec is provided natively in the 2.6 kernel, however in order to run an IPSec tunnel to or from a system running 2.4, it is required that we patch the kernel with *FreeS/WAN*, as shown in the image above left. *FreeS/WAN* provides all of the kernel and user-space tools required to create either an *IPSec* client, or an end point for client connections.

As *IPSec* is a standard across devices, we can connect a Linux-based *IPSec* system to Cisco Pix boxes, Checkpoint firewalls, and pretty much anything else out there that has pretensions to be a security product. This helps in reducing costs by recycling existing hardware, rather than replacing it all with brand new hardware running Linux.

Intrusion Detection Systems

A firewall is always the first line of defence, however it is important to secure the traffic we allow to pass through.

While the primary purpose of a firewall is to block unwanted traffic entering our network, obviously we do still have to permit some packets to enter and leave our network, such as those for mail, web or remote access services. As we can't avoid such traffic on our network, it's important to be able to distinguish valid requests and information from malicious attacks against our systems.

An Intrusion Detection System inspects traffic passing through our firewall for common or well-known attacks, such as Code Red, Nimda and many other exploits against network services. Rather than simply looking at layer three and four information as a standard stateful firewall does, an IDS will investigate the specifics of the packet payload. In the instance of a web server exploit, it will detect the signature of a specific malformed HTTP request, or a requested URL which may reference a known exploit against PHP or CGI code.

The most well known Open Source IDS is Snort, which can be found at **www.snort.org/**. As with any signature based process, be it an IDS, virus scanner or other data validator, ensuring that the latest signatures are used is vitally important to ensure the accuracy of any notifications or reports we get.

False positives

An IDS can either run passively, where we will be notified of any event via email or through a web interface, or we can run an active IDS – or Intrusion Prevention system – which watches packets and blocks either individual ports to hosts, or whole hosts themselves in the event that an attack should take place against our network. Naturally, the latter is a more hands-off way to approach, however its always a wise move to log all blocks which are put in place, and ensure that the number of false positives are limited to avoid blocking valid traffic entering our network.

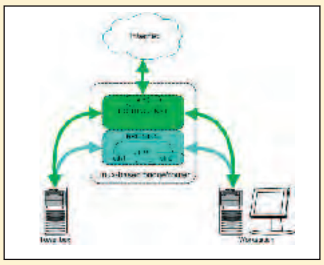
Looking for a great firewalls resource with plenty of software in one place? See *LXF51's Roundup* – backissues available on page 97.

TRANSPARENT FIREWALLS

What if our firewall is a switch?

Usually a firewall is placed on a router, between our local network and the Internet. Packets which are routed back and forth are filtered by the rules we place on the firewall, so traffic either passes through the router, or is blocked before it hits our network. This design is great, however it is not always the most ideal in terms of architecture. It relies on the outside and inside network existing on two separate IPv4 subnets, which may not always be the case within a large network, such as a corporate campus or datacenter.

Linux supports Ethernet bridging, so we can create a switch out of a Linux box with multiple network interfaces. Rather than routing the packets between the interfaces, it will bridge them, as is commonly done on a switch. When an Ethernet frame is switched from one network interface to another, the switch does show up in IPv4 traceroutes, essentially making it transparent to either end of the IP connection. Of course, with standard hardware, it's awfully difficult to replace a network switch with a Linux box, as it's somewhat difficult to provision an x86 box with 24 or 48 network interfaces.



A bridge can filter traffic without being a hop in the routing process, making it transparent to the client and server.

Switched Network

With three or more network interfaces, we can create a simple switched network with an upstream connection and two local network blocks. Depending upon the requirements of our ISP or whatever device is upstream from us, we may have to either bridge or switch packets to or from the Internet. If we have a router that supports a E-1 or T-1 circuit, we can offload all of the firewall and NAT capabilities onto a Linux device without having to separate our network using smaller subnets and more complex routing. A Linux device can simply sit in between the inside network and our router, performing whatever function is required. When a packet or frame enters our system from any network interface, we can make a decision as to bridge or route the packet depending upon the destination and source of the frame or IPv4 packet.

Even when a frame is bridged between two ports, we can perform IPv4 packet filtering on the contents using *iptables*. For example, we could block *ssh* between the two Ethernet networks unless the source MAC address of the frame is that of a known system administrator. Filtering layer two based on layer three information completely breaks the OSI model, however it can be particularly useful when we want to create a completely transparent firewall between two Ethernet networks.

Information on bridging and filtering packets across a bridge can be found at http://ebtables.sf.net/.

What on Earth is... THE CREATIVE COMMONS?

Jono Bacon licenses content without resorting to restrictive copyrights and simultaneously retains the respect of the Free/Open Source communities.

Seemingly contradictory information from organisations like the UK's Federation Against Copyright Theft and a throng of similar US-based bodies tells me that using non-copyright material is killing the music and movie industries. Am I close?

Not at all. The Creative Commons (CC) is a project to *spread* the availability of Free and Open content that you can use.

So, Creative Commons is some kind of filesharing program, then?

No, no. This is all about Free and Open content that is created and licensed in a way that gives you the kind of freedoms that are typical with of much of the software that you use on your computer. The difference with the CC is that the project is primarily based around the licensing of creative content.

What kind of creative content?

Much of the content that is often license under the scheme includes music, video, images, artwork, writing and more.

As much as licenses don't exactly thrill the life out of me, exactly how are these types of content different in licensing software?

I know that licensing is not exactly pants-wettingly exciting, but The Creative Commons is actually quite cool. The project offers a range of different licenses that are in some ways similar to the kind of licenses offered under Free and Open Source licenses.

Such as the GPL?

That's right. The CC licenses are somewhat different however, in the sense that they are not quite so community-focused as the GPL. When Richard Stallman created the GPL, he wanted to create a license that would enable a community to be fostered around a GPL-licensed project. This is why the source code must always be Free and accessible, and few quarrel the fact that the GPL has pretty much served this purpose. It does have some issues though...

Issues? From what I've read on the Web, I thought the 3mighty GPL was perfect?

It is great, and it works well for many, many people, but it does not always fit the purpose for all projects.

Please elaborate!

Well, the GPL is very much based around software. When you read the eyeball-glazing dull legalese that outlines the remit of the license, you can see that it is quite clear that the license is orientated around software. This can be seen in the way it refers to terms such as 'source code', 'compilation' and 'executable'. Due to the fact that these terms refer specifically to computer software and source code, the license is not the best choice for audio/video/textual content.

Right, so it seems...

Hang on a second, there is more. Not only is the GPL not suitable in terms of its terms, but the concept of its freedoms are slightly different to the freedoms that are often applied to audio/video/written work.

As an example, if we take video content as the medium to be licensed, there is no source code for this medium. The only parallels you could draw to the terms of the GPL would possibly be that raw video footage is the source, editing it is the compilation process and the final video is the executable. If you were to license a video under the GPL, you would really need to make the raw video footage available as the equivalent source code. Not only would this be unfeasible in terms of disk space and hosting requirements, but most people would probably just want to be able to modify the final edited product rather than its raw component parts.

This all makes sense. But why would I possibly want to have such a complicated license around a typical video? Surely I just need to have the the rights to give it away for free?

Well, not necessarily. You may want to take a video and make some edits to it to use in another piece. You may also want to add some bits and pieces to the video for yourself. An example of this could be bleeping out those naughty words in a video and cutting more adult scenes if the video is to be played in front of children. These are all modifications that need to be allowed within the remit of the license.

This sounds sensible. Surely the GPL would let me do this though?

Not exactly. One key difference with the GPL is that it is designed to allow the 'improvement' of software. Software is rarely created to be left languishing in a state of disrepair, and most people who write software want to extend and improve it. This is the reason why Free software has taken off so much – the source code is freely available so people can improve it further. This may not be the case for creative content.

How so?

This is where licensing takes into account the rights of the artist. As an example, I am a musician and I write songs that I put online. When I write a song and record it, I want it to be heard in the form in which I wrote it. I would prefer that someone listens to my songs as they were originally written and not some kind of remixed, edited song that has been created from my work on the net.

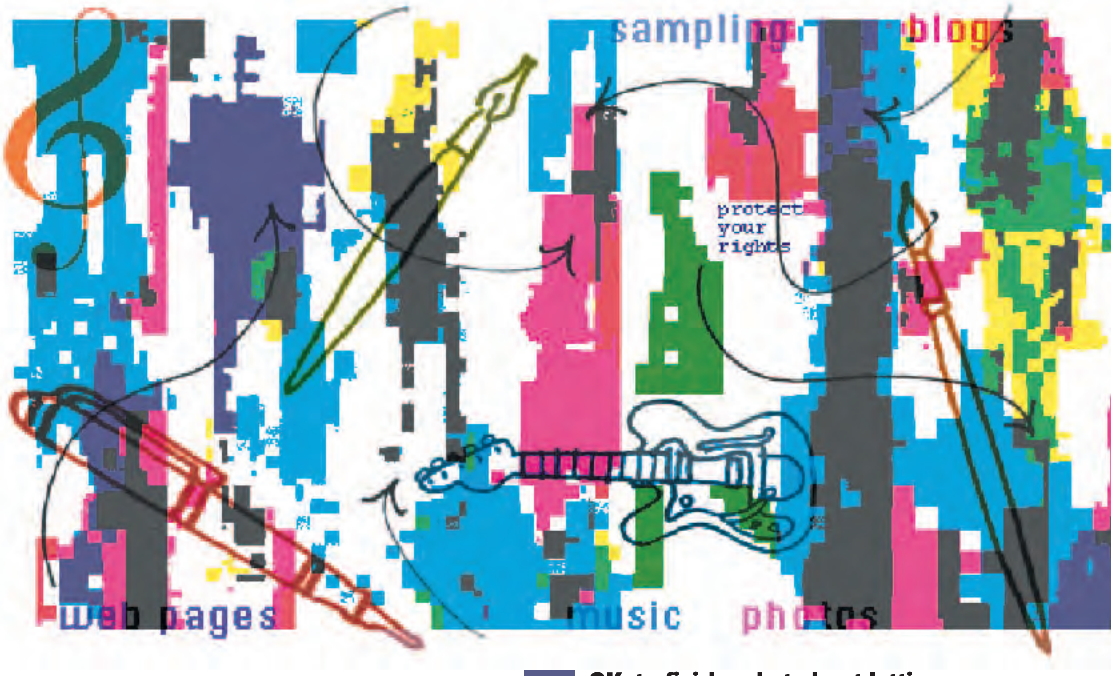
But surely this is not free?

My songs are free in the sense that you can listen to them freely, copy them and distribute them, but they are non-free in that you cannot modify them. The reason for this is that I have made a choice that my work should be consumed in the way that it was intended. This is a common desire for many artists who feel that their work is a creative unit/artefact and a piece of art. In this regard it is common that creative content is not intended to be modified and 'improved' in the same sense as software.

Right, so how does this link in the Creative Commons?

The licenses that are available as part of CC are tuned towards the kind of content that we have been discussing here. There are a variety of different licenses on offer for the different requirements that you may have for your content.

OK, let us ponder a while over an example. Let us say that I write a song about my dog and want to put it online. I want to make sure that people can download and copy it, but I don't want them to be able to change it.



Right – I will make an assumption here that you want to be credited for your work. I will also assume that you do not want people to use it commercially.

Oh yes, that sounds fine...

In this case, you should use the Creative Commons Attribution-NoDerivs-NonCommercial License. This license will give the listener the ability to freely distribute and listen to it under the condition that they credit you and do not use the song commercially. Any of these rights may be waived however if they get written permission from you. This is useful for example if your song was very successful and a commercial radio station wanted to play it. This would count as commercial use, and would therefore require written permission. If the radio station were to seek this permission, you are well within your rights to allow this.

OK, let us now assume that I am not bothered about people using it for commercial use.

In this case, you should use the Creative Commons Attribution-NoDerivs License. This gives the same set of rights but allows commercial use of your work.

Right, lets stir it up a bit: I would like to allow modifications.

In this case, you need the Creative Commons Attribution license. This will allow any kind of modification, but the modifier does not need to necessarily contribute the changes back to you, although this can make sense in the context of audio/video. You may want modifications to be given back to the original work, particularly important in the case of written work and other mediums. In this case, you should use the Creative Commons Attribution-ShareAlike license. This will ensure that the kind of modification rights that GPL fans are fond of can be used in your license.

OK, to finish, what about letting me give them all rights, and I am not even bothered about being credited?

In this case, you can let people modify the work without giving anything back, distribute it freely, allow commercial use and not demand credit with the Creative Commons Public Domain Dedication. This is basically a free-for-all license, in which you grant all rights to the public domain. This is irreversible and you should be completely sure you're happy to sign over your rights to the public domain with this license.

That sounds a bit drastic; is it really that bad?

It is not bad at all, if this is what you want. You may well want to simply create something that anyone can do anything with. If this is your desire for your content, then this is a great choice. If you otherwise want to control some of your copyright (such as commercial use/modifications/credit), then this not the right choice.

OK, so now I have a good idea of all of the options. How do I make use of a license?

The first thing you should do is to go to the CC website and have a good read through the licenses to get acquainted with them. The website is at www.creativecommons.org/.

Just stop there. I am no lawyer and I don't particularly relish the idea of reading through dull legalese...

Don't worry – the CC website is actually a really simple and easy-to-use site. There is a 'Choose a license' button through which you can answer some simple questions, and an appropriate license will be suggested based on the information you provide. When the suitable license is suggested, you will also be given a quick and simple summary of the rights »

as well as a more typical legal jargon-filled diatribe of the license conditions. Not only does this give *you* a great summary of the rights offered by the license, but it also gives *the people using your content* a great summary of their rights.

This sounds really good. The general area of licenses seems so complicated that this sounds like a genuine breath of fresh air.

Oh, it doesn't stop there. The CC team has done a really good job of creating a site filled with easy-to-understand information about choosing licenses, the reasons behind creating Open content and other vital and interesting information. There are also some animated movies and comics that help get the message across.

Once I have chosen my license, how do I actually license my content?

Literally licensing your work is as simple as placing some form of copyright notice with your content in order to indicate how it is licensed to the end-user – the people that will actually be reading/listening watching it. This is often accomplished by putting a small CC button on your website where the content is available; and you should also put a copyright notice in your work if possible. When you have chosen your license on the CC website, you are given links to the simple human-readable list of rights and the complicated legalese license information. You can link to these pages from your website/content.

How do I put a button on a web page. Where do I get the Creative Commons button from?

The buttons are available on the CC website and you can link directly to the buttons from the CC website. As an example you could use the following code:

```
<!-- Creative Commons License -->
<a rel="license"
href="http://creativecommons.org/licenses/by-nd-
nc/1.0/"><img alt="Creative
Commons License" border="0"
src="http://creativecommons.org/images/public/
somerights.gif" /></a><br />
This work is licensed under a <a rel="license"
href="http://creativecommons.org/licenses/by-nd-
nc/1.0/">Creative Commons
License</a>.
<!-- /Creative Commons License -->
```

When you have selected a license, you are given the code above that can be cut and pasted into your website. You are also given a special chunk of RDF code as shown below:

```
<!--

<rdf:RDF xmlns="http://web.resource.org/cc/"
    xmlns:dc="http://purl.org/dc/elements/1.1/"
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-
syntax-ns#">
<Work rdf:about="">
    <license
    rdf:resource="http://creativecommons.org/
licenses/by-nd-nc/1.0/" />
</Work>

<License rdf:about="http://creativecommons.org/
licenses/by-nd-nc/1.0/">
   <permits rdf:resource="http://web.resource.org/
cc/Reproduction" />
   <permits rdf:resource="http://web.resource.org/
cc/Distribution" />
   <requires rdf:resource="http://web.resource.org/
cc/Notice" />
   <requires rdf:resource="http://web.resource.org/
cc/Attribution" />
   <prohibits rdf:resource="http://web.resource.org/
cc/CommercialUse" />
</License>

</rdf:RDF>

-->
```

What is RDF all about? The only RDF I've heard about is Radical Dance Faction, a 1980s-1990s band...

RDF (Resource Description Framework) is a special chunk of code that can feed different types of software information. This code is used to give these special types of software information about the content and how it is licensed.

What kind of 'special types of software'?

As part of the CC effort, the people behind the CC are working to get software publishers to include an RDF reader in their software to display license information.

As an example, if you were using your web browser to look at a website, it may be useful to be able to read the block of RDF code and give you a simple summary of your rights. There is already an extension to the Mozilla web browser that can do something like this called MozCC (**www.yergler.net/projects/mozcc/**). The idea is that filesharing programs can use this information too.

Surely a filesharing program does not visit a website though? How does this work?

The concept is that inside the information tags of some content (such as the information tags in an MP3/OGG files), you should include a link to a website where the content is hosted and displays the license information. The filesharing program would then read in the RDF information automatically, and display it within the interface of the program. Not only does this verify the license information, but it also prevents people making up lies about how your work is licensed. This ensures that your licensing is secure and it could also have the added benefit of bringing people over to your website.

This sounds pretty nifty. So is there a big vat of CC-licensed content available on the Internet then?

There is indeed. The first place you should look is the Get Content link on the main CC website. Here, you cannot only browse through the variety of content available, but you can also search the archive too. In addition to this, you can specifically hunt out content with commercial and modification allowances in their licenses. You should also take a look at the Commons Content website at **http://commonscontent.org/**. This site has a large list of CC-licensed content available for download.

Other than music and video, what other types of content are there that I can license with the Creative Commons licenses?

There are many different types of content that are begging to be licensed with a CC agreement. One common type of content is blog entries.

Huh?

For anyone that's been living on Mars: blogs (short for 'weblogs') are frequently updated personal journals and lists of links to other sites typically hosted on a website intended for public consumption. – a 'blog' entry is an entry on such a site. Since the blogging phenomenon went mainstream in 2002 though, there have been complaints that the sheer volume of 'blog noise' can cripple search engines' effectiveness on certain subjects, if there are a lot of blogs that mention for instance, a particular news story. The CC has been used extensively for these types of blog entries and even O'Reilly uses them on the O'Reilly Network (**www.oreillynet.com/**).

But why would you want to use them for blog entries?

One of the great things about blogs is that the information from someone's blog is often useful and quoted in someone else's blog. Due to this sharing of information, the CC licenses are ideal for licensing this kind of content. As an example, my blog entries on jonobacon.org are licensed under the Creative Commons Attribution-NoDerivs License. This means that people can use my blog entries for commercial and non-commercial use, but they cannot be modified and credit must be appended. I chose this license because I am the only person who writes my blog and I could not think of a reason why my blog entries should be modifiable.

Hang on a second. Surely the Free Software Foundation (FSF) has a suitable license for written works?

The FSF has indeed created a license called the GNU Free Documentation License (FDL) that serves a similar kind of purpose to the the GPL, but for documentation. The problem with this license though has been that due to some aspects of the license it has been deemed non-free by a number of groups, including the Debian project. The Debian Free Software Guidelines (DFSG) that mandate how software is allowed in Debian (they are there to ensure Debian is 100 per cent free) clash with the FDL.

Why do they clash? Surely if it comes from the Free Software Foundation, it must be free?

The Debian DFSG are not compatible due to a section in the FDL about 'invariants'. This invariant text cannot be modified or taken away and is therefore deemed non-free. This has caused a number of discussions and debates on the Internet about how free the FDL really is. At the time of writing, the situation is still not fully resolved though, and as such any FDL-licensed content cannot be included in the official Debian archives.

So it seems that the Creative Commons is the right way to go then?

I would say so, yes. Not only does the CC have a variety of licenses available to closely match the different requirements you might have for your content, but the site is simple to use and it is easy to license your work. With the addition of the RDF code that you can add to your web page to allow future software to make use of CC content, you have a fairly formidable licensing system. Certainly in terms of written/audio/video content, the CC licenses are very attractive.

What is the future for the Creative Commons?

The CC project has a large number of enthusiastic contributors who are eager to make the licenses as simple-to-use, flexible, and as useful as possible. This not only includes creating new licenses, but it also involves creating special licenses for special types of content. An example of this is the Creative Commons Sampling license project that is creating a new type of license that retains credit but is very specific in how the content can be used in the context of music sampling. (the use of snippets of existing music in a collage fashion in a new work) This is making licensing that type of content much easier.

This is how licensing should be – nice and simple.

You're not alone in that opinion! There are many people who are unsure about how to protect their rights as either originators of content or contributors. Not only is the CC project creating a number of useful licenses, but there is a lot of useful and fun content available on the Internet that can be used and enjoyed in a variety of different ways. This can be useful for those who need free content for their work and can also be very handy for those who want to create content for the fun of creating it and giving it away safely. The key to the success of the CC project has been its simplicity and flexibility. I am sure that the CC license and services offering will no doubt expand further in the future, and I look forward to seeing how the project approaching the licensing challenges of new media.

Tutorials

Our experts offer help and opinions on a whole host of Linux applications

YOUR GUIDE TO GETTING THINGS DONE!

Whether you are just starting out in Linux, or an experienced veteran, there's always more to learn. Every issue of *Linux Format* is packed full of practical advice, and nowhere is it more concentrated than in our tutorials. pages.

Here you'll find expert guides to all sorts of things, from Basic Linux usage to understanding and deploying network solutions, from simple script coding to the complexities of Perl regular expressions, Java server apps and more. We aim to bring a good mix of tutorials to each issue, but if you have any suggestions for topics you'd like us to cover, why not contact us, by email at **linuxformat@futurenet.co.uk** or by snail mail, or log on to **www.linuxformat.co.uk** and post your suggestions in our special forums? Hope to hear from you soon!

Nick Veitch EDITOR

HOW CODE IS REPRESENTED

Including code in magazines can be tricky, but we hope our notation will help it become clear. When lines are too long for our columns, the remaining text appears on the next line in a solid blue box:

```
procedure
TfrmTextEditor.mniWordWrapClick
(Sender: TObject);
```

otherwise, there is usually a gap between lines:

```
begin
  mniWordWrap.Checked := false
end;
```

Usually, you'll find the code on our CD/DVD too.

THIS MONTH TEACH YOURSELF...

66

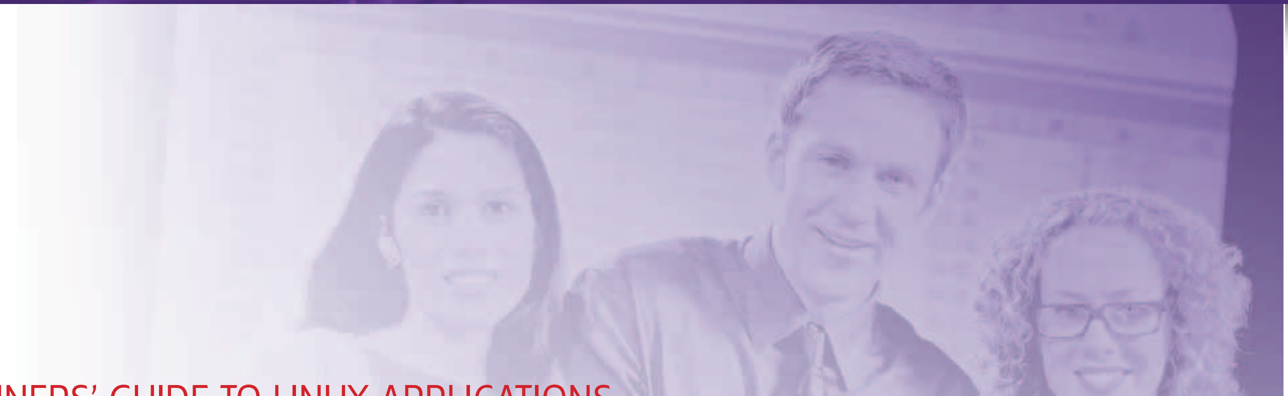
TIP OF THE MONTH! OpenOffice.org Gems

OpenOffice.org is a huge application, so it's no surprise that people often miss out on some of its smartest features. To give you a head start in looking around and getting the most out of the suite, here are the top three features in *OOo Writer* we think people might have overlooked: document comparing, change tracking, and easy dictionary installation. Of the three, the first two are available in *OOo1.1*, and the last is available as of v1.1.1.

Comparing one document with another, regardless of format, can be done using the Compare Document option under the Edit menu. First you need to open the first source document using File/Open, then use Compare Document to select the second source document. *OOo* will then compare the two and highlight changes in red, and allow you to accept or reject each change.

Using the Track Changes option (Edit/Changes) you can have *OOo* automatically record what edits you made to a document – what was removed, added, or altered. This is again done by highlighting text in red (although this does change automatically depending on which user edited the text), although you can turn it off by clicking the 'Show' item in the Changes submenu. If you find all your changes are marked as by 'Unknown', it's because you didn't enter your name when you first ran *OOo* – to rectify this, go to Tools/Options, then select User Data and fill in your name.

Installing new dictionaries used to be a hassle until the 'Install new dictionaries...' option was added in *OOo1.1.1*. It's under the File/AutoPilot menu, and it allows you to select precisely which dictionaries you want to download and install, meaning that you only need follow the wizard then restart *OOo* to get your new language working.



On the CD and DVD

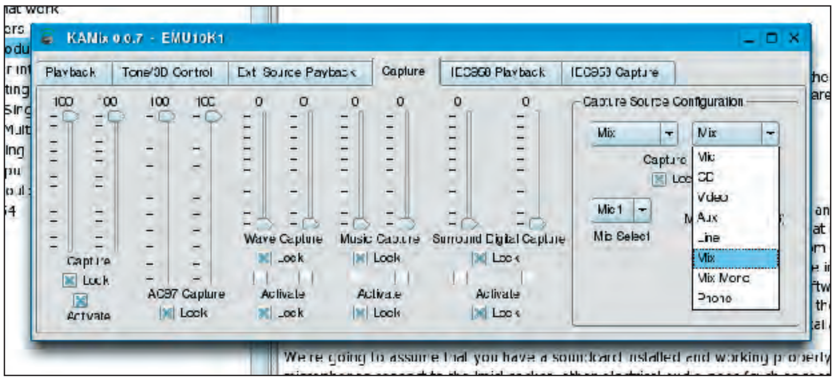
BEGINNERS' GUIDE TO LINUX APPLICATIONS

Audio-editing with Audacity

Sound can be played with in the same way as text in a word processor or photographs in *The GIMP*. So grab a few WAV files and prepare to take a trip into stereoscopic sound with Andy Channelle.

Past tutorials have looked at applications for playing CDs and ripped music files and the usability enhancements a little sound can add to, for instance, email applications. But what if that isn't enough to satisfy the creative itch? Where can you turn to if the ready-made sound themes from www.kde-look.org don't cut it, or if you have the conviction that the music world would be all the better for your presence in it? The answer, obviously, is your PC. And while Linux doesn't yet boast the breadth of audio software available on Mac or Windows, what is available is very good. We're looking at *Audacity* this month, the audio equivalent of *The GIMP* – so even if you only want to dabble occasionally, it is well worth the installation.

We're going to assume that you have a sound card installed and working properly; and that you know that microphones connect to the 'mic' socket, other electrical audio gear (such as record players) connect to the 'line' socket and speakers connect to the 'speaker' socket.



Soundblaster Live cards have lots of options for recording and playback.

You can get the latest version of *Audacity* – 1.2.0 – on all the main platforms, which may be vital if you end up collaborating with Mac or Windows-using musicians, and for Linux it is available in a variety of guises. In addition to a source release, you can install to RPM, Debian or Gentoo-based systems, there is also a self-contained binary file that can be simply uncompressed into a folder and launched. You can download from http://audacity.sourceforge.net/unix.php. Windows and Mac users have the luxury of VST plugin support, but due to licensing issues it is not yet available for Linux. There is, however, a pretty good selection of native effects plugins that are bundled with the application by default.

The first thing to do is grab a sound file. Depending on the installation, it should be possible to open up files in WAV, MP3 or Ogg format. Opening up a single file just means the normal 'File> Open' move, but *Audacity* is not limited to single files, it can mix as many tracks together as your CPU and hard disk can manage. To add a track to an already open sound file, go to 'Project> Import Audio' and select the necessary file using the browser. Hitting the Play button will start both files. Welcome to the world of digital multi-track recording!

Get it in!

There are two main ways of getting sound into *Audacity*: either importing a previously created/ripped/downloaded sound file or recording your own. The former we've already mentioned briefly (and there isn't much to it) so here are a couple of examples of real-world uses for *Audacity* in two contexts:

- Single track recording used to 'back up' a vinyl LP or non-copyright CD for 'fair use' purposes such as listening on your PC or an in-car CD player; or capturing an audio stream for later unconnected listening, perhaps on a different device.
- Multi-track recording of a band or other performer, with each element of the performance on a separate track for mixing.

Any old vinyl!

Anyone over 25 years old is likely to have a few old LPs: chances are that quite a few of them will not have been re-released on CD. In this short project we're going to take an album, rip it into *Audacity* and break it up into tracks ready for exporting in a format suitable for playback on a PC or burning to CD.

THE LAW IS AN ASS

Did home taping kill the music biz?

Technically, under the recently passed European Union Copyright Directive (EUCD), making copies of CD and LPs that you have purchased for your own personal use is against the law. The issue of 'fair use' which has been enshrined in copyright law since its inception has yet to be tested under this new regime, though. Likewise, time-shifting broadcasts was judged legal in the eighties, but the EUCD muddies the waters. It is unlikely the BPI is going to knock down your (or my) door for digitising a 13th Floor Elevators album or recording *Woman's Hour* from BBC Radio 4, but my opinion doesn't constitute legal advice in any way, shape or form.

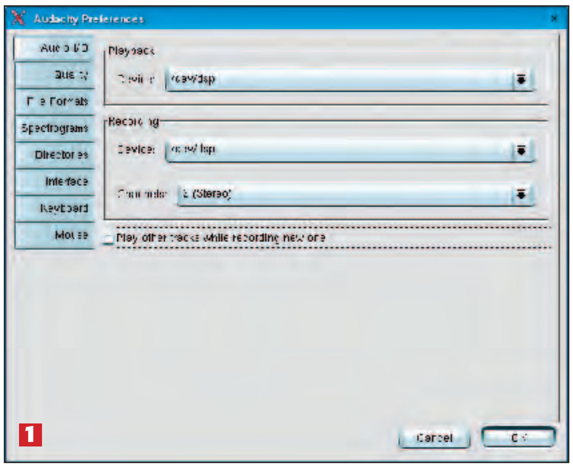
To record from an LP, we can choose to record individual tracks and save them out or – and this is easier – simply record a whole side, and then break up the file into tracks afterwards. If you have an older hard disk with the resulting space limitations, the former will certainly be best; but we're doing the latter as we have acres of space.

The first step is to connect our turntable's AUX out ports to the 'line in' socket on the sound card and then select the correct source – 'line' – in *Audacity*. Before we actually record anything, it makes sense to sort out how the sound will be recorded.

First steps with Audacity

1 Select 'File>Preferences' and go first to the Audio I/O section. Ensure the correct card is selected for input and output (in our case the device is /dev/dsp) and, assuming you're recording a stereo album, make sure the Channels drop-down is set to '2 (Stereo)'. As we're not yet concerned with multi-track recording the 'Play other tracks...' radio button doesn't need to be selected.

Go into the 'Quality' dialogue and make sure the default sample rate is set to 44100Hz, and the default sample format is set to 16-bit, unless you have hardware than can handle better sample rates.



Playback

Device: /dev/dsp

Recording

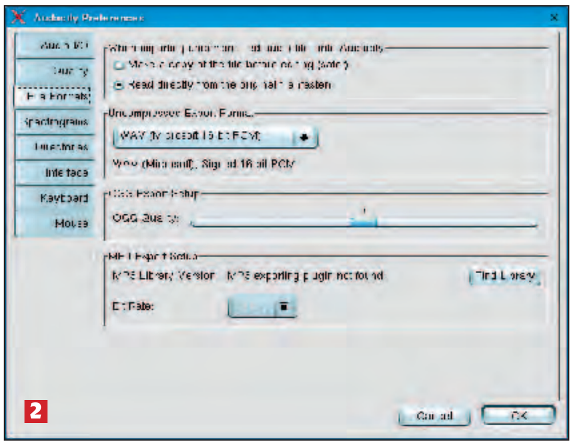
Device: /dev/dsp

Channels: 2 (Stereo)

□ Play other tracks while recording new one

Selecting the correct card for input and output.

2 Select the 'File Formats' dialogue and choose a format appropriate for your project. In our case we're using a 16-bit uncompressed WAV format for compatibility purposes. Hit the 'OK' button to go back to the main screen.



When importing uncompressed audio files into Audacity

○ Make a copy of the file before editing (safer)

● Read directly from the original file (faster)

Uncompressed Export Format

WAV (Microsoft 16 bit PCM)

WAV (Microsoft), Signed 16 bit PCM

OGG Export Setup

OGG Quality:

MP3 Export Setup

MP3 Library Version: MP3 exporting plugin not found

Bit Rate:

Once you've selected your parameters, it's time to record.

Now put the needle on the record, hit 'Record' and sit back to wait for Side A to finish playing. We had the best results with the source volume set to about 0.8 on the slider, but you'll need to experiment.

Once the tracks are recorded, we can select each individual track waveform and select 'File>Export Selection As WAV...' (use the Ogg or MP3 option if the tracks will only be played through XMMS) and give the file a path and name. Do this for each track before recording the second side to complete the project.

Before output, it is also possible to run the file through a filter to get rid of some of the crackles that have characterised the vinyl sound (unless you like that sort of thing). Simply select part or all of the file and do 'Effect> Noise Removal...'. Depending on the size of your files and the speed of your hardware this could take some time.

Laying down the tracks

Multi-track recording is the more complex (and more creatively satisfying) way to use *Audacity*, and while the application is not quite as comprehensive as some commercial software, it is much better – and easier to use – than the old 'portastudios' that many bedroom bands used to record their tracks in the past.

»

TRACKS

More than just an audio-editing application

There's more to audio recording than just sound. Beyond the audio tracks mentioned already, *Audacity* can also use:

Time tracks

These alter the speed of playback for an audio track. However, be aware that this also affects the pitch of a sample, so use with caution. Clicking on the options button in the left-hand panel allows us to set the upper and lower 'range' of pitch shifting as a percentage of the original pitch.

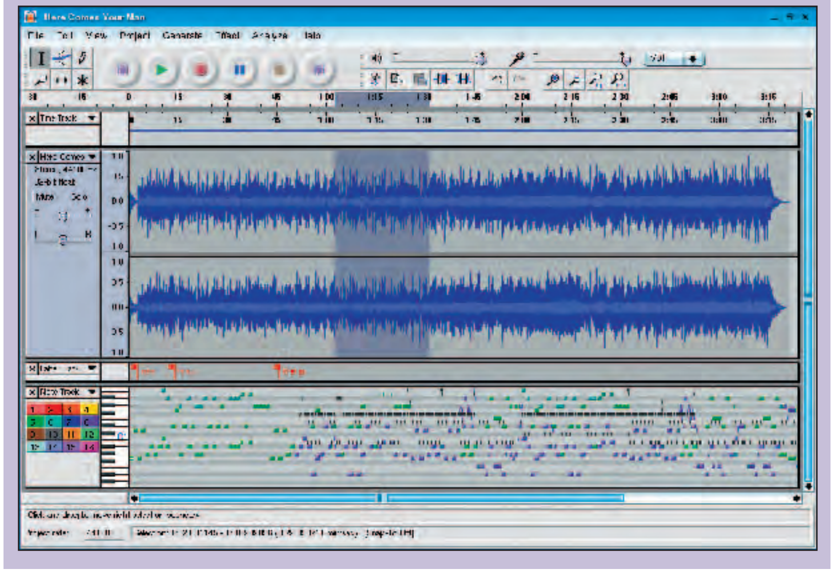
Label tracks

This track type simply allows us to put a short text string in a track – for instance *"vocal begins"*, *"guitar solo" etc* – to bring a little more organisation to a recording session. Think of it as a cue sheet. Once you have a label track available, add a new label by positioning the cursor somewhere in the audio file and selecting 'Project>Add Label At Selection...' or use the keyboard shortcut by hitting **Ctrl+B**.

Note tracks

One for the future: note tracks contain MIDI data, but though it is possible to load and accurately display a file, it will not be playable or editable. When this feature is added, *Audacity* will metamorphose from run-of-the-mill app into a pretty hot studio all-rounder. Until then though, it's probably best to ignore this feature.

You can add or import any of these tracks to your project by selecting the relevant entry in the 'Project' menu.



« If you've followed the first part of the tutorial, you'll need to go into the 'Preferences' dialogue and ensure the 'Play other tracks...' radio button is highlighted (you ought to make sure your sound card is capable of duplex operation too).

It will also be a good idea to think about your needs before choosing between '1 (Mono)' and '2 (Stereo)' as the recording method. The former will give better performance and is adequate for most uses while the latter will allow for a more polished project.

Select the appropriate source, hit the 'Record' button and "Cue" the performer. While the track is being recorded, *Audacity* provides visual feedback on the waveform: if it shows an almost solid block of blue, that's your cue to turn down the input volume until the waveform shows distinct peaks and troughs, and then start the recording again.

When the initial recording is satisfactory, you can overdub the second element (make sure the new instrument/mic is plugged in) simply by hitting the 'Record' button again. *Audacity* will create a new track on the default specification and start recording. Magically the previous track(s) will be playing!

The number of tracks available depends completely on hardware: RAM, hard disk speed, sound card and processor are all significant. As a benchmark we've been recording on *Audacity* using an old Celeron 500MHz PC with 192MB RAM/40GB disk and Soundblaster Live/512 sound card. This modest system handled five stereo (or 10 mono) tracks without any difficulty.

Get it out!

Once your masterpiece has been faithfully captured, it is time to mix it down into a format suitable for burning to a CD or uploading onto the Internet. Due to space constraints, we'll have to cover the process of editing and mixing tracks in a future tutorial; here we'll just be covering the two methods that *Audacity* provides for outputting a final track in the format of your choice.

Audacious Outputting

3 The first option is called a Quick Mix, which simply 'bounces' the selected tracks (including Envelopes and effects) onto a single stereo track. To create a quick mix, use Shift+left-click to select the various tracks that you want to include, and then do 'Project>Quick Mix'. It's worth noting that this method will remove the originally selected tracks, so *make sure you save the project before committing*. It's always possible to undo should a disaster occur though.

AUDACITY – NAMING THE PARTS

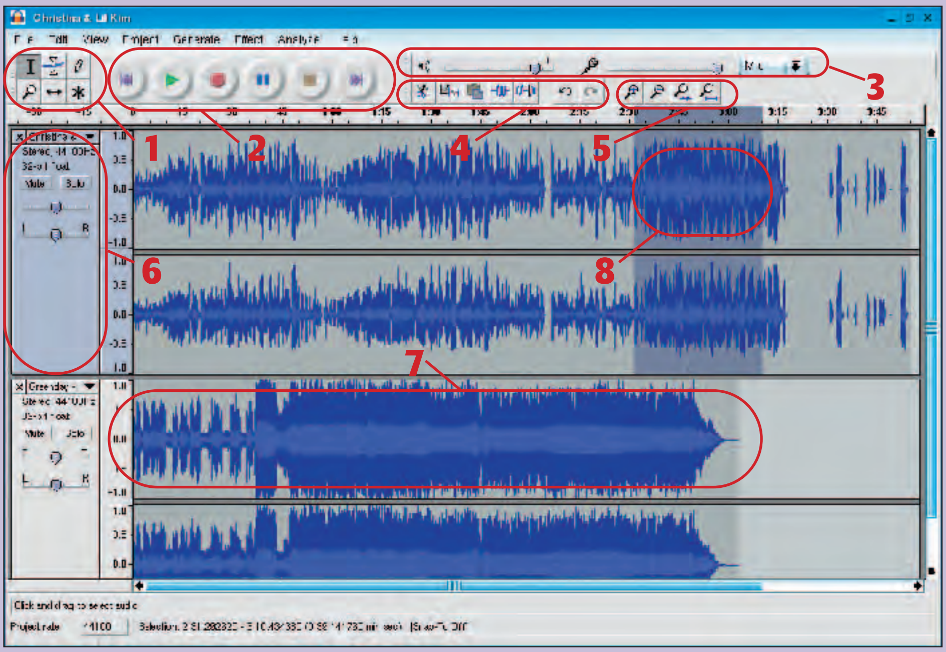
A whistle-stop tour of the GUI

1 Toolbox

This suite of six tools in *Audacity*'s Toolbox in the top-left corner of the window controls the way the mouse works within the main window. These are:

Selection

Using this tool, we can select specific areas of the sound file to cut, copy or alter in some way. Simply click and drag across the waveform (see below) to highlight it. By positioning the mouse at either extreme of the selection, it is possible to adjust it without having to reselect the whole thing. Depending on the project, we can alter the selection method by doing 'View>Set selection format>[option]'. For general music use, choosing 'min:sec (snap to samples)' is sensible, but if you are putting together the soundtrack for a video project, it's better to opt for one of the PAL settings. You will see the effect of these settings on the status bar at the bottom of the main window. However, be aware that this *doesn't* alter the format shown in the time bar ranged across the top of the window, that will always display minutes and seconds.



Many *Audacity* tools are covered by the Undo process, but do make sure that you incrementally save files that you are working on, in case you want to step back further than the program allows.

Envelope

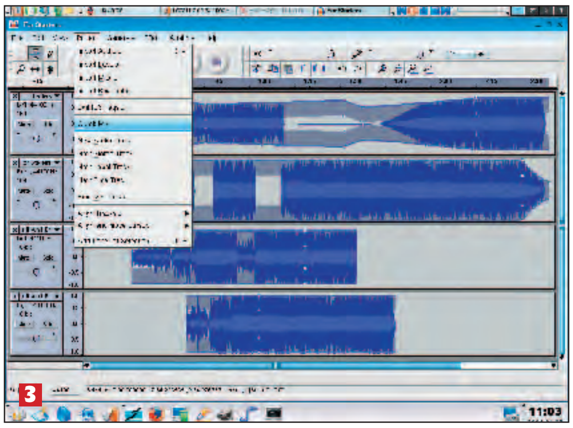
This tool is used for changing the volume of a sound file over time. Simply click anywhere on the waveform to add a new control point, and adjust it by moving the mouse up/down and left/right. Cleverly, *Audacity* does its best to ensure you don't get rapid volume jumps – so you will see, as you make more radical changes, that the envelope assumes a graceful curve. There are ways to change the 'Gain' of the file itself, but this works on a universal basis; using the envelope tool gives you some of the advantages of a very expensive automated mixing desk while providing the control necessary for making very fine adjustments before playback.

Draw

The Draw tool is one that you may never need, though it's good to have it available. It is used in situations where, for example, you have a pristine vocal take with a small imperfection in the wave form, or a recording of a precious piece of vinyl with a tiny but hideous scratch on your favourite bit. With the Draw tool, we can zoom into individual samples and adjust the position of sample nodes on the track, removing the click and replacing it with a smooth line. Using this effectively takes some effort, so make sure files are backed up before beginning the surgery, and is not really useful for solving any major difficulties.

Multi-tool mode

A pain to start with, this tool comes into its own after some practice. In fact, you may not ever touch the other tools after a few hours' work. When in Multi-tool mode, the left mouse button acts as the select tool, while the right becomes a zoom tool, and hovering over an envelope line (thick blue on the waveform) will give the pointer the properties of the Envelope Tool. A single right-click will zoom out one level, and zooming in requires dragging with the right button across the area to be viewed. Zooming into the closest view turns it into the Draw tool, and, using the

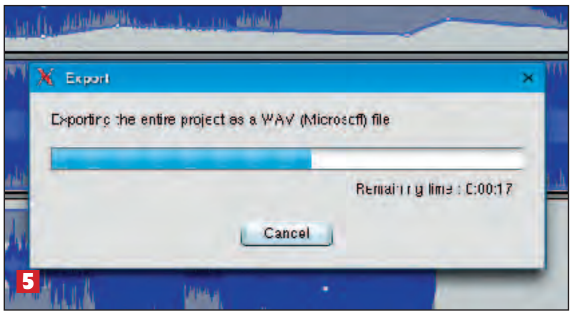


Always save work before using Quick Mix, just to be safe.

Quick Mix is useful if, for instance, you've created a complete guitar part from various takes. Your collage of individual riffs can be assembled across a number of tracks with the appropriate effects, and then merged into a more manageable single file before final mix – very handy indeed if your instrumental skills are a bit rusty!

4 The second option renders the entire project as a single file, which can be a WAV (for CD burning), or a compressed format such as Ogg or MP3. You'll need to make sure you have the correct libraries installed for output, these are 'oggenc' and 'Lame' respectively.

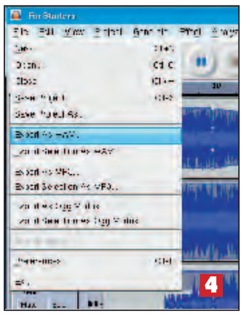
5 To complete the operation, simply select 'File > Export as (format)...', navigate to the desired location and supply a file name. The process for a four track, four-minute song on the test hardware mentioned above took approximately 30 seconds to render. LXF



Exporting the entire project as a WAV (Microsoft) file

Remaining time: 0:00:17

Even on an old machine, it won't take too long to render your track.



'Export as WAV' means your track is the right format to listen to on a regular CD player.

handles at each end of the window it can be used to time-shift the sound file.

Time Shift

When working on multi-track projects, it is of course essential to be able to move individual tracks around. Backing vocals from a first chorus may sound better under the second one, for instance, or the various elements of a a Richard X-style sound clash need tweaking. With this tool, you can simply click and drag the appropriate element to its new location within the project.

Zoom

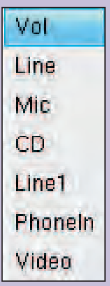
This zoom tool is perhaps the most simple and self-explanatory. Left button zooms in, right button zooms out, and to look at a specific location click and drag across the sound file. You can also use the Zoom tools in conjunction with the selection tools for a little more control. There are other Zoom tools – see 5.

2 Transport controls

These should be familiar to everyone that has used a cassette or CD player! These controls do as expected: skip to beginning, play, record, pause, stop, skip to end. However, occasionally its useful to play a small section over and over to exactly define a start or end point for cutting or effecting; in this case, hold down shift before hitting the play button to 'loop' a selection. While a section is looping, it is possible to redefine the start and end points, but its important to remember that the new selection is *not* reflected in the playback until the file is stopped and started again.

3 Output and input

This section defines the input and output volumes of the entire project, and also allows us to set a source for recording. Setting the volumes is just a case of dragging the widget around until results are satisfactory. The recording sources available will depend upon the sound system in the host PC.



On a standard Creative Labs Audigy/Live card there are options to record from the line in, CD, Mic socket, Video, Phone or, often most usefully, through the sound system itself (the 'Vol' entry), *ie* whatever is playing through *RealPlayer, XMMS* or *JuK* is routed into *Audacity* when you hit record. This is great for capturing (in the copyright-friendly, legally tested manner of a video player) audio streams for later playback. It is a sensible precaution to do some test recordings to make sure the input 'levels' are not too low or high: you don't want to end up with three minutes of unlistenable fuzz (unless that's what you are aiming to sound like...)

4 Clipboard tools

The first three icons in this section are the standard cut, copy, paste triumvirate; the fourth will trim – that is remove – everything that is not selected; while the fifth icon will replace whatever is selected with silence. This last is very good for getting rid of breathing, fret noise, snare fuzz or other annoying noises that can occur when the performer is waiting to let rip! Finally there are the undo and redo buttons which are good for a number of undo/redo operations.

5 Zoom tools

The first two icons in this set zoom in and out of a sound file centred on the middle of the window. To zoom in to a more specific location, it's best to use the Zoom tool from the Toolbox. The other pair of icons are different sides of the same coin: the first zooms to a selection, allowing it to fill the window, while the second zooms (in or out) to fit the entire project into the window.

6 Track options

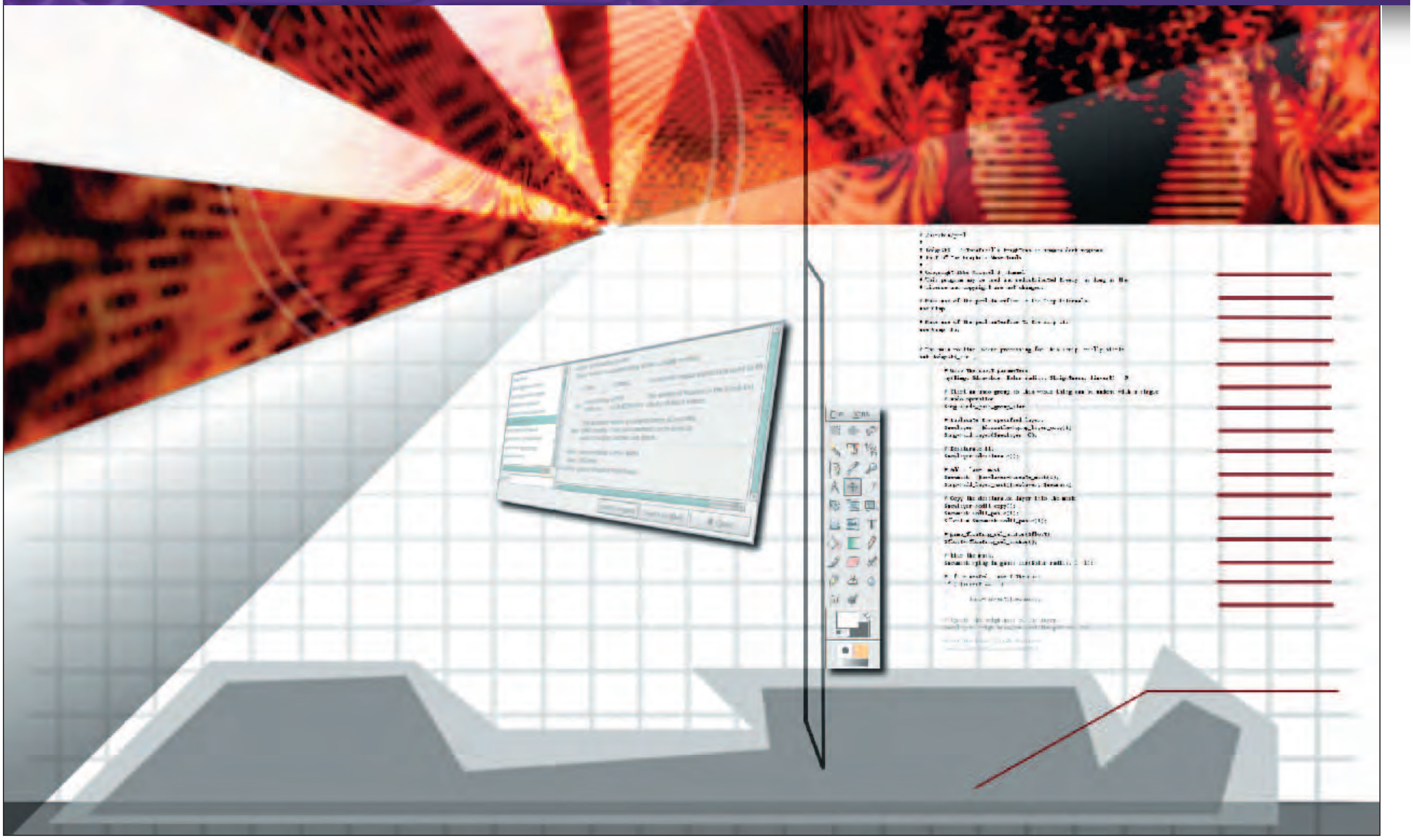
Mixing, at its most basic, involves changing the relative volume and pan levels of individual tracks to make a pleasing whole. In *Audacity,* real-time mixing – that is things that are adjustable during playback – are housed within the track options. This will turn a darker shade on the currently selected track. While the tools are a little primitive compared to *Sonar* or *Cubase*, they are adequate for making small alterations to volume, panning left or right and muting or soloing (listening in isolation) a track. At the top of the section is a drop down menu which provides tools to name tracks, change the way in which sound data is displayed, and alter the 'format' of the sample in terms of both bits and KHz. There are also options for splitting stereo tracks into two mono tracks, or combining mono channels to make a stereo track.

7 Waveform

Ever wondered what the word "salubrious" looked like? Grab a mic, hit record and see it appear as if by magic in *Audacity's* main window. Ranged along the top of the main window is a 'time bar' which allows us to see where we are within the track, so rather than being delimited by spatial measurements, it is broken up into hours, minutes and seconds. By default, playback always begins at either the start of the track or the beginning of a selection, to start elsewhere simply click on the location within the waveform and hit play (or record).

8 Selection

Just as a word processor allows selection of a range of data, so it is with *Audacity*. Click and drag across the waveform from left to right, or *vice versa*, to select a region. This can be adjusted by hovering the mouse over either extreme to redrag the point. Selections can be cut or copied for pasting or they can have effects applied without affecting the rest of the file. There are a number of feedback options that make selection more intuitive: the selected region is darker, as it the section of the 'time bar' at the top of the window, and on the status bar at the base of the screen the selection is defined in terms of the default unit (See Selection, above) for fine tuning.



GIMP PROGRAMMING

Writing GIMP Plug-ins in Perl

PART 1 The magic of *The GIMP* lies not just in what Wilbur can show you, but in what Wilbur can be taught.

On the CD and DVD

For the past year, the tutorials in this *GIMP* series have presented readers with creative ideas and solutions for use with *The GIMP*. Most tutorials followed a step-by-step format with accompanying images for clarification. This month, we're starting to take a more traditional approach to learning, to cover a more technical topic: *GIMP scripting with Perl*. This tutorial is part one of two on that subject.

Scripting is a means to automate common and/or repetitive tasks with *The GIMP*. Examples might be the generation of certain styles of logos, or processing directories of images in a uniform manner. *The GIMP* has a native scripting language called Script-FU, but this language has all the friendliness of a rabid dog. Script-FU is based on the Scheme language (related to the Lisp language, if you care about such things) and the lack of available documentation for that language – not to mention its technical orientation – makes it more suited to the tech-only crowd.

Perl, on the other hand, is a language that is widely understood. There is a wealth of printed material on the subject. Unlike Scheme, Perl is not a truly difficult language to learn. It provides both procedural and object-oriented APIs and doesn't abuse curly braces the way Script-FU does. That means that once you learn the *GIMP* specifics, you need only pick up a few extras of the language to start your own set of power tools. Scripts written in Perl for use with *The GIMP* are referred to as GIMP Perl scripts. GIMP Perl is also used to refer to the extensions to Perl that allow these scripts to work with GIMP.

This tutorial will be of most interest to those that do a lot of image processing, as opposed to those who just use *The GIMP* to create a holiday card or two. But even if you only use *The GIMP* to manage your scanned photos, scripting in *The GIMP* can make your graphics work far more productive. You need not be a Perl expert to follow along, but you should have at least a little background in programming: such as knowing what functions are, or how to do assignment and loop statements. Specifics to the Perl language can be found at the **http://perl.com** website.

Installation is an apt-snap

The Perl extension to *The GIMP* is not included with the application by default, and getting all the prerequisite software to build it manually can be difficult. Fortunately, this issue is moot if you use apt-get to update your *GIMP* packages. A quick search of some common RPM-based *apt-get* repositories shows the following packages:

- **gimp** – The GNU Image Manipulation Program.
- **gimp-data-extras** – Extra files for *The GIMP*.
- **gimp-devel** – *The GIMP* plug-in and extension development kit.
- **gimp-perl** – Perl extensions and plug-ins for *The GIMP*.

The names on the left are the names of the available *GIMP* packages. This list was generated using:

```
apt-cache search gimp
```

There are other *GIMP*-related packages, but for scripting with *The GIMP* this is all you need. To find out which packages you already have, simply use **rpm**:

```
rpm -qa | grep gimp
```

The results will look something like this:

```
gimp-1.2.3-16
gimp-devel-1.2.3-16
gimp-data-extras-1.2.0-8
```

In this case only the *gimp-perl* package is missing. To install the missing packages, you need to have *apt-get* installed. If you were able to run *apt-cache,* then you already have *apt-get*. If not, check out **www.freshrpms.net** to find a version that will work with your system. Once *apt-get* is installed, you can retrieve the missing packages:

```
apt-get install gimp-perl
```

Because *apt* uses RPM to install packages in system directories, this command will need to be run as the root user.

Meat and garnish, hold the taters.

The basic structure of a GIMP Perl script looks like this:

```
#!/usr/bin/perl

use Gimp;
use Gimp::Fu;

register();

sub my_script {
}

exit main;

```

The script is composed of two bits of meat and some garnish. The meat of the script consists of a call to the registration function and a subroutine that does the real work. Before we look at those, we need to look at the garnish – the first three lines of the script.

```
#!/usr/bin/perl
use Gimp;
use Gimp::Fu;
```

The first two lines are required for all GIMP Perl scripts. The third is used by nearly all GIMP Perl scripts except for those that require complex user interfaces. Line 1 tells the system to run this as a Perl script. Line 2 tells Perl to use a Perl module called **Gimp** that is the glue between scripts and *GIMP* functions. The third line is used to automate creation of the user interface. We'll look more in depth at what happens in the third line in next month's concluding episode of this two-part series.

When a GIMP-Perl script starts, the first thing it must do is let *The GIMP* know it's available and where it will be found in the menus. The script does this by calling a function provided by the GIMP Perl module (ie the module referenced by **use Gimp** at the top of the script) called **register()**.

Here's an example piece of code using the **register()** function with GIMP Perl:

```
register (
   "gfxoffset",
   "Find all layer offsets",
   "Find all layer offsets",
   "Michael J. Hammel",
   "Copyright 2004 Michael J. Hammel",
   "V1.0",
   "<Image>/Filters/GFXMuse/GFXOffsets",
   "*",
   [
     [ PF_STRING, "filename",
        "Where should the output be saved?",
        "/tmp/offsets.txt"
     ]
   ],
   \&OffsetGFX_Run
);
```

The call to **register()** includes many arguments. The first is the name of the script. GIMP Perl doesn't like it if you use uppercase for this name, so use the same name as the script but in all lowercase. This name is used internally by GIMP Perl and is not displayed to the user except possibly in the window title bar (depending on your window manager). The second argument is a short description while the third is a longer one. The only difference here is that the short description must be no more than one line long. The next two arguments are used to specify the author of the script and any copyright information. After that comes the version number.

Now we get to the truly important parts of the call to **register()**. The seventh argument is a menu path. The first part of this path is used to define where in the *GIMP* menus to place your script. Normally you'll put your scripts under the Image menu, usually in the Filters submenu though that is not required. To put the script in the Image menu, you prefix your menu path with **<Image>**. The only other option is **<Toolbox>**, which puts your script under one of the menus in the Toolbox. In the preceding example, the script is destined to be found in the Image menu under Filters>GFXMuse>GFXOffsets. 'GFXMuse' is a new submenu under Filters, which shows you that you can create »

GIMP 2.0
Perl vs Python

GIMP 2.0 does not include GIMP Perl by default. Python has been selected as the default language (beyond Script-FU). Users of *GIMP1.2* may wonder if learning GIMP Perl for that release is of any value now that *GIMP 2.0* is here and doesn't appear to actively support Perl. The answer is a resounding "Yes!" – it is of value.

GIMP Perl for *GIMP 2.0* will be released as its own add-on package at some point after *GIMP 2.0* is released. Don't be alarmed. This change in packaging is not unique to GIMP Perl. The *GAP* plug-ins (*GIMP Animation Plug-ins*) are also being moved to their own package separate from *The GIMP*. This has happened in the past as well, when *GIMP-Print* moved out on its own, not to mention the eldest brethren of *The GIMP's* offspring, the *GTK+* toolkit.

GIMP Python will not have nearly as much public support initially, since many script writers grew up on GIMP Perl. Additionally, it will take some time before *GIMP 2.0* is actively included in distributions. Perl is also not terribly different than Python in its syntax (the look of the language), which means learning one helps in learning the other. Finally, many example scripts already exist for GIMP Perl while few exist for Python. For these reasons, learning GIMP Perl now definitely has long-term value.

« your own submenu this way, perhaps grouping all your personal scripts in a single menu.

The eighth argument determines the types of images the script can work on. *The GIMP* understands RGB, Greyscale and Indexed images, either with or without transparency. An asterisk can be used as a wildcard. So **RBG*** would mean both RGB and RGB with transparency (aka RGBA) would be supported. In this example, all image types are supported. You can also use **GRAY**, and **INDEXED** (with a trailing **A for** transparency – the **A** means "includes an Alpha channel").

The next argument is a set of Perl arrays that define the layout of the window used to configure the script prior to running it. This section can get complex so for now we'll use this simple example that only provides a text entry field to query the user for where the output from this script (which will be ordinary text, not image data) should be saved. In next month's tutorial, we'll dive into this part of GIMP Perl a bit deeper.

Finally, the last argument to **register()** tells GIMP Perl the name of the function we wrote that does the actual work. After the user fills in the window presented by the script and clicks the 'OK' button, GIMP Perl will call the function named here (which is in our script) to do whatever work we require.

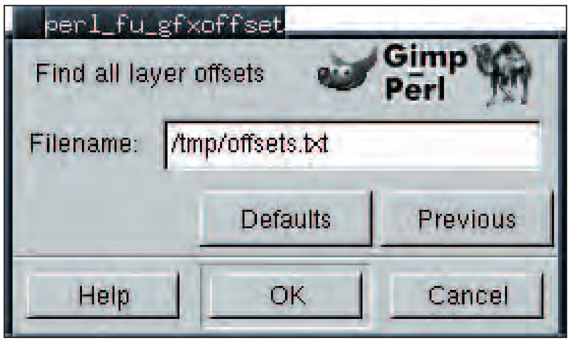
On to the code

GIMP Perl is so easy to learn that we can start by showing the complete code for a real-world example script now, leaving the details for later. At I company for whom I periodically do some work, a recent requirement arose to produce a set of images from a single photograph. The images produced would be overlaid onto a background, one image at a time, by a web-based backend graphics processor. To make this work easier, the photograph was first manually broken into its component pieces as individual layers, using a common naming scheme for the layers. Then, a script was written to output the offsets of the layers so that the layers, once saved to separate image files, could be overlaid properly by the backend.

There were actually two scripts for this project, one for saving each layer to a separate file, and one for creating a single file with all the layer offsets. The first we called GFXLayerSave.pl and the latter GFXOffsets.pl. Note that the work of creating the layers was manual – if the bits of the photograph to convert to layers were uniform in location and shape (which they were not), this too could have been automated. Both of these script files are available on our website at **www.linuxformat.co.uk/gimp/54.zip**

We start by looking at the smaller of the two scripts, GFXOffsets.pl:

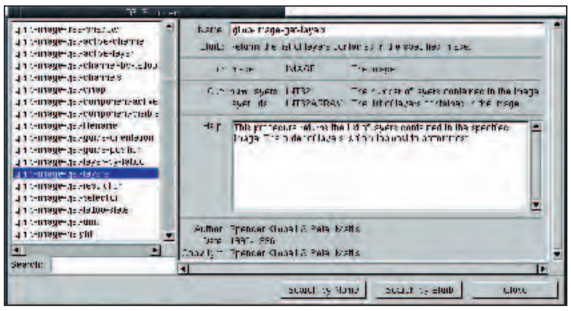
```
#!/usr/bin/perl
use Gimp;
use Gimp::Fu;

# Our function - get the list of layers and print their offsets to
a file.
sub OffsetGFX_Run {

   my($img, $drawable, $filename) = @_;

   my @layers = $img->get_layers();

   my $count = scalar(@layers);
   Gimp->progress_init("GFXLayerSave is working...");
   my $progress_increment = 1 / $count;
   my $progress = 0.0;

   open (FD, ">$filename") || die "GFXOffsets: Can't open
$filename\n";
   print FD " X    Y     Layername\n";
   print FD "---- ----   ------------------------------------------
----\n";
   foreach (@layers)
   {
      my ($xoffset, $yoffset) = $_->drawable_offsets();
      my $layername = $_->layer_get_name();
      printf(FD "%4d %4d   %s\n", $xoffset, $yoffset,
$layername);

      # Update the progress bar.
      $progress += $progress_increment;
      Gimp->progress_update ($progress);

   }
   close(FD);
   return();
}


# Register this script with the Gimp's PDB.
register (
   "gfxoffset",
   "Find all layer offsets",
   "Find all layer offsets",
   "Michael J. Hammel",
   "Copyright 2004 Michael J. Hammel",
   "V1.0",
   "<Image>/Filters/GFXMuse/GFXOffsets",
   "*",
   [
      [ PF_STRING, "filename",
         "Where should the output be saved?",
         "/tmp/offsets.txt"
      ]
   ],
   \&OffsetGFX_Run
);

exit main();
```



Filename: /tmp/offsets.txt

The user interface for GFXOffsets – simplicity is a built-in feature with GIMP Perl.

Here we see the **register()** function follows a function called **OffsetGFX_Run()**. Functions in Perl are prefixed by the keyword **sub**, which means 'subroutine', that happens to also be a synonym for 'function'. Note also that the order of the two



The DB Browser allows you to search the procedural database quickly and easily.

functions doesn't matter – GIMP Perl will find either function and call them when needed. The meat of our code is in our subroutine, **OffsetGFX_Run()**. Let's put in plain English what we're doing in our code:

1 Retrieve important values passed to us by GIMP Perl
2 Retrieve the set of layers in our image
3 Run through that set to
a retrieve the pixel offset of the current layer
b print the relevant data to a file

In addition, we also update the display to let the user know that something is happening. For images with a small number of layers, this isn't very important. For images with a very large number layers – as was the case for our selection of images – then this helps let the user know something is really happening and, just as important, when its done.

GIMP Perl passes in three values to our script: A value used to identify the image being worked on (*ie* the Canvas), a value used to identify a 'drawable' (which would be the currently active layer), and the name of the file as provided by the user via the window created using argument 9 of the **register()** function.

```
my($img, $drawable, $filename) = @_;
```

The **$img** value is important to us because we'll use it to find all the layers in the image on which we're working. The **$drawable** value isn't used here because we aren't interested in just the current active layer, we'll be iterating over all the layers in the image. But many (if not most) other GIMP Perl scripts work primarily on the active drawable and use this value extensively. The name of the file to save to, as provided by the user, is passed in as the **$filename** argument.

The subroutine specified in the **register()** call will be passed the values supplied by the user in the order they are listed in the **register()** function in argument 9. Again, we'll cover developing user interfaces in part two of this tutorial next issue.

Now that we have these values, we can start to do real work.

```
my @layers = $img->get_layers();
```

The next thing to be done is get that list of layers. The Perl language can be used either with a procedural (like the C language) or object oriented (like C++) interface. These two methods can even be mixed. Here we use the object methodology to ask the **$img** object to run the **get_layers()** method and save the layer ids to an array called **@layers**. How did we know about **get_layers()**? We used *The GIMP*'s Procedural Database.

The DB Browser (found in the Toolbox menus as Xtns>DB Browser) lists all functions that can be called by a GIMP Perl script. We'll cover this next month as well, but until then, you should know that anything that is prefixed with **gimp-image** in the procedural database can be accessed using the object-oriented syntax, minus that prefix. Alternatively, you can call **gimp_image_get_layers()** – the dashes in the database name are replaced with underscores. This is the procedural interface. Arguments provided differ between the procedural and object interface. The object interface doesn't need image or drawable ids depending on the class the function belongs to.

```
  my $count = scalar(@layers);
  Gimp->progress_init("GFXLayerSave is working...");
  my $progress_increment = 1 / $count;
my $progress = 0.0;
```

These lines are all used to provide feedback to the user. The first line assigns the count of layers to the variable **$count**. The next line prints a message at the bottom of the image window. The next two lines are used to initialise a progress bar – a horizontal scrollbar that fills in as we do our work. When the scrollbar runs end to end, we're done.

```
open (FD, ">$filename") || die "GFXOffsets: Can't open
$filename\n";
print FD " X    Y    Layername\n";
print FD "---- ----  ------------------------------------------
----\n";
```

The next three lines open the output file and print a header to it. If the file can't be opened for some reason, the script will exit (*ie* **die**) and print a message.

```
my ($xoffset, $yoffset) = $_->drawable_offsets();
my $layername = $_->layer_get_name();
printf(FD "%4d %4d   %s\n", $xoffset, $yoffset, $layername);
```

If the subroutine is the crispy, griddled meat of this scripting steak, then these next three lines must be the moist and bloody red meat at the centre. The first line grabs the offsets for our layer. The **foreach()** loop inside which we find these lines sets the **$_** variable to each layers object. So the call to **$_->drawable_offsets()** is how we get a single layers offsets.

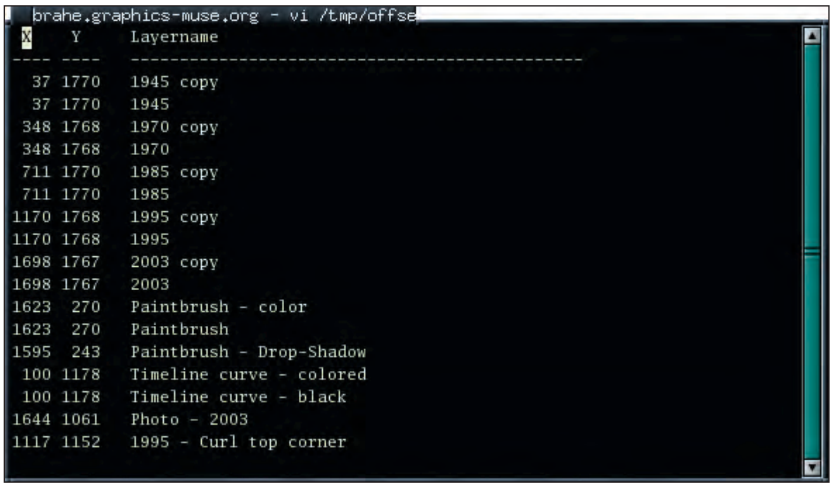
The next line grabs the layer name and that, along with the offsets, is then written to our output file.

```
$progress += $progress_increment;
Gimp->progress_update ($progress);
```

These lines print visible status changes to the progress bar, incrementing it an equal amount for each layer we process.

Where to go from here

The output from this script is very basic. It could be parsed manually or fed to a parser that feeds the graphics processor that builds dynamic images for the web. LXF



```
brahe.graphics-muse.org - vi /tmp/offse
X    Y    Layername
---- ----  -----------------------------------------------
  37 1770   1945 copy
  37 1770   1945
 348 1768   1970 copy
 348 1768   1970
 711 1770   1985 copy
 711 1770   1985
1170 1768   1995 copy
1170 1768   1995
1698 1767   2003 copy
1698 1767   2003
1623  270   Paintbrush - color
1623  270   Paintbrush
1595  243   Paintbrush - Drop-Shadow
 100 1178   Timeline curve - colored
 100 1178   Timeline curve - black
1644 1061   Photo - 2003
1117 1152   1995 - Curl top corner
```

The output is simple, but formatted for easy parsing.

NEXT MONTH

IN the concluding part of our examination of The GIMP and Perl, we'll look at the other script, GFXLayerSave.pl, which is a little more complex (but barely) and dig deeper into creating user interfaces and using the DB Browser to find the functions we need for our scripts.

CODING WITH SDL

Game programming

OK, so we missed an instalment last issue, but to make up for it this month, Paul Hudson brings you six whole pages of fishy goodness...

Calling all Trout fans! Drop your linen and start your grinning – *Trout Wars* is back this issue with six whole pages of marvellous scaly madness and a guide to actually doing cool things with your game. Last time we looked at the absolute basics: getting SDL installed, creating a window, and also clearing the screen. This issue we're going to be taking that a big step forward by creating a character that we can move around the screen and do things with. We'll also be looking at how text rendering works in SDL using the *SDL_ttf* library – more on that later. As you should recall, our game engine was encapsulated in the class **CTWGame**, and all our new functionality will also be encapsulated in classes. Not only is it easier to do this, but it makes for much easier to teach! So, without further ado: it's time to get some player action.

Making 1-up

As the old arcade machines used to say, *"1UP: Insert Coin to Play!"* and that's the point we're at currently. We have a flashing screen, which is pretty poor for gameplay, so what we're going to do is create a **CPlayer** class that draws an object on the screen that can be moved around using the cursor keys.

For this purpose, I dug out an old Windows program that I used to make games back in the mid-1990s called *The Games Factory* (*TGF*) – it allowed you to drag-and-drop graphics and actions into a game and string it all together quickly and easily. The guys who created are still going strong on their latest iteration of the package, and they were kind enough to grant permission to use a selection of their artwork and sounds under the GPL licence – you'll find them on your coverdisc as they are used.

The 'player' picture I chose is shown at the top of the opposite page, and it's a fairly generic-looking spacecraft. Perfect for *Trout Wars*, I think, but before we plug it into the game itself, we first need to define the basic player class.

Now, what should the most basic player be able to do? Although later on, the player should be able to shoot things, die, collect power ups *etc*, this is only the very beginning, which means we can encapsulate the entirety of our player's actions in just four points. A player should be able to:

- Move left, right, up, and down. These should work in combination to allow diagonals.
- Load and free its sprite
- Draw itself, and any objects it owns
- Store its current position

With that in mind, we can define the player in troutwars.h like this (put this before **CTWGame**):

```
class CPlayer {
public:
  int xpos;
  int ypos;
  CTWGame* game;
  SDL_Surface* sfcSpaceship;
  CPlayer(CTWGame* thegame);
  ~CPlayer();
  void Reset();
  void Draw();
  void MoveUp();
  void MoveDown();
  void MoveLeft();
  void MoveRight();
};
```

The first line says we're defining a new class called **CPlayer**, and the second says that everything that follows is **public** – available

QUIT FLASHING!

The last thing we did in the previous tutorial - in a somewhat desperate bid to make our efforts look unwasted – was to make the screen cycle through various colours. Be sure you undo that before starting this month's tutorial. In the ClearScreen() function you only want this:

```
SDL_FillRect(sfcScreen,
NULL, 0);
```

for access by objects outside of the **CPlayer** class. The first four lines after that define the basic variables of a **CPlayer** object: it needs an integer X and Y position (**xpos** and **ypos** respectively) that store the pixel position of the player respective to the top-left-hand corner of the screen, a pointer to the game object we created previously so that it can access shared data, and a pointer to a variable of type **SDL_Surface**. If you're not sure what pointers are, see the *Pointers to pointers to pointers* box overpage.

As explained last month, the **SDL_Surface** data structure is a graphical surface that can draw onto other surfaces and have other surfaces draw onto it. The easiest way to think of **SDL_Surface** is like a piece of paper: you can draw all you like on the paper with paints and inks, but you can also take that paper and stick it onto another, larger sheet of paper as part of a montage. Of course, you can even take the larger sheet and use it on an even larger surface, and the same goes with SDL surfaces. In the case of **CPlayer**, the **SDL_Surface** is **sfcSpaceship** – this will be where we store the spaceship picture.

Next up, we have eight functions, the first two of which are the constructor and destructor for the class. Note that to construct a **CPlayer**, you need to pass in a reference to the main **CTWGame** – this is because we'll be housing various standard functions (such as **DrawImage()** – more on that later) in the main game class, and we need an easy way to reference it from our other classes. This could be done with global functions, but in later issues it will become clear why it's important for the player (and other game objects) to have explicit knowledge of other parts of the game.

In order to have a **CPlayer** object in the game, you'll need to add this following line to the **CTWGame** class, just below the definition of **sfcScreen**:

```
CPlayer* Player;
```

In order to actually *create* the player, you'll need to put the following into the **CTWGame** constructor, **CTWGame()**:

```
Player = new CPlayer(this);
```

As always, anything we create also needs to have its memory freed, so in troutwars.h add this line just below **CTWGame()** (just above **void ClearScreen()**):

```
~CTWGame();
```

That's the destructor for **CTWGame()**. Back in troutwars.cpp, we need to write the actual ~CTWGame() function. Right now it only needs free the player object, so here's all you need:

```
CTWGame::~CTWGame() {
        delete Player;
}
```

Now, here's a minor conundrum. Our **CPlayer** class references **CTWGame**, and now our **CTWGame** class references **CPlayer**. Clearly one of the two needs to be defined first, but which? Fortunately, there's a way around this apparent dilemma, which is to forward declare the class – to tell the compiler "yes, I know **CPlayer** calls a class called **CTWGame** that I haven't told you about yet, but trust me – you'll get told later". Put this line of code before the **CPlayer** definition in troutwars.h:

```
class CTWGame;
```

Now, we need to fill in the **CPlayer** constructor, **CPlayer()**, the destructor, **~CPlayer()**, and the **Reset()** function. When the player is created, we need to store a reference to the game object, load its image, and set initial position values. The first two of those actions will be done in the **CPlayer()** function, with the latter being done in **Reset()**. When the player is destroyed, it should free up its **SDL_Surface** resource so that we have no memory leaks. If you've been following *LXF*'s *Flex and Bison* tutorials, you'll know the thinking there has been "leak as much memory as you like – the OS will clear it up when we're done", which is fine for a compiler – something that runs in just a few seconds at most. With a game, however, there's potential for the player to play for hours – or even days? – at a time, and if we let big resources leaks through their computer will soon grind to a halt.

Here's how that all looks:

```
CPlayer::CPlayer(CTWGame* thegame) {
  game = thegame;
  sfcSpaceship = game->LoadImage("spaceship.bmp");
  Reset();
}

CPlayer::~CPlayer() {
        SDL_FreeSurface(sfcSpaceship);
}

void CPlayer::Reset() {
  xpos = 20;
  ypos = (int)((SCREEN_HEIGHT / 2) - (sfcSpaceship->h / 2));
}
```

As you can see, **CPlayer()** calls the **LoadImage()** function of **CTWGame()** with the parameter **"spaceship.bmp"**. We'll be looking at how **LoadImage()** works soon enough, but for now just understand that it loads a bitmap file into an **SDL_Surface** and returns it. Also note that the game resource is being stored, and the **Reset()** function gets called. Destroying a player is as a simple as a call to **SDL_FreeSurface()** right now, because it only has that one resource. You should always destroy your **SDL_Surface** structures, with the only exception being the **SDL_Surface** returned by **SDL_SetVideoMode()** – this is freed automatically when **SDL_Quit** is called at exit.

The reset function is a little trickier. The first line simply sets the starting X position to 20, putting the top-left hand corner of the player's object 20 pixels away from the left-hand side of the screen. However, the second line is there to centre the player's spaceship vertically, and it does that by halving the screen height and subtracting half the height of the spaceship sprite. The **sfcSpaceship->h** line is the spaceship height – SDL automatically puts **->h** and **->w** into the **SDL_Surface** structure for us to tell us what the width and height of the picture are. In the first tutorial we saw that there's also a **->format** which contains information about the bitmap format the picture is in. Note that the reason **Reset()** is in here is so that we can **Reset()** when the player dies and needs to respawn – it will automatically reset its position, but eventually it will also reset health back to 100 per cent and the like.

Drawing the player is as simple as a call to another standard function we'll be looking at momentarily, called **DrawImage()**. Put this code into troutwars.cpp also:

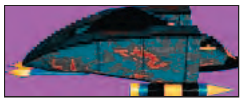
```
void CPlayer::Draw() {
  game->DrawImage(sfcSpaceship, xpos, ypos);
}
```

It might look like just calling **DrawImage()** is a cop-out, but, on the flip-side it does make writing code much, much easier!

Moving the ship

Thanks to the wonders of object-orientation, making the player move is almost laughably simple. As you've seen, we need to fill in the functions **MoveLeft()**, **MoveRight()**, **MoveUp(), and MoveDown()**. Naturally our player should be able to move

»



Our hero's spaceship - ignore the magenta, as that's just being used at this stage to define transparent areas.

OUR IMAGE SOURCE...

Although the source code is GPLed and therefore free for you to use in any shape or form you like, some of images and sounds used in our game were borrowed from *The Games Factory*, a Windows-based game development tool from Clickteam (www.clickteam.com). Clickteam has also created *Multimedia Fusion* and *Jamagic*, which are much more advanced game development programs, both of which are currently being ported to Linux as you read this. Clickteam very generously agreed to place the resources used in this game under the GPL as well, which means you are free to use them as you please as long as you mention that they are from Clickteam.com.

freely within the compounds of the screen, so we need to use our **SCREEN_WIDTH** and **SCREEN_HEIGHT** #defines from troutwars.h to limit the movement. With that in mind, here's the code to make the player move around:

```
void CPlayer::MoveUp() {
        if (ypos - 8 > 0) ypos -= 8;
}

void CPlayer::MoveDown() {
        if (ypos + sfcSpaceship->h + 8 < SCREEN_HEIGHT)
ypos += 8;
}

void CPlayer::MoveLeft() {
        if (xpos - 6 > 0) xpos -= 6;
}

void CPlayer::MoveRight() {
        if (xpos + sfcSpaceship->w + 6 < SCREEN_WIDTH)
xpos += 6;
}
```

I'll explain **MoveUp()** and **MoveRight()**, as the other two work the same way. **MoveUp()** checks to see whether the current player Y position minus 8 is greater than 0 – essentially, "has the player got space to move up?" If so, the player is moved up 8 pixels on the screen. You can change the number 8 if you want to make the player move faster. **MoveRight()** uses the X position of the player as well as the width of the player's sprite to check whether the player has enough room to move right 6 pixels. The width of the player's sprite is required because our X and Y positions are equal to the top-left hand corner of the player object – adding **sfcSpaceship->w**, the width of the player's sprite, gives us the right-hand corner, which is what we want for moving right as we want the right-hand edge of the player to be the limit of movement against the right-hand edge of the screen.

POINTERS TO POINTERS TO POINTERS

It only gets confusing when you think about it...

C and C++ both have the concept of 'pointers' – a data type that is used to point to another object. Many other language – including Basic and Java – do not have this concept, and often it causes problems. Let me give you a very simple example where a lack of pointers might cause problems, as that's the easiest way to see why pointers are helpful. Consider a space-based strategy game where there's a huge galaxy full of planets and star systems. Each planet has unique properties, such as population count, race of population *etc*, and takes up maybe 1KB of memory. Each star system has an array of planets it holds, and each galaxy has an array of the star systems it holds.

Now, what if we wanted to have each player in the game to have an array of all the systems they owned? And what if we wanted each race to have an array of all the systems that has that race as its primary population? We'd end up having each planet in potentially up to three places – once in the Planets array of a star system, once in the PlanetsOwned array of a player, and once in the PlanetsPopulated array of a star system.

With the data in three places, each individual planet now takes up 3KB rather than just the 1KB, and what happens when you *change* the information about a planet – do you update all three? What a waste of resources!

With pointers, all you have is a handle to an object without actually having the object itself. For example, if I have a class Foo that takes up 1MB of RAM, I could have 100 pointers to it and it would take up about 1.001MB of RAM, because only one copy of the original exists – the others are just re-using that same object. This saves a lot of RAM, as you can imagine, but it also saves copying (why pass the entire object into a function when you can just pass a pointer to it?), and also makes updating much faster. In the Foo example, all you'd have to do is change the single Foo object, and all the 100 pointers to it would also update as they all point to the changed object.

Languages without pointers usually use *references*, which are essentially the same thing. You can also use reference in C++, and they are the same as pointers except that they must always resolve to something. That is, you can have a null pointer – a pointer that points to nothing – but you cannot have a null reference.

Note that the reason a pointer is named a pointer is because it literally points to a position in RAM – for example: "the object you're looking for is at 0xc4d38a".

These four functions should be called if any of the cursor keys are pressed, so what we're going to do is run a check before the scene is drawn, to see what keys are being pressed. SDL makes this very easy through the **SDL_GetKeyState()** function, which you pass **NULL** to as its only parameter and it returns an array of key codes and whether or not they are pressed. So, we need to rewrite our **Play()** function to this:

```
void CTWGame::Play() {
  int done=0;
  Uint8* keys;
  while(done == 0) {
    SDL_Event* event = new SDL_Event;

    while ( SDL_PollEvent(event) ) {
      if ( event->type == SDL_QUIT ) done = 1;
    }

    keys = SDL_GetKeyState(NULL);
    if ( keys[SDLK_UP] ) { Player->MoveUp(); }
    if ( keys[SDLK_DOWN] ) { Player->MoveDown(); }
    if ( keys[SDLK_LEFT] ) { Player->MoveLeft(); }
    if ( keys[SDLK_RIGHT] ) { Player->MoveRight(); }
    DrawScene();
  }
}
```

Note that **Uint8*** called **keys**. This is a pointer to an unsigned integer – essentially an array of integers. This gets set to the return value of **SDL_GetKeyState()**, and therefore will contain a list of all the keys that have been pressed. The four lines before the call to **DrawScene()** are all checking to see which keys are pressed – **SDLK_UP** is the SDL constant for the Up cursor key, and the others match the other cursor keys. So, if **SDLK_DOWN** is set, the Down cursor key must be pressed, so **Player->MoveDown()** is called to make the player move down. As each of the keys are tested individually, this also allows diagonal movement – if both up and left are being held, the player will move diagonally up-left. Note that in the situation of a player pressing both the left and right keys, they will cancel each other out – the player will move left six pixels and move right six pixels every time the **Play()** loop goes around.

Loading and drawing

So, our player object now moves around the screen, which sounds like we finally have something worth looking at, but hold your horses - we haven't actually written any code yet to make the player's spacecraft get loaded up from a bitmap file! In the **CPlayer()** function we called **LoadImage()**, and in the **Draw()** player we called **DrawImage()** – both of these two are functions we're now going to be adding to the main **CTWGame** class, so add this code after the definition of the **Play()** function in the **CTWGame** class in troutwars.h:

```
SDL_Surface* LoadImage(char* filename);
void DrawImage(SDL_Surface *sfc, int x, int y);
```

Now, go back to troutwars.c to fill these functions in. Loading an image will be done by providing a filename (the **char*** part, which is a pointer to an array of characters) and returning a pointer to a surface. If you want to be C++ pure (something I'm not really fond of as I like the C file-handling functions far too much!) you might want to look into the **String** class. Anyway, back to the **LoadImage** function – to do the task, we'll be using three new functions: **SDL_LoadBMP()** to load the file, **SDL_SetColorKey()** to have SDL make one colour transparent, and **SDL_DisplayFormat()** to have SDL optimise the loaded surface by converting it to the format the user's screen is set to. Using that last function is a lot faster than having the screen in one format and the surface in another, and constantly having to dynamically convert during a draw operation.

In code, **LoadImage()** looks like this:

```
SDL_Surface* CTWGame::LoadImage(char* filename) {
  SDL_Surface *tmp1, *tmp2;
  tmp1 = SDL_LoadBMP(filename);
  SDL_SetColorKey(tmp1, SDL_SRCCOLORKEY|
SDL_RLEACCEL, SDL_MapRGB(tmp1->format, 255, 0, 255));
  tmp2 = SDL_DisplayFormat(tmp1);
  SDL_FreeSurface(tmp1);
  return tmp2;
}
```

The first line declares two temporary **SDL_Surfaces** that we'll be using to get everything right within the confines of the function. The second line calls **SDL_LoadBMP()**, which takes a filename as its only parameter and returns an **SDL_Surface** – if you really wanted to, the whole function could end there, because that's the bitmap loaded. Easy, huh?

As you can see from its picture on the opening pages of this tutorial, our player's spaceship has a magenta background to mark where transparency should lie, we need to use **SDL_SetColorKey()** also. This takes three parameters: the **SDL_Surface** to work with, the options to set, and the colour to use as the colour key. Note that the third parameter uses the **SDL_MapRGB()** function we looked at last month, which creates a colour in the format specified in parameter one using the RGB colours specified in parameters two, three, and four. As **255, 0, 255** is specified (full red, no green, full blue), **SDL_MapRGB()** will return magenta, and so the entire call tells SDL to set the transparent colour of the **tmp1** surface (which has the bitmap file loaded) to magenta. Note that there are two constants combined together in parameter two: **SDL_SRCCOLORKEY**, which is used to tell SDL we're setting the colour key, and **SDL_RLEACCEL**, which tells SDL to try to speed up drawing operations by using RLE acceleration on the surface.

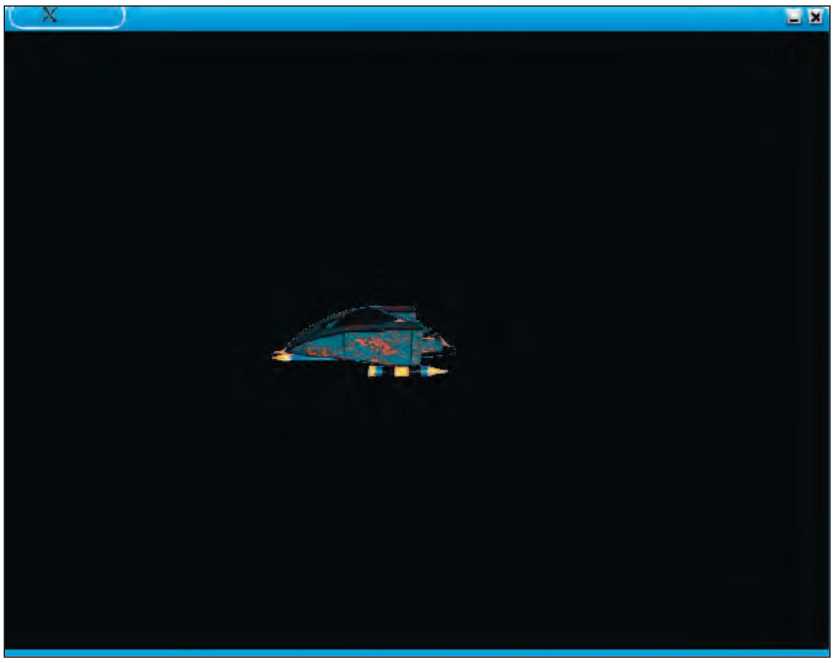
RLE (Run-Length Encoding) is a very simple compression algorithm that stores colour information as a colour followed by the number of times it is repeated. So, for example:

```
RED RED RED RED RED RED BLUE BLUE BLUE RED RED
```

could be replaced by...

```
Red x 6, Blue x 3, Red x 2.
```

As you can imagine, RLE encoding is not suitable for images that don't have long runs of similar colour, but I wouldn't worry about that if I were you – adding **SDL_RLEACCEL** is enough to have SDL use RLE compression automatically, which will speed up most of your image-related functions. Moving on, the next line calls **SDL_DisplayFormat()** to convert the **SDL_Surface** to the current screen format – note that we pass in **tmp1**, which holds the existing image, and assign the result to **tmp2**. At this point, **tmp1** is now useless, so we call **SDL_FreeSurface()** on it, and return **tmp2** back to the calling function. Note that **tmp2** *is not freed*. This is crucial: if we freed **tmp2**, then the calling function would receive nothing back – we leave it up to the caller to free the surface when it is done with it.



It's a spaceship, in a big, black screen. It's pretty sad that this is exciting stuff compared to our screenshot last issue!

AFTER THE TROUT

LXF reader Ed Mack wrote in to tell us that he'd like _Trout Wars_ to become a real-time strategy game. Sadly, that's not the plan we have – instead, we're working towards a scrolling shooter somewhat similar to _Xenon 2_ from the old Amiga days. Except, perhaps, without the über-cool _Xenon 2_ 'Megablast' remix! The reason for this is because right now we're more concerned with teaching pure SDL – yes, there will be some elements of how to make the _game_ work, but our focus is all about SDL, for now at least. If there is sufficient interest in further SDL games such as strategy games – real-time or otherwise – then it's certainly something we can look into. Perhaps it's a decent poll idea for the website: Do you want to see more 2D, some 3D, or specific game types?

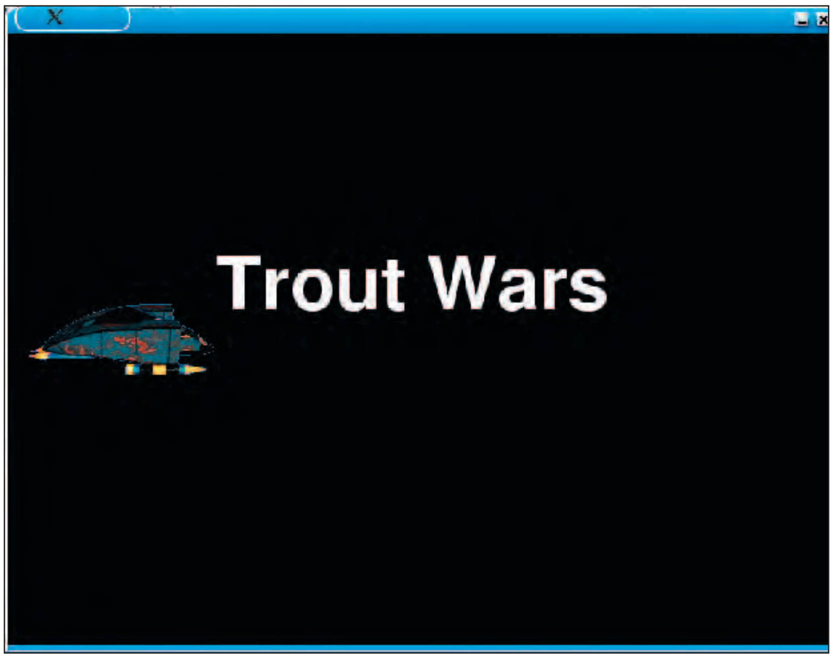
Keep in mind that many games are effective mixes of the two – if you ever played the Windows game _Age of Empires 2_ (my wife's personal favourite!), you may have noticed that the arrows fired by archers fly in very smooth arcs towards their targets. If the entire game were done in an isometric viewpoint, this would actually be very difficult to program. Instead, by having the game written in 3D but with a fixed viewpoint that simply makes it _appear_ to be isometric 2D, firing arrows is suddenly a cinch – things that move in 3D are best _calculated_ in 3D.

Send your ideas to the usual address – our vote goes to a remake of another Amiga classic, _Syndicate_ – just in 3D!

So, that's image loading done – we can now specify a bitmap file to load and have it imported onto a surface. The next step is drawing a surface. Again, this is going to be a standard function stored in the **CTWGame** class, as this saves us having to have the same function in every object that has to draw things. The code for this will be using a new data structure, **SDL_rect** (a rectangle), as well as a new function, **SDL_BlitSurface()**, which copies one surface to another. Here we go:

```
void CTWGame::DrawImage(SDL_Surface *sfc, int x, int y) {
  SDL_Rect dest;
  dest.x = x;
  dest.y = y;
  SDL_BlitSurface(sfc, NULL, sfcScreen, &dest);
}
```

Now, as you can see, this function takes an **SDL_Surface** to draw, »



This time the spaceship flies cunningly under the text "Trout Wars." We've seen finished games that are worse than this!

as well as an X and Y integer for position information – again, this is pixels on the screen relative to the top-left-hand corner. So, the first three lines set up the position for drawing: **SDL_Rect** stores a height (**h**), width (**w**), X position (**x**), and Y position (**y**), which allows you to accurately position a given rectangle on the screen. This is used to decide where the source surface should be drawn on the destination surface – in the case of our player, where it should be drawn on screen. If no height and width are specified, as above, SDL places no limits, essentially drawing the entire thing.

The function call that does the magic is **SDL_BlitSurface()**, which takes a source surface as the first parameter, a rectangle of the data to take from the source as the second parameter, the destination surface as the third parameter, and the destination rectangle as the last parameter. We have already looked at parameters one, three, and four, but parameter two is new – it allows you to use only part of the source surface for copying to the destination, which can be helpful now and then, but not for our current plan. If we get sufficient support to move onto 3D games once *Trout Wars* is complete, you'll see that using a source rectangle for textures is actually quite common – very often a 3D model has one skin that covers the entire mesh, which means you need to juggle the numbers just right to make sure the right source rectangles are used for the right parts of the mesh.

Praise the Lord!

OK, that was a huge chunk of code, and, as always (pffft!), I think the code should be self-explanatory. However, at this point, the code should now compile and work because we've covered everything to make the player work. So, using the same commands as last time, build and run your game! Note that you will need to copy spaceship.bmp from the coverdisc to your working directory or replace it with your own magenta-background player sprite, and I suggest you use the diff command to compare my source files against yours just in case I missed something while rambling.

You should see a black screen with the spaceship in the corner. If not, go away and sulk for a while then, post-sulking, try and track down the problem – check the diffs between your files and mine for obvious reasons. All being well, you should be able to move the spaceship around the screen – and be stopped at the edges – by using the cursor keys.

Naturally, we want the player to actually do things, but this is far enough for this month – we're going to swiftly move on to how to use text in SDL using the **SDL_ttf** library. All being well we'll be able to continue on making our player class more interesting in the issue-after-next (LXF56).

Fonts are fun!

All the fonts in SDL are done using the **SDL_ttf** library, which provides core functionality for loading fonts and turning them into surfaces. We will be using fonts extensively later to display game information such as score and hit accuracy, and also to show results such as level statistics and a high-score board. For now, we're just going to look at how the technicality of it all works, with the aim of having the words "Trout Wars" displayed on the backdrop of our otherwise black screen.

To get started, we need to include SDL_ttf.h file in troutwars.h, so add this line beneath the include for SDL.h:

```
#include <SDL/SDL_ttf.h>
```

If you followed the detailed install instructions in the original *Trout Wars* instalment, you should already have **SDL_ttf** installed. If you get errors that SDL_ttf.h can't be found during compilation, go back and check you did everything properly!

To work with fonts, we need to have two variables: a **TTF_Font** structure that holds the actual font shapes and style information, and an **SDL_Surface** that holds a rendering of our text based on the **TTF_Font**. So, what we're going to do (for now at least) is have two new variables inside the **CTWGame** class: one to hold a font, and one to hold a surface for the words "Trout Wars". To make this change, edit troutwars.h and add this after the definition of the **CPlayer** in the **CTWGame** class:

```
TTF_Font* fntMain;
SDL_Surface* sfcTxtMain;
```

Making fonts work is remarkably easy – much like the rest of SDL, really. In the constructor for **CTWGame()**, where **SDL_Init()** and **SDL_SetVideoMode()** are called, we need to add a few lines to initialise the **SDL_ttf** library. The **TTF_Init()** function initialises it, the **TTF_OpenFont()** function (which takes a TrueType font and a point size as its two parameters) opens the font and returns a **TTF_Font** value for it, and we also need to call **atexit()** again so that **TTF_Quit()**, the **SDL_ttf** shutdown function, is automatically called when the program shuts down along with the main **SDL_Quit()** function.

So, in code, that becomes this:

```
TTF_Init();
fntMain = TTF_OpenFont("freesans.ttf", 72);
atexit(TTF_Quit);
```

You'll need to put that after the existing **atexit()** call for **SDL_Quit()**. Note that that will set **fntMain** to be a 72-point font in the style of freesans.ttf. You can juggle the font size around as you want in order to get some variation on your screen, but note that actual font colouring is *not* done at this stage – the **TTF_Font** is just the style and shapes of the letters. The freesans.ttf file is the Free Software Foundation's Free Sans Bold – a very nice, very clear font that's easy on the eye and is perfect for text in our game. It's on your coverdisc, and is free to use and distribute (naturally).

GET THE CODE

Due to a production screw-up on my behalf, the *Trout Wars* code on your discs this month is actually the code for the tutorial the month after next! As I put it on there by accident, I can make no guarantees to its quality, but you're welcome to read it through to see what's coming up. Please do not write in if it crashes (it *will* crash).

The code for this issue's tutorial is available for you to download at www.linuxformat.co.uk/sdl/54.zip.

In order to keep the resources cleaned up, you'll need to also add this line to the **CTWGame** destructor, ~**CTWGame()**:

```
TTF_CloseFont(fntMain);
```

With **SDL_ttf** initialised, we can now create the text we'll be drawing on the screen. What we don't want to do is create the text each time **DrawScene()** is called because we just want to display a static string. For data that changes often – such as a timer that ticks down as seconds go by – you might want to break this rule so that the re-render the text each time **DrawScene()** is called. For static text, you should create it at the start of the game and delete it at the end, so we're going to set up the surface with our text in in the **CTWGame()** constructor and delete it in ~**CTWGame()**. Rendering text can be done with several different functions, but the two key functions are **TTF_RenderText_Solid()** and **TTF_RenderText_Blended()** – the former renders standard text, whereas the latter uses alpha shading to smooth the edges of the font and make it look nicer against any backdrop we choose. **TTF_RenderText_Blended()** does take a smidgen longer to perform, but the extra quality is well worth it – that's what we'll be using here.

TTF_RenderText_Blended() takes three parameters: the **TTF_Font** to use for the text, the text to render, and the colour you want it. The colour must be specified in the **SDL_Color** type, so we'll be need to define what we mean by white before it's used. We can create the rendered text by adding these next two lines to **CTWGame()**, just after the call to **SDL_SetVideoMode()**:

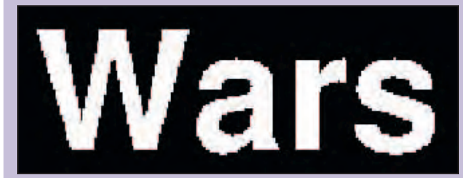
```

SDL_Color clrWhite = { 255, 255, 255 };
this->sfcTxtMain = TTF_RenderText_Blended(fntMain, "Trout
Wars", clrWhite);

```

That's the text rendered in white – the three **255** figures are RGB values. That's a bit of a temporary hack – ideally we want to have our colours defined somewhere that any object can get to easily, but this kludge works fine for now as we're not doing anything else with colour just yet. In the ~**CTWGame()** destructor function, we need to free the rendered text's **SDL_Surface** as well as free up the **TTF_Font** resource, so change it to this:

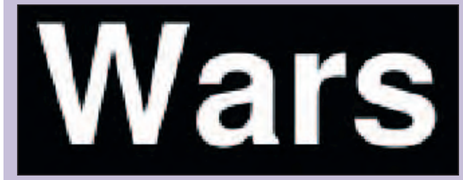
```

CTWGame::~CTWGame() {
   delete Player;
   TTF_CloseFont(fntMain);
   SDL_FreeSurface(sfcTxtMain);
}

```

ANTI-ALIASED TEXT: WORTH THE EFFORT

The two pictures below show aliased and anti-aliased text, with the one on the top being aliased [created with TTF_RenderText_Solid()] and the one on the bottom being anti-aliased [created with TTF_RenderText_Blended()].



'Jaggies' are prevalent in this blown-up picture, although even when shown at normal size, it's still very obvious to the naked eye.



The jaggies have disappeared now, as extra time was taken to smooth the edges off nicely – it makes quite a difference, doesn't it?

The last step is to actually draw the font once it is rendered to a surface, and this is accomplished by treating it like any surface – using **DrawImage()** and giving it X and Y co-ordinates.

```

void CTWGame::DrawScene() {
  ClearScreen();
  this->Player->Draw();
  DrawImage(sfcTxtMain, 200, 200);
  SDL_Flip(sfcScreen);
}

```

If you want your player to fly over the text as opposed to under it, move the **sfcTxtMain DrawImage()** line above the call to **Player->Draw()**. Similarly, you can try specifying different co-ordinates – perhaps the X might be **(SCREEN_WIDTH - this->player->xpos)** and the Y might be **(SCREEN_WIDTH - this->player->ypos)** to make the text move inversely to the player..

That's almost all

Six pages is an awful lot of *Trout Wars* for anyone to take, even if you already consider yourself a seasoned games programmer. However, I hope you'll agree the space has been used well – we've managed to cover heaps of stuff in quite a small space here, enough that if you really wanted to you could be spending the time between now and you receiving next issue adding some extra new features to your game.

To recap, adding a player to our game was actually quite a straightforward exercise, because we're keeping everything neatly segregated in individual classes. This programming style is going to continue for the rest of this series, so please get used to it! By having the player trapped inside the window you have also implemented – whether you realised it or not – simple collision detection. That is, when the player 'collides' with the screen edge, it gets stopped. When you think about it, it's only a short jump from here to collision detection with other objects – compare the X, Y, width, and height values of two objects, and if they overlap you've got a collision. We'll probably be looking at this the issue-after-next (*LXF56*), but hopefully this will give you enough of an idea to try it out for yourself.

Fonts, as you've seen, are very easy using the **SDL_ttf** library. For maximum speed, particularly in 3D games where fonts are always a hassle, many programmers use simple fixed-width fonts where the entire alphabet and symbol set is one large texture, and individual characters are drawn using specific source rectangles. Although this is very fast, it's also painful to use – unless speed is a real issue for you, I'd stick with the **SDL_ttf** library as far as you can.

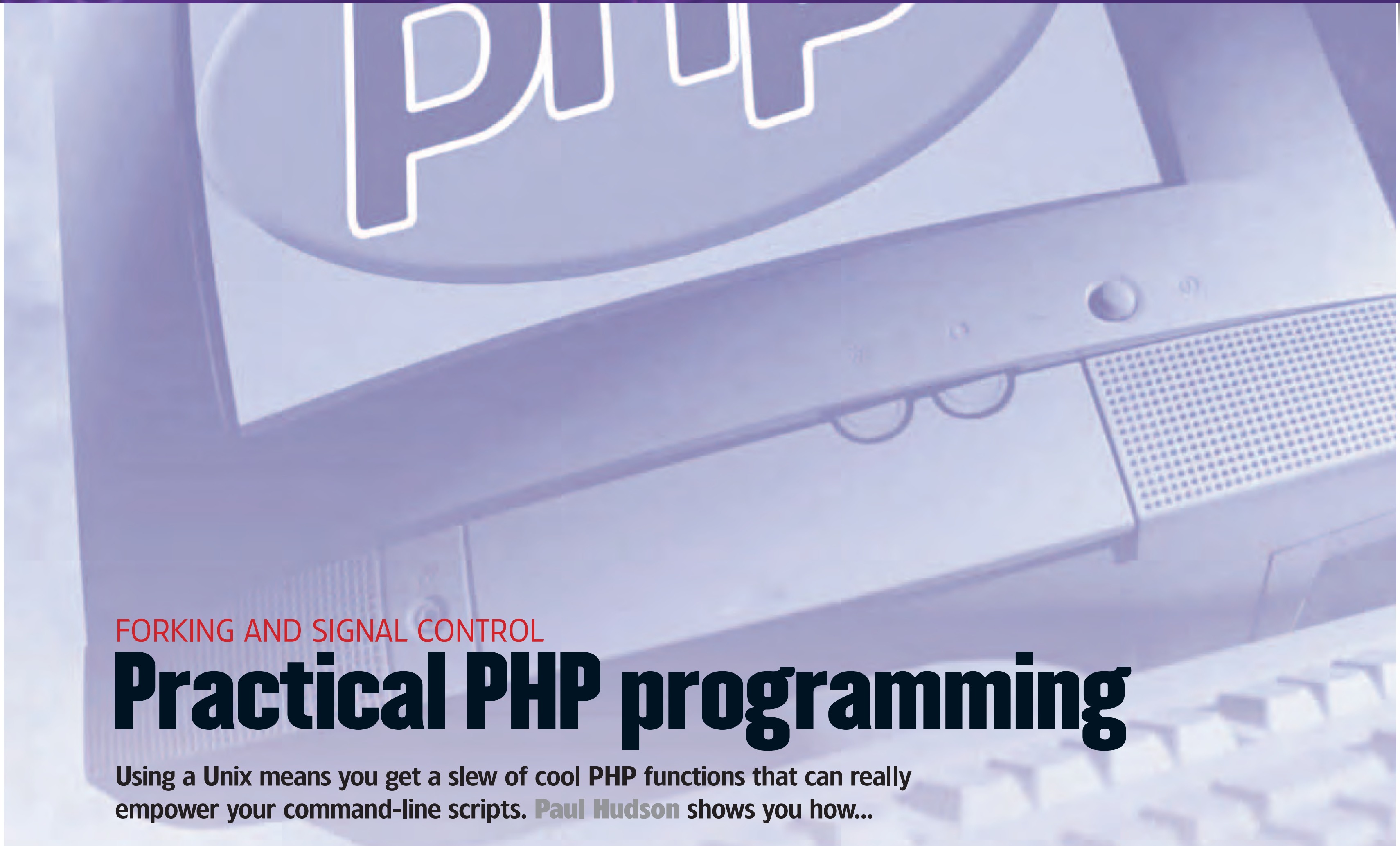
Hopefully we shan't miss any more *Trout Wars* tutorials, and we certainly have big plans for the future. So, until next month – when we'll be looking at how to add music and sound effects to the game – happy coding! LXF

SUPPORT SDL

SDL is wholly supported by funding from the community, so why donate just a little money to help pay for development costs? The SDL homepage is at www.libsdl.org and there are instructions online for how to donate.

NEXT MONTH

In this six-page tutorial we now have the very beginnings of a real, playable game. The player sprite should have been loaded, replete with transparent colour (go ahead and try commenting out the SDL_SetColorKey line to see the difference!) and we also have player movement. Furthermore, we've covered how to render text in whatever colour and size you want. Still, this is only the very tip of the iceberg from a programmatic point of view – next month, we're going to start making things more interesting by adding a better backdrop, some sounds and music, and perhaps even some *enemies* to shoot. Isn't games programming fun?



FORKING AND SIGNAL CONTROL

Practical PHP programming

Using a Unix means you get a slew of cool PHP functions that can really empower your command-line scripts. Paul Hudson shows you how...

Command-line PHP programming brings with it a number of cool possibilities that are either not possible in the *Apache* module or just not sensible – that's part of the fun. As CLI SAPI programming is gaining more and more use in the community, people are starting to experiment with its potential and finding it's capable of some really cool stuff. In order to give you a headstart, we're going to be looking at two particularly cool ways to use PHP on the command-line: process control and the POSIX functions, so you can start experimenting yourself.

Spawning children

To get process control working on your PHP machine, you need to configure it with **--enable-pcntl**, rebuild it, and re-install it.

Once that's done, you can get started writing process control scripts. The first thing we're going to do is look at the **pcntl_fork()** function, which takes no parameters and returns an integer. If you get back **-1**, the forking failed – we couldn't create a new process. If **0** is returned, the process has now been forked and you're now in the child process. If anything else is returned, the process has now been forked and you're now in the parent – the return value is the PID (process ID) of the child that was created.

Now, this is the point where people usually get confused – if **0** is returned, you're in the child, if anything larger than **0** is returned, you're in the parent. Yes, you *can* be in two places at once now because we have multiple processes. When you fork, what happens is that a complete copy of the existing process is made – including the values of all the variables previously used – and also the current position the PHP script is up to in terms of execution; and that copy becomes the child process. As a result, the child process starts executing from the line after the call to **pcntl_fork()**.

As always, this makes more sense when you see it in code, so try this one out:

```
<?php
  $pid = pcntl_fork();
  if ($pid != -1) {
    if ($pid) {
      print "In the parent: child PID is $pid\n";
    } else {
      print "In the child\n";
    }
  } else {
    echo "Fork failed!\n";
  }
?>
```

When run, you should get something like this as output:

```
In the child
In the parent: child PID is 8616
```

But then running it again you might get this instead:

```
In the parent: child PID is 8618
In the child
```

As you can see, the position of the parent and child output is pretty much variable – it's down to whichever of them hits its echo statement first, and that is down to how much time they are given by the OS. The scheduler in Linux (particularly in 2.6) is very smart, so it's not really worth trying to outguess it unless you have a very firm grasp of kernel semantics. Heck, even then don't rely on it – the scheduler may change at any time, and you don't want to get into the mess of relying on exact timings.

Now, try editing the script to this:

```
<?php
  $pid = pcntl_fork();
  if ($pid != -1) {
    if ($pid) {
      print “In the parent: child PID is $pid\n”;
    } else {
      sleep(1);
      print “In the child\n”;
    }
  } else {
    echo “Fork failed!\n”;
  }
?>
```

Running that back, you'll now always find the parent's message is printed out before the child's, because as the child process is sleeping the parent continues on. However, there's more of interest in that script – here's the output I get on my machine:

```
[paul@hud-lxf lxf54]$ php fork2.php
In the parent: child PID is 8666
[paul@hud-lxf lxf54]$ In the child
```

Note that we get our command prompt back immediately after the parent's message is printed out, and that the child's message is printed after the command prompt? The reason for this is because the command prompt waits for the parent process to terminate before it re-appears, and the parent process terminates before the child process has printed its message out. So, the parent prints out its message, terminates, then the child process wakes up (after the command prompt re-appears) and prints out another message to the screen. This should amply demonstrate that even after the parent process is dead, the children live on.

Quadruplets

Forking one child is a cinch, as you can see, but how do you fork more than one? The problem here is that you need to be very careful with what process does the spawning – if you slip up, you might end up in a recursive spawning loop where hundreds of thousands of processes try to spawn and will probably drag your machine down pretty sharpish! The easiest way to avoid problems is to have the parent process do all the spawning, and this can be done using a simple loop. Try this out for size:

```
<?php
  for ($i = 0; $i < 4; ++$i) {
    $pid = pcntl_fork();
    if ($pid != -1) {
      if ($pid) {
        print “In the parent: child PID is $pid\n”;
      } else {
        sleep(1);
        ++$i;
        print “In child $i\n”;
        exit;
      }
    } else {
      echo “Fork failed!\n”;
    }
  }
?>
```

This time, the code loops around four times in total, but note there are three minor tweaks in the child code. The first two are that we increment **$i** by one in the child so that each child has a number from **1** to **4** and it prints it out, and the third is that **exit** is called immediately after the child process prints out its message. This last part is absolutely crucial: if the child process isn't terminated, it will go back through the loop itself and spawn more children, which will spawn more children, and more children. It's not infinite, though, because each child inherits the loop from its parent as well as the value of **$i**, so the loop doesn't last too long. If the **++$i** line was excluded, the loop would last a little longer, but would still not be infinite – that's a *good* thing.

So, if you want to create four child processes at once, the easiest way to do it is by using a tightly controlled loop – unless you're really confident of your own abilities, it's almost certainly best not to try to spawn processes from spawned processes.

Waiting up for the kids

As we've seen, child processes are able outlive their parents, but what if you don't want that to happen? Well, parent processes are able to wait for children to finish through the **pcntl_waitpid()** function, which takes a process ID to wait for and a reference where the exit status can be placed. The exit status of a child is made up of an integer value returned back as well as how the child exited. To extract the actual return value from a status code, you need to use the **pcntl_wexitstatus()** function, which takes a status code as its only parameter and returns the return value of the child. So, with that in mind we can rewrite our original forking script like this:

```
<?php
  $pid = pcntl_fork();
  if ($pid != -1) {
    if ($pid) {
      print “In the parent: child PID is $pid\n”;

      pcntl_waitpid($pid, &$status);
      echo “Back in parent\n”;
      echo “Child exited with ”, pcntl_wexitstatus($status), “\n”;
    } else {
      sleep(1);
      print “In the child\n”;
      exit(19);
    }
  } else {
    echo “Fork failed!\n”;
  }
?>
```

Now, note that we are waiting for **$pid** and storing its return value in **$status**. When the parent hits that line, it will pause indefinitely until the child with that **PID** has returned – if has already returned, it will continue immediately. The child scripts waits for one second, prints out a message, then returns the value **19** (totally arbitrary, just to demonstrate how it works) and terminates. The parent, which was waiting for this child to terminate, wakes up again and stores the child's exit status in **$status**. Then another message is printed out (**“back in parent”**) and its finishes off by printing out the child's return value using **pcntl_wexitstatus()**. On my machine, here's the output:

```
paul@hud-lxf lxf54]$ php fork3.php
In the parent: child PID is 8894
In the child
Back in parent
Child exited with 19
```

As you can see, the child executed fully while the parent was waiting, then returned data back successfully. Things get a little more difficult when working with multiple processes, as you will want to wait for all child processes to terminate before the parent can terminate. The easiest way to solve this problem is to count back from the loop

»

used to spawn the children and use **-1** as the first parameter to **pcntl_waitpid()** – that causes PHP to wait for any child to return, as opposed to specific PIDs. Take a look at this script:

```
<?php
  for ($i = 0; $i < 4; ++$i) {
    $pid = pcntl_fork();
    if ($pid != -1) {
      if ($pid) {
        print "In the parent: child PID is $pid\n";
      } else {
        sleep(1);
        ++$i;
        print "In child $i\n";
        exit($i);
      }
    } else {
      echo "Fork failed!\n";
    }
  }

  if ($pid) {
    while ($i > 0) {
      pcntl_waitpid(-1, &$status);
      $val = pcntl_wexitstatus($status);
      echo "Child $val returned\n";
      --$i;
    }
    echo "Parent complete!\n";
  }
?>
```

Note first that the children now return their child number, but the main block of new code is at the bottom. First we need to check **$pid**, as we only want to wait if we have actually spawned any processes. Then, the parent enters a loop: while **$i** is greater than zero, wait for children. As **$i** has already been incremented to the number of children we have as part of the loop, this essentially reverse the loop and waits for all children to complete. Note that the **-1** is in there for **pcntl_waitpid()**, which means the loop will wait for any children and print out its value in the order it arrive. As a result, and again this shows just how unusual the scheduler may appear to be, here's what I got out of a test run with this code:

```
[paul@hud-lxf lxf54]$ php fork.php
In the parent: child PID is 8963
In the parent: child PID is 8964
In the parent: child PID is 8965
In the parent: child PID is 8966
In child 1
In child 2
In child 3
Child 3 returned
Child 2 returned
Child 1 returned
In child 4
Child 4 returned
Parent complete!
```

As you can see, children **1**, **2**, and **3** had all executed and returned even before child **4** was able to print out its message.

Smoke signals

A key aspect to IPC is the ability to handle signals sent from other processes, and this is fully available in the CLI SAPI. The primary function here is **pcntl_signal()**, which installs a callback function for a given signal, and takes two parameters: the signal to catch, and the function to call when that signal is received.

Now, if you're unsure what signals are, let me briefly explain. When you press **Ctrl-C** to halt a program, what is it that actually makes it halt? How about when you type **killall php** – what actually makes the PHP scripts halt? The answer is signals, and there are quit a few types. The most popular are **SIGINT** (interrupt), **SIGHUP** (hang-up), **SIGTERM** (terminate cleanly), and **SIGKILL** (terminate immediately, clean or otherwise), but there are many others. To give you an idea how the various signals are generated, try out this script:

```
<?php
  declare(ticks = 1);

  function signal_handler($signal) {
    switch ($signal) {
      case SIGTERM:
        echo "Caught SIGTERM\n";
        break;

      case SIGQUIT:
        echo "Caught SIGQUIT\n";
        break;

      case SIGINT:
        echo "Caught SIGINT\n";
        break;
    }
  }
```

WHAT ARE PROCESSES?

The who, what, where, and why of your OS's lifeblood

One process can be thought of as a single unique running instance of a binary file or script. That is, while the vim binary usually exists only once on your computer, you can launch it many times and have many vim processes. Each of these processes is wholly standalone – you can quit one, kill one, halt or background one, and the others remain unaffected. Linux (as well as any half-way decent OS) automatically does process scheduling so that each process gets a slice of the CPU's time (a *timeslice*, no less) and everyone's happy.

In some ways, processes can be thought of like threads, however they are quite different beasts once you start getting deep. In Linux, and also most other Unixes, creating a process is very, very quick. This needs to be the case because so much of Unix is about inter-process communication (IPC) – we create a process of *ps*, pass it through *grep*, then finally through *wc* to find out how many instances of one program is running. Running three programs as if they were one combined program is absolutely the norm in Unix, so it would be terrible for performance if starting a process was slow. However, on other OSes (most notably Windows) start a process *isn't* very fast, and so programs rely on threads, which are essentially virtual processes, to mimic processes.

The whole point of having processes (or threads) is that they are functionally separate parts of a program. If you take an average PHP script, it runs from top to bottom (with occasional interludes when calling functions), but it always, always, *always* only executes one line at a time. However, when you have two scripts running, you have two processes each doing their own thing, and the OS automatically balances the two so that they both appear to be running at the same time. Of course, if you have a single CPU machine, the two processes aren't running at the same *exact* time because your CPU can only do one thing at once; but if you have a multi-CPU server, it is able to run more than one process at once, which allows your two scripts to run side-by-side.

For instance, consider that the *LXFBench 2004* benchmark has some elements written in PHP. If that were to run entirely in one process, a 4-way 3.2GHz Xeon server would return the same score as a 1-way 3.2GHz Xeon, which is of course unrealistic. However, by spawning processes, the OS automatically balances the processes out over the other CPUs, and so it returns a score inline with the number of CPUs.

Hopefully, you should now have a firm grasp of what processes are, and also why you'd want to use them. However, if not, here's a quick nutshell-type roundup: *processes let you run more than script at once, and create these child scripts from inside another script.*

```
  pcntl_signal(SIGTERM, “signal_handler”);
  pcntl_signal(SIGQUIT, “signal_handler”);
  pcntl_signal(SIGINT, “signal_handler”);

  while (1) {

  }
?>
```

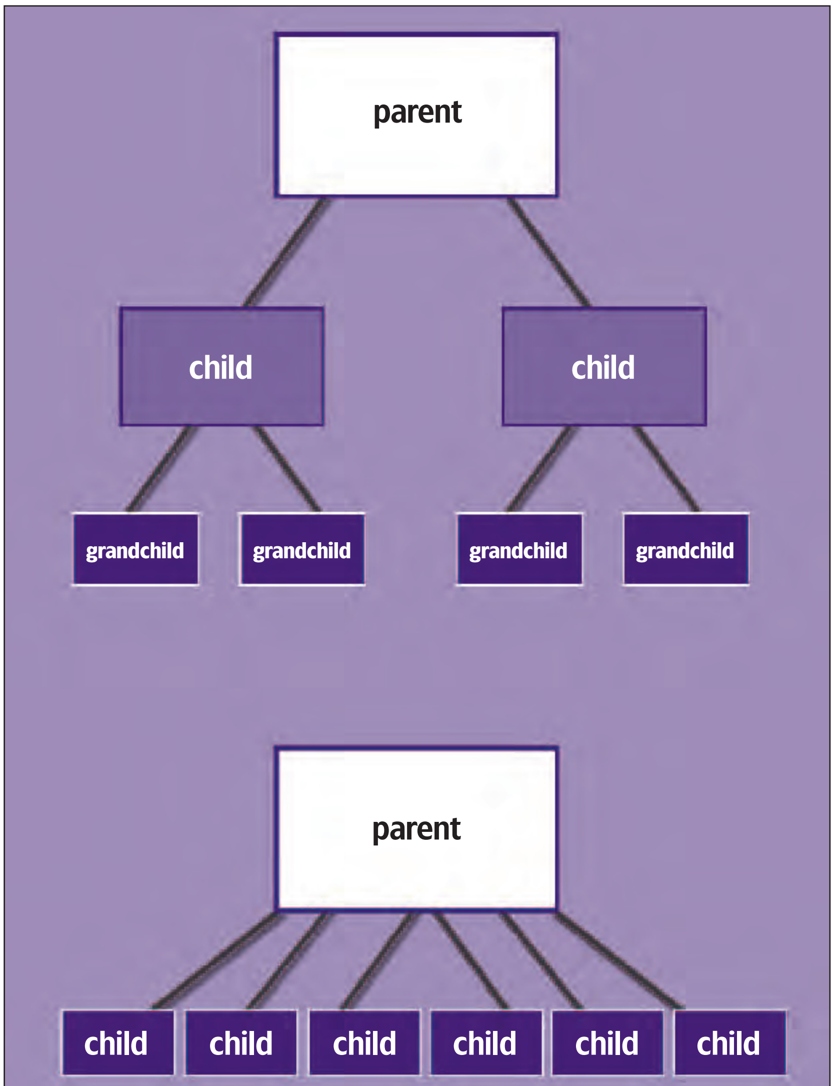
Before we go into the technicalities of how that script works, I first want you to run it. Now, try pressing **Ctrl-C** or **Ctrl-**. Then try opening a new console up and typing **killall php**. All being well you should have seen **Caught SIGINT** on the screen every time you pressed **Ctrl-C**; see **Caught SIGQUIT** on the screen every time you hit **Ctrl-**, and seen **Caught SIGTERM** on the screen every time you typed **killall php**. Now try typing **ps aux | grep php** into the other terminal to get the PID of the PHP process, and typing **kill -9 <PHP PID>**. This time you should see **Killed** on the screen, and the script should terminate.

Running that little test should have shown that **Ctrl-C** sends **SIGINT** to the running script, **Ctrl-** sends **SIGQUIT**, **kill <PID>** and **killall php** sends **SIGTERM**, and **kill -9 <PID>** actually kills the script. The reason for the latter is because **-9** sends the special signal **SIGKILL**. This signal cannot ever be overridden in our scripts – whereas **SIGTERM** means “I'd like you to terminate”, **SIGKILL** means “you're getting killed whether you like it or not”. You could, for example, use **kill -3 <PID>** to send the **SIGQUIT** signal, but pressing **Ctrl-** is much easier!

Now that you have an idea of what the script does, it's time to see how it works. The first line seems to call the **declare()** function – something we haven't covered before. Usage of declare is quite advanced, so we won't actually be covering it here beyond “you need this line at the top of your script if you want to use signals”.

Next comes the signal callback function, **signal_handler()**. Note that it takes a signal as its only parameter, and when it gets called by PHP the signal that was received will get passed in. So, the contents of the callback function is simply a switch/case statement where we print out a relevant message according to the signal received – we'll be looking at this more in a moment, but for now continue on after the function. The three calls to **pcntl_signal()** set up **SIGTERM**, **SIGQUIT**, and **SIGINT** to call **signal_handler()** when the appropriate signals are received, and finally the program goes into an infinite loop to cause it to sit around waiting for signals.

Hopefully that should explain it all, but before you start thinking you're the master of signals, let me briefly re-iterate that **SIGTERM** and **SIGQUIT** are there as polite notices to your script that the user has asked them to shut down. As you saw earlier, our script currently prints out a message when the signals are received, but it doesn't actually do anything about it, which is incredibly frustrating if you're an end-user trying to shut a script down! Generally, you should use **SIGTERM** and **SIGQUIT** to clean up – close database connections, save files, etc - then shut down. **SIGINT**, however, is a little different. Though it can be used to halt a program (and often is), it is also often used to say “user wants to know what's going on” or “stop doing the big time-intensive function, but keep running the program”. It's your call what you do, but by default we strongly recommend you exit on **SIGTERM**, **SIGQUIT**, and **SIGINT**.



FORKING HELP!

Although spawning new processes isn't overly hard with PHP, you shouldn't think it's easy, as it's a long way away from being *that*. The diagrams above show the two possibilities: far above you can see a parent process spawning children, and those children spawning grandchildren; near above you can see the parent application spawning six children directly. Of the two, the first seems the most logical from a theoretical point of view – it's more balanced, and allows you to segment tasks up in a more controlled manner. However, practically it's best to go with the second option – spawning all children from the original application. The reason for this is that recursive spawning (as the far above example essentially is) is potentially dangerous. Dangerous, that is, unless you're either a supremo programmer who never writes bugs, or if you're an average programmer who just likes to gamble now and then. The problem is that one minor bug in your code can easily lead to infinite recursion – processes spawning processes that spawn processes that spawn... well, you get the idea. Unless you want to run the risk of bringing your server to its knees, spawning all your children from one point is *much* safer.

Future echos

We've only managed to touch on the very beginning of the special Unix functions in PHP, but I hope you agree that it makes for a much more advanced (and much more *standard*) application. Linux users, from newbies to veterans, nearly always know that pressing **Control-C** ends an application, and being able to catch that and work with it constructively is a great way to increase the usability of your scripts. Having said that, I hope you can also see why much of this functionality isn't really a good idea for an *Apache* module! There's more to come, though – next month we'll be looking at how the special alarm signal works and also how POSIX functions let you program in PHP in much the same way as C. LXF

NEXT MONTH

Next issue we'll be looking at alarms as well as the rest of the POSIX functions – see you then!

PERL TEMPLATE TOOLKIT

Using Templates from Perl

PART 3 Dave Cross completes his series on the Template Toolkit by looking at how to use it from a Perl program.

ABOUT THE AUTHOR...

Dave Cross is a full-time Perl consultant and the author/co-author of a number of books on the subject, including the recently published *Perl Template Toolkit* (O'Reilly, ISBN: 0-596-00476-1)

Over the last couple of months, we have been looking at using the Template toolkit from the command line using the utilities *tpage* and *ttree*. None of the examples that we have looked at have involved us writing a single line of of Perl code.

That's fine for simple projects, but as things get more complex, it makes sense to do a lot of the heavy lifting in something a bit more powerful than the Template Toolkit's presentation language. The TT language was never designed to be a general-purpose programming language. It's a specialised language for controlling the presentation of data. As the Template Toolkit is written in Perl, it's easiest to use it from within Perl programs. So that's what we will use in this article.

Having decided that we are going to split the processing between a Perl program and a template the next thing we need to do is to decide exactly where to make this division. In my experience this is usually a pretty simple decision to make. Most programs fall quite neatly into one section that gathers all of the required data and another which presents the data to the user.

If you're having trouble deciding where to make this split, then it's often useful to consider an alternative display medium for your data. For example, if you're building a plain text file, consider what you would need to change if you were to build an HTML page containing the same data. The data is exactly the same, it's just the presentation that has changed. So the bits of processing that need to change are the bits that should be in the template.

Using the Template Toolkit from Perl

The main interface to the Template Toolkit from a Perl program is the Template module. Template is an object-oriented module, but don't let that scare you. It's really very simple to use.

Like all Perl modules, you load the Template module into your program with the **use** statement like this.

```
use Template;
```

You then need to create a Template processor object using the **new** method. This can be as simple as this:

```
my $tt = Template->new;
```

But there is also an optional parameter to **new** – we'll be looking at that a bit later on in this tutorial, as trying to explain it at this point may create some confusion.

To use the Template processor object, we call the **process** method, passing it the name of a template to process.

```
$tt->process('template.tt') or die $tt->error;
```

The template processor looks for the template file in the current directory (we'll also see how to change that later) and processes it in exactly the same way as *tpage* or *ttree* would. The results of processing the template are written to STDOUT (but we'll see how to change that very soon).

Notice that if there is any problem processing the template then **process** returns a false value. We can check for that and use the **error** method to produce a suitable error message as we terminate the program.

Passing Variables to the Template

Of course, most templates need some kind of input in order to do anything useful. With *tpage* and *ttree* we used the **--define var=value** options to pass variables into the template. There must be a way to do something similar from Perl.

And, of course, there is. The **process** method takes an optional second parameter that defines the variables that the template will use. This parameter is a reference to hash. The keys of the hash are the names of the variables and the values are the associated data. You can therefore define variables like this:

```
my %vars = (name  => ‘Dave Cross’,
        email => ‘dave@example.com’);

$tt->process(‘template.tt’, \%vars) or die $tt->error;
```

This code defines two variables called **name** and **email**, which can be referenced within the template. You don't have to stop at scalar values like the ones seen here. You can build any kind of complex data structure.

```
my %vars = (invoice => {
      number => ‘101’,
      date   => ‘1st April 2004’,
      client => ‘Example Inc’,
      addr   => ‘1000 Example Road, Exampleton’,
      lines  => [
               {
                desc  => ‘Reversing polarity’,
                price => ‘1000’
               },
               {
                desc  => ‘Regeneration care’,
                price => ‘2000’
               }
              ]
             });
```

This example shows a complex, multi-levelled data structure that models an invoice. The **invoice** variable is a hash and its parts can be accessed within a template as, for example, **invoice.number** and **invoice.date**. The value for **invoice.lines** is an array, so you can access the individual items as, for example, **invoice.lines.0.desc** or you could use it in a FOREACH loop. We'll see more of this example later, but if you want more information about using complex data structures in Perl, see the *perlreftut* and *perldsc* manual pages.

Most Perl programs that use the Template Toolkit will spend a large part of their time building an appropriate data structure to pass to **process**.

Controlling Output

As I mentioned previously, **process** sends its results to STDOUT by default. You can change by using its optional third parameter. This can take a number of different types of value. The most common of them is a string which is assumed to be the name of a file. The output from the template is written to this file.

Another option is to pass a reference to a scalar variable. In this case. the output from the template is put into that variable. This is useful if you want to post-process the output in some way. There are a few other more esoteric alternatives. For details, see the documentation that comes with the Template Toolkit.

More Options

Last month, we saw some other options that *ttree* uses to control exactly how the template is processed. We can do the same thing with the Template module. In fact this method gives us even more options. The processing options are set up when you create a template processor object with **new**. The **new** method takes an optional argument which is a reference to a hash of options.

```

my %options = (INCLUDE_PATH => ‘./lib’,
           OUTPUT_PATH  => ‘./out’,
           PRE_PROCESS  => ‘header.tt,
           POST_PROCESS => ‘footer.tt’);

my $tt = Template->new(\%options);

```

In this code we set four options. **INCLUDE_PATH** defines a directory where the template processor will look for any templates. If you want more than one directory then set this option to a reference to an array of directories.

```
INCLUDE_PATH => [ ‘.’, ‘./lib’, ‘/opt/templates’ ]
```

OUTPUT_PATH defines the directory where any output files will be written.

PRE_PROCESS and **POST_PROCESS** define templates that will always be processed before and after and templates passed to **process**. This can be useful if, as in this example, you want to add a header and footer to every template. I often use **PRE_PROCESS** to process library templates that contain

»

THE INVOICE DATABASE

Navigating tables

We'll assume that we have a very simple database that stores details of our invoices. The database has three tables which contain data about clients, invoices and invoice lines. Let's look at the tables one at a time.

The Client table

Each of our clients has one row of data in the client table. Each row contains the client's name and address together with a unique identifier for the client. In *MySQL*, the definition of the table looks like this:

```
CREATE TABLE client (
  id int(11) NOT NULL,
  name varchar(50) default NULL,
  address varchar(250) default NULL,
  PRIMARY KEY  (id)
);
```

The invoice table

Each invoice that we send will create one new row in the invoice table. It contains the invoice number and date along with the id of the client that the invoice was sent to. The table definition looks like this:

```
CREATE TABLE invoice (
  id int(11) NOT NULL default ‘0’,
  invdate date default NULL,
  client int(11) default NULL,
  PRIMARY KEY  (id),
  INDEX fk_cli (client),
  FOREIGN KEY (client) REFERENCES client(id)
);
```

Notice that we have also given the table a foreign key which declares that the client column in this table is a reference to the id column in the client table.

The line table

Within an invoice, we have a number of invoice lines. Each of these represents one of the items that the invoice charges for. An invoice line has a number, a description of the goods or services sold and a price. It also contains the number of the invoice that it belongs to. Here's the table definition:

```
CREATE TABLE line (
  invoice int(11) NOT NULL default ‘0’,
  line_no int(11) NOT NULL default ‘0’,
  description varchar(250) default NULL,
  price float(10,2) default NULL,
  PRIMARY KEY  (invoice,line_no),
  INDEX fk_inv (invoice),
  FOREIGN KEY (invoice) REFERENCES
invoice(id)
);

```

Notice here, that once more we've defined a foreign key linking the line table back to the invoice table.

« configuration data. Both of these values can also be set to an array reference if you want to pre-process or post-process multiple templates.

```
PRE_PROCESS => [ 'config.tt', 'header.tt' ]
```

Creating Invoices

For this month's example, we're looking at creating invoices. Assume that we have details of invoices in a database and that we have a Perl module called 'Invoice' that gives us access to the invoice data. See the boxes on *The Invoice Database* (on the previous page) and *Invoice.pm* (below) for more details on how these are set up.

Here's a simple text template for an invoice:

```

 INVOICE [% invoice.id | format('%05d') %]

 Date: [% invoice.invdate %]

 To: [% invoice.client.name %]
[% FOREACH addr_line = invoice.client.address.split('\n') -%]
    [% addr_line %]
[% END -%]

[% FOREACH line = invoice.lines.sort('line_no') -%]
  [% line.description | format('%-40s') %] £[% line.price |
format('%.2f') %]
[% END %]
  [% 'Total:' | format('%40s') %] £[% total | format('%.2f') %]
```

This template expects an Invoice object to be passed to it in the variable **invoice** and also the total value of the invoice in the variable **total**. Most of the data that it needs is in the invoice object, and it can access this by calling the object's various methods using the dot notation. Notice that these dots can be strung together to create expressions like **invoice.client.name** to get the name of the client associated with the invoice.

We've also made good use of the *format* plugin. We use it to ensure that the invoice number always has five digits and to ensure that the prices all have two decimal points, but we also use it to ensure that the descriptions are all padded to forty characters and therefore the price column lines up correctly. All of these considerations are about the presentation of the data, so they quite rightly belong in the template.

The Invoice Program

Here's the program that calls the **invoice** template.

```
#!/usr/bin/perl

use strict; use warnings;

use Template; use Invoice;

my $id = shift || die "No invoice number given\n";xi
my $fmt = shift || 'txt';

my $inv = Invoice->retrieve($id)
  || die "Invoice $id not found\n";

my $total; $total += $_->price foreach $inv->lines;

my $tt = Template->new;

my %vars = ( invoice => $inv, total => $total );

$tt->process("$fmt.tt", \%vars, "inv$id.$fmt")
  or die $tt->error;

```

The program starts by loading the **strict** and **warnings** modules – which no Perl program should be without. It then loads the **Template** and **Invoice** modules that we will specifically need for this application.

The program expects two arguments. The first is the number of the invoice to process. This is mandatory and the program dies if it isn't given. The second argument is optional and defines an output format. The default format is **'txt'**.

The program then retrieves the invoice from the database using a method that **Class::DBI** has created for us in our Invoice class. Having successfully retrieved the invoice we calculate the total by adding together the price fields from all of the lines in the invoice. Again, the **lines** method was automatically created by **Class::DBI**.

Then all that's left to do is to create our template processor object and process the template. We use three arguments to the **process** method. The first argument is a string containing the name of the template to use. For the text format, we'll use **txt.tt**. The second argument is a hash containing the variables. The third is the name of the output file. For text invoices, the name will be something like 'inv1.txt'.

Running this program with some simple test data gives the following output:

```

 INVOICE 00001

 Date: 2004-05-01

```

INVOICE.PM

Perl module writing is easier than you think

When you are creating a Perl application that talks to a database, it's generally a good idea to isolate all of the database interaction in one place. And usually it makes sense to write a module to handle it all.

Writing Perl modules really isn't as hard as you might think and there are plenty of tools out there to make it as easy as possible. I have recently started using the module Class::DBI for all of my database work. It's a wrapper around Perl's standard database interface module (called *DBI*) and it's a very easy way to create very useful classes built around database tables. Here are the contents of my file Invoice.pm which contains all the database code for this application.

```
package Invoice;

use Class::DBI::Loader; use
Class::DBI::Loader::Relationship;

my $loader = Class::DBI::Loader->new(dsn =>
'dbi:mysql:invoice:tma2',
                          user => 'invoice',
                          password =>
'inv01ce');

my @rels = (
  'a client has invoices',
  'an invoice has lines');

for (@rels) {
  $loader->relationship($_);
}

1;

```

That's about a dozen lines of code, and when we load it into our program (with use Invoice) we will get three new classes: Client, Invoice and Line; which are object-oriented interfaces to our three tables.

The cleverest thing about this tool is that it also understands the relationships between our tables. This means that if we've got an Invoice object, it is very easy to get its associated Client and Line objects.

For more information about how this all works, go to http://search.cpan.org/ and look for Class::DBI, Class::DBI::Loader and Class::DBI::Loader::Relationship.

```
To: The Doctor
    The TARDIS
    Somewhere in space and time

  Reversing polarity      £1000.00
  Regeneration care       £2000.00

                  Total: £3000.00

```

But what happens when we decide that we also want to put out invoices on our website? Or that we want to create RTF versions of invoices that can be edited in *OpenOffice.org*?

This is where the work of separating the data collection from the presentation really pays off. By just creating a new template, we can create a new view of our data very easily. Here is an example HTML template.

```

<html>
  <head>
    <title>Invoice [% invoice.id | format(‘%05d’) %]</title>
  </head>
  <body>
    <h1>INVOICE [% invoice.id | format(‘%05d’) %]</h1>

    <table>
      <tr>
        <td>Date:</td><td>[% invoice.invdate %]</td>
      </tr>
      <tr><td colspan=”2”>&nbsp;</td></tr>
      <tr>
        <td valign=”top”>To:</td>
        <td>[% invoice.client.name;
               invoice.client.address.split(‘\n’).join(‘<br>’) -%]
        </td>
      </tr>
    </table>

    <table> [% FOREACH line = invoice.lines.sort(‘line_no’) -%]
      <tr>
        <td>[% line.description %]</td>
        <td>&pound;[% line.price | format(‘%.2f’) %]</td>
      </tr> [% END %]
      <tr>
        <td align=”right”>Total:</td>
        <td>&pound;[% total | format(‘%.2f’) %]</td>
      </tr>
    </table>
  </body>
</html>

```

Basically, this does the same things as the text template, but it produces HTML instead of plain text. Instead of worrying about lining up the text columns, we have used HTML tables. Instead of displaying a pound sign, we use the **&pound;** HTML entity. Simply put this template into **html.tt** and our existing script can be used unchanged to create HTML pages. The fact that we find it so easy to create another view of the data means that we must have got the separation of processing and presentation just about right. LXF

INSTALLING THE IDE

KDE Programming with KDevelop

PART 1 Jono Bacon kicks off with a brand new series teaching KDE development by preparing our *KDevelop* installation.

Welcome, one and all, to a brand new series that is going to be focusing on the art of KDE development. There is little doubt that KDE is one of the most popular desktop environments for Linux, and the KDE team has worked hard to not only produce a quality desktop environment, but also quality programming environment. We will be exploring various facets of this coders' toolbox in the coming months.

As with any other programming environment on Linux such as GNOME, XFree86, SDL etc, there are a large number of programming tools and utilities that are available. The Open Source culture of Linux has been consequential in hackers creating nice tools for them to hack with. The *'scratch your own itch'* philosophy coined by Eric S Raymond has evidently created a lot of scratching, and there is a huge range of tools available to make coding that bit more fun and simple. The problem with this enormous batch of tools is choosing which ones are appropriate to use in specific circumstances, and learning how they fit together. This is where we introduce *KDevelop*.

KDevelop is without a doubt one of the most popular programming environments available for Linux. With *KDevelop* you can create a variety of applications including KDE, GNOME, *Qt, GTK, wxWindows* (or *wxWidgets*, as it is latterly known) PHP, Perl, Python *etc*; and the environment includes a large number of features and facilities that make coding easier. Not only does *KDevelop* include a lot of its own unique functionality, but it can utilise other tools to create a single grand unified environment. Due to the fact that *KDevelop* pulls these different strings together in a single place, it provides a compelling environment to work in; the only real complication is installing everything.

This first part of our series is really more of a precursor to the tutorial series, rather than a coherent step in its own right. Over these next couple of pages, we will install and configure *KDevelop* and mention the different tools that are needed to get the most out of the environment. This article will provide the preparation for the next issue, when we actually begin coding.

KDevelop dependency

Like any comprehensive application, installation can often include dependencies on other programs, libraries and tools. *KDevelop* is no different in this regard, and although you can just install *KDevelop*, you can also install optional programs that can be used with *KDevelop* to expand its functionality.

A stock *KDevelop* relies on two primary chunks of software; *Qt* and KDE. More specifically, *KDevelop* relies on the most recent version of the *Qt* library and the most recent version of the KDE Libraries. There is no specific need to have the actual latest version of these libraries, but you need to have version 3.x of both, and we do recommend that you try to use the most recent version of this software. Another important point to make is that the actual KDE interface (known as the KDE base) is not required to run *KDevelop*. If you would like to use a different environment such as GNOME, you will not need to install the KDE base package.

KDEVELOP FROM THE CVS

Postcards from the Bleeding Edge...

Are you are the kind of person that likes to experiment with pre-release code? In that case, you may want to try to compile the bleeding-edge *KDevelop* that the developers are currently working on. As with all test releases, there is no guarantee that this will actually work properly or compile, but you may want to have a go. To do this you first need to set the CVSROOT variable thus:

```
export CVSROOT=:pserver:anonymous
@anoncvs.kde.org:/home/kde
```

Next log in with:

```
cvs login
```

Press Enter when asked for a password. You can then check out the *KDevelop* module by passing the following:

```
cvs co kdevelop
```

You can also check out the latest stable release by using the KDE_3_2_BRANCH:

```
cvs co -r KDE_3_2_BRANCH kdevelop
```

With the code checked out, you can then proceed to compile it in the normal way.

Installing binary packages

Installing pre-compiled packages of *KDevelop* and its dependent/optional software may initially look a bit off-putting when you read through the list in the box on the right, but don't worry! It is actually quite simple: the main thing is to remember to install dependent software (such as *Qt*, *CVS* etc) FIRST.

If you are using an RPM-based system such as Red Hat/Fedora, Mandrake, SUSE *etc*, you can install a package with:

```
rpm -i package.rpm
```

You can find many of the RPMs needed for running *KDevelop* on the **http://rpmfind.net** website.

If you are using a Debian-based system you can use the mighty apt-get to install the software. Simply type in:

```
apt-get update
apt-get install kdevelop
```

You will need to be using Debian Unstable to get the latest (and rewritten) version of *KDevelop* (version 3.x).

Compiling KDevelop

If you would like to compile *KDevelop* yourself, you will need to have the programming libraries of all the software required. You will also need the programming libraries of the optional software that you would like to use with *KDevelop*. In many distributions this is named as *package-dev* or *libpackage-dev*. You will need to have this software installed BEFORE you compile *KDevelop*.

You can download the *KDevelop* source code from the KDevelop homepage at **www.kdevelop.org/**. When you have downloaded the code, you can unzip it with:

```
bunzip2 kdevelop.bz2
tar xvf kdevelop.tar
```

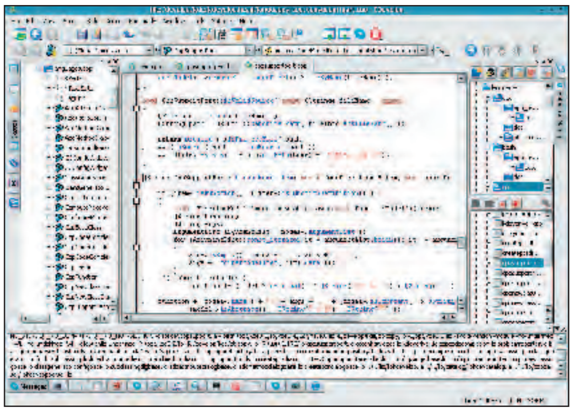
Before you compile, you will need to set some special environmental variables that point to different resources on your system. The main two variables are the KDEDIR and QTDIR variables. These point to the location of your KDE and *Qt* installations. You can set these in the bash shell with:

```
export KDEDIR=/location/of/kde
export QTDIR=/location/of/qt
```

As an example:

```
export KDEDIR=/opt/kde
export QTDIR=/opt/qt
```

You should also set the PATH variable to include the binary directory of your KDE and *Qt* applications. This is particularly important so that *KDevelop* can find KDE applications and tools such as *Qt Designer*. You will probably have a PATH variable set at the moment. Check this with:



***KDevelop 3.0* in IDEAl mode – you can write commercial/closed software with it despite *KDevelop* itself being GPL.**

```
echo $PATH
```

If for example, the value of your PATH is /usr/bin:/usr/local/bin, you should add the KDE/*Qt* directories before this:

```
export PATH=/opt/kde/bin:/opt/qt/bin:/usr/bin:/usr/local/
bin:$PATH
```

Another two variables to set are the lib directories (libraries) for KDE and *Qt*:

```
export LD_LIBRARY_PATH=$QTDIR/lib:$KDEDIR/lib:
$LD_LIBRARY_PATH
export LIBRARY_PATH=$QTDIR/lib:$KDEDIR/lib:
$LIBRARY_PATH
```

With these variables set we are ready to compile the code. To do this, we first need to configure the code:

```
./configure
```

The configure script actually has a number of different options that can be set and unset. You can find out these different options by running:

```
./configure --help
```

With our source configured, we can then go on to compile and install the code in the familiar way:

```
make
make install
```

You will of course need to be root to do this usually. Reading through some of the man pages and other documentation will put you on the starting-grid for next issue's four-page tutorial! LXF

SOFTWARE REQUIREMENTS

Properly equip your PC for KDE development

These are the specific software requirements for *KDevelop* that you need to follow this series:

- **KDE (3.0.2+)**
  www.kde.org
- ***Qt* (3.0.5+)**
  www.trolltech.com/products/qt
- ***g++* (2.95.3 + or compatible)**
  gcc.gnu.org
- ***GNU make***
  www.gnu.org/software/make
- **Perl (5.004+)**
  www.perl.com
- ***autoconf* (2.52+)**
  www.gnu.org/software/autoconf
- ***automake* (1.6+)**
  www.gnu.org/software/automake
- ***Flex* (2.5.4+)**
  www.gnu.org/software/flex
- ***Berkley DB* (3.0 - 4.1)**
  www.sleepycat.com

In addition to these essential requirements above, the following tools are optional:

- **ht://Dig (3.1.6+)**
  www.htdig.org
  A library used for searching for text.
- ***Valgrind***
  http://valgrind.kde.org/
  A tool for finding and fixing memory leaks. Memory leaks are important to avoid when create KDE applications in C++.
- ***GDB* (5.0+)**
  www.gnu.org/software/gdb
  A full featured debugger for finding and squashing those bugs. *KDevelop* integrates *GDB* within the *KDevelop* environment.
- ***CVS* (1.10.6+)**
  http//www.cvshome.org
  The most popular source control system, *CVS* is commonly used when a number of developers are working on one codebase.
- ***Perforce* (2003.1 +)**
  www.perforce.com/perforce/products.html
  Version control and configuration management.
- ***Clearcase* (4.0 +)**
  www.rational.com/products/clearcase/index.jsp
  Integration of *Clearcase*.
- ***CTags* (5.x +)**
  http://ctags.sourceforge.net
  Special tags for easing the editing and development of code. This is built right into the *KDevelop* editor.
- ***Doxygen* (1.3.4 +)**
  www.doxygen.org
  Automatic generation of programming APIs and documentation.
- ***dot* (1.8.7 +)**
  www.graphviz.org
  Generation of graphical class diagrams. This is a highly recommended tool, as you will find that the class viewer is very useful when programming KDE applications.

Most of these tools are available in RPM and Debian package formats, and source code tarballs. If you are using a distribution that allows you to download packages from the Internet and install them automatically, you should have few problems with installing *KDevelop* and its optional extras.

NEXT MONTH

This month's two-pager is more of a preamble than anything else – with *KDevelop* installed and ready to go, we are all set for the series to kick off properly next issue. In the meantime, I suggest that you fire up *KDevelop* and have a play with it. *KDevelop* may seem a little complex at first, but we will be covering how to use its different facilities in the coming months.

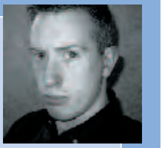
I look forwarding to resuming next month and taking our first steps of creating our first full-featured KDE application!

Answers

If you are really stuck and the HOWTOs yield no good result, why not write in? Our resident experts will answer even your most complicated problems!

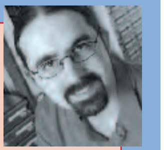
Our experts

Whatever your question is, we can find an expert to answer it – from installation and modem woes to network administrations, we can find the answer for you – just fire off a letter or email and it'll all be taken care of.



LXF answers guy **David Coulson** is a networking and security guru with plenty of sysadmin experience to boot.

NIck Veitch is the editor of the magazine, and answers your easy questions! Or indeed anything to do with *Grub, LILO, netatalk, vi...*



Hans Huberland is Rackspace Managed Hosting's Linux expert. Send any Linux system admin questions to **sysadminqa@rackspace.co.uk**

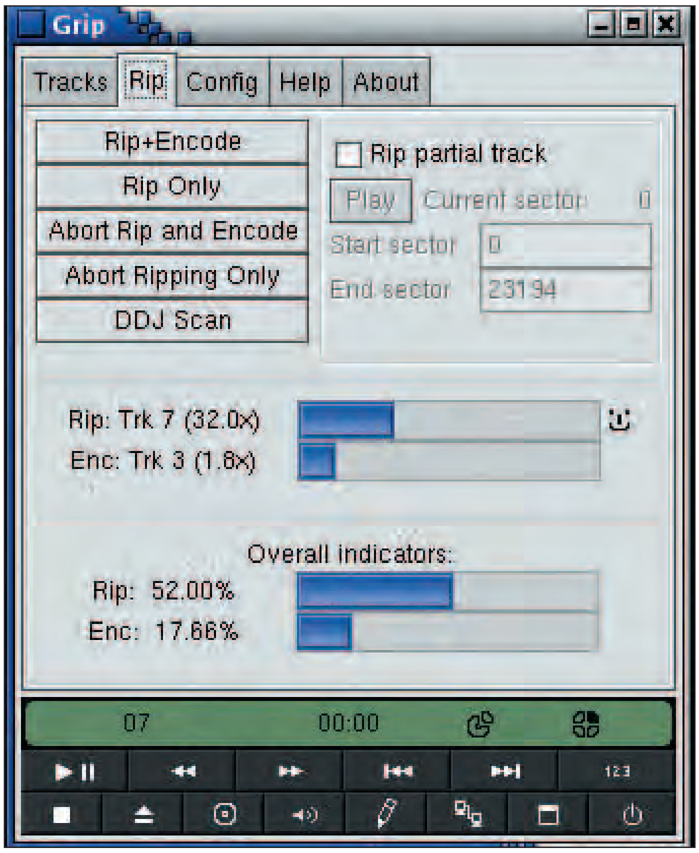
Last Will and testament

Q I was 'forced' against my better judgement to get Will Young's CD for my wife as part of her Christmas present. Of course, it would not play in our DVD player (which is a Denon all-in-one unit). I remembered your article and took it up to my computer. I had tracks ripped in order to save onto a blank CD in minutes.

OK, so it involved going into (urgh!) Windows and using *Nero Multimounter*, but it was a doddle. How will that stop piracy? It just stopped my wife playing her CD legitimately and forced me to make an 'illegal' copy to play in our living room. Pirates will obviously have no problem creating a ripped copy and duplicating it to their heart's content. Is there a *Multimounter* equivalent for Linux?

***"Honest" Dave,** via email*

A If you want to rip a CD in Linux, the easiest way is to use one of the many front-ends for *cdparanoia*, such as *ripperX* or *grip*. These let you pull the music from the CD, and output either WAV or MP3 format files. Of course, one can always use *cdparanoia* from the command-line, which is actually very simple, as the format of the command-line is very straightforward. To create a new audio CD, one can use a front-end for *cdrecord*, such as *gtoaster* and burn from MP3 or WAV audio. Various front ends exist for *cdrecord*, however they all essentially do the same thing as they actually burn their information through the same command-line process.



```
Rip: Trk 7 (32.0x)
Enc: Trk 3 (1.8x)

Rip:  52.00%
Enc:  17.66%
```

***Grip* is a *GTK+*-based tool for ripping and encoding music from audio CDs, which you can then burn back to CD or play in MP3 format.**

Tar baby

Q I am a complete novice with Linux and as a computer person probably a danger to myself and my computer rather than a wiz. But my excuse is that its better to die trying than not to bother trying at all!

My questions will show you the level of skills I (don't!)have in Linux:

1 Where do you put tar files how do you get them to work? (I've figured out RPMs, they work nicely and go where they are meant to with out me fiddling.)

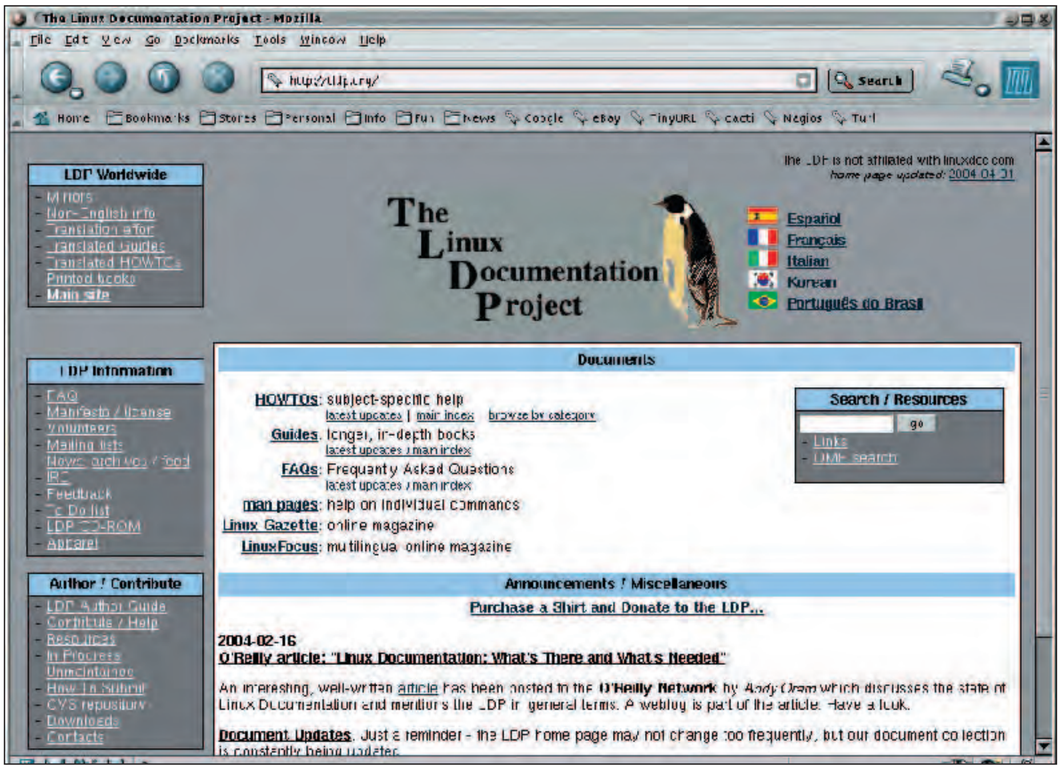
2 How do I navigate Howtos? (I'm not even sure what they are or where to find them.)

Some really (and I mean *really*) basic stuff in the magazine would be an inducement to those of us wanting to migrate to Linux.

Now to the real problem, I have loaded Mandrake 10 (it found my Internet and network system, soundcard works *etc*) on to my hard drive, which is shared with Windows 2000. Windows is on the primary, and Linux on a logical drive. It is an IDE drive, and then I have two SATA on a RAID setup.

The problem now is, I can't get the Windows to boot – in fact, I can't even get the system to catch the Windows CD on boot up. My Windows 2000 boot floppies (of course) have also decided to corrupt. Help! How do I get the boot to find Windows? As much as I want to only use Linux, it's a steep learning curve for those of us used to the point-and-click ease of Windows...

***Phill Taylor,** Sweden*



The Linux Documentation Project - Mozilla

File Edit View Go Bookmarks Tools Window Help

http://tldp.org/

Search

Home Bookmarks Stores Personal Info Fun News Google eBay TinyURL cacti Nagios Tu'l

The LDP is not affiliated with linuxdoc.com
home page updated: 2004-04-01

LDP Worldwide
- Mirrors
- Non-English info
- Translation effort
- Translated Guides
- Translated HOWTOs
- Printed books
- Main site

The Linux Documentation Project

Español
Français
Italian
Korean
Português do Brasil

LDP Information
- FAQ
- Manifesto / license
- Volunteers
- Mailing lists
- News: archives / feed
- IRC
- Feedback
- To Do list
- LDP CD-ROM
- Apparel

Author / Contribute
- LDP Author Guide
- Contribute / Help
- Resources
- In Progress
- Unmaintained
- How To Submit
- CVS repository
- Downloads
- Contacts

Documents

HOWTOs: subject-specific help
latest updates | main index browse by category

Guides: longer, in-depth books
latest updates / main index

FAQs: Frequently Asked Questions
latest updates / main index

man pages: help on individual commands

Linux Gazette: online magazine

LinuxFocus: multilingual online magazine

Search / Resources
go
- Links
- OMF search

Announcements / Miscellaneous

Purchase a Shirt and Donate to the LDP...

2004-02-16
O'Reilly article: "Linux Documentation: What's There and What's Needed"

An interesting, well-written article has been posted to the **O'Reilly Network** by *Andy Oram* which discusses the state of Linux Documentation and mentions the LDP in general terms. A weblog is part of the article. Have a look.

Document Updates. Just a reminder - the LDP home page may not change too frequently, but our document collection is constantly being updated.

***The Linux Documentation Project* (TLDP) has HOW-TOs and documentation on nearly all popular Linux projects, including help for newbies.**

A Tar files generally contain source code, which you must compile before you can do something useful with it. Unless you have an understanding of the implications of compiling your own code – such as overwriting existing binaries or libraries – then it is generally best to stick to the packages provided by the creator of your distribution, or the distributor of the software you are using. Building something from source requires additional packages for headers of libraries you already have installed, as well as compiling tools so you can build something that works.

HOW-TOs for most Linux tools are available from **www.tldp.org**/ (a full mirror of which we often include on our covermounted DVDs in the 'Help' section), including a beginners guide to Linux covering all the basic stuff you need to know to get somewhere with Linux. You may want to check out the *LILO* HOW-TO which will explain how to configure *LILO* to boot your Windows filesystem.

Newbie niggles

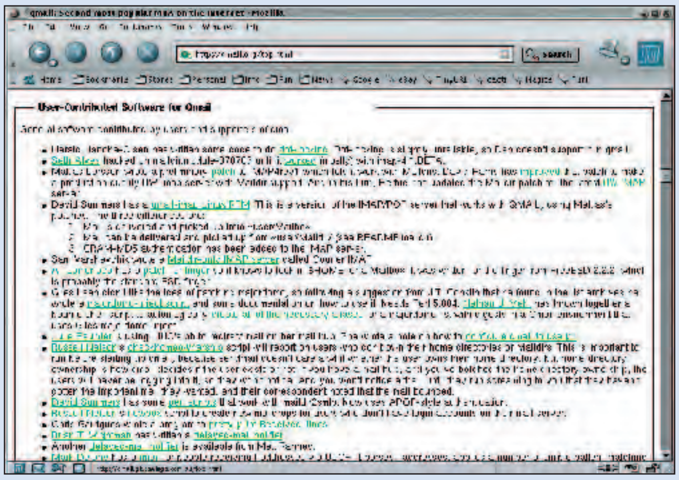
Q At the moment I am trying Mandrake 9.2, but earlier attempts with SUSE 8.1 and Fedora Core 1 (which was probably the best distro I've tried so far), plus earlier versions of Mandrake all seem to be agonisingly slow compared to any version of Windows.

A QUICK REFERENCE TO: qmail

***Sendmail* is certainly the most well-known Open Source mail server, but is by no means the only one, nor the best. Picking a mail server is down to personal choice, but a particularly powerful *Sendmail* alternative is the popular *qmail*.**

***qmail* has a very simple configuration system, which relies on having individual files within /var/qmail/control, rather than one massive configuration file which is a nightmare to maintain. *qmail* was designed with scalability and virtual domain hosting in mind, so there is no need for ugly configuration options to make it handle more than one domain, plus users can handle their own mail aliases. This removes some administration control from the postmaster, but it does mean that their job is much easier as all they have to do is point a domain to a particular user, rather than add a whole load of aliases for each domain.**

dot-qmail files are a particularly important part of *qmail*, as these decide how email will be handled by the server once it has been accepted. For example, if we have a domain domain.tld controlled by the user bob, then bob will get all email for the domain send through his *qmail* configuration. If we email test@domain.tld, then *qmail* will first check for ~bob/.qmail-test, then for ~bob/.qmail-default. We can, of course, create separate configurations for individual domains, so rather than having domain.tld controlled by 'bob', we can have it controlled by bob-domain.tld. Now *qmail* will check for .~bob/.qmail-domain.tld-test, then for ~bob/.qmail-domain.tld-default and finally for ~bob/.qmail-default. For local mailboxes, qmail will use ~bob/.qmail, so if you send a mail to bob on the local machine, *qmail* can handle it differently to a virtual domain.



qmail: second most popular MTA on the internet - Mozilla

User-Contributed Software for Qmail

[illegible]

***qmail* is a very popular MTA, with numerous third-party patches with additional capabilities.**

These dot-qmail files can either forward email to another email account, or insert it into a mailbox. As well as regular mbox format files, *qmail* can also use Maildirs, which are generally considered to be more flexible and less prone to problems than mbox files. If an mbox file becomes corrupt, then all your email is gone. If a Maildir entry is corrupt, then you lose one email, which is naturally much better. It is also quite easy to index files in a Maildir, so it's much quicker to pick out an individual message from the Maildir. Not all mail clients support Maildirs, although *mutt* and the popular *Courier* IMAP client will use Maildirs on a *qmail* system.

»

FREQUENTLY ASKED QUESTIONS THE LINUX KERNEL

FAQ I HAVE JUST INSTALLED LINUX BUT WANT TO UPGRADE THE KERNEL. HOW DO I FIND OUT WHAT KERNEL I'M RUNNING?

Most current distributions ship with 2.4 kernels, although some use 2.2 kernels instead. You can easily check what kernel you're running with the **uname** command. Simply do:

```
$ uname -a
Linux tailtiu 2.4.19-rc1-pe-lb-brnf #1
 SMP Wed Jun 26 18:48:32 BST
2002 i686 unknown
```

This system is running **2.4.19-rc1**, with some extra patches. Likely your distribution creators will have applied their own patches so it won't be a simple 2.x.y kernel version.

FAQ WHERE DO I DOWNLOAD A KERNEL FROM, AND WHAT DO I HAVE TO GET?

You can download a tarball of the current kernel from **www.kernel.org/**. Generally one will simply want to grab linux-2.4.25.tar.gz from linux/pub/linux/kernel, which is the complete tarball of the kernel source. If you already have a kernel source tree available, one can obtain the patch between that and the next revision, saving some time downloading the entire source tree.

FAQ I'M RUNNING THE KERNEL WHICH CAME WITH MY DISTRIBUTION. DO I HAVE TO USE THEIR KERNEL OR CAN I GO AND BUILD MY OWN?

Red Hat distributes a kernel which is – to many Linux users' perception – patched way beyond all acceptable limits. However, some of RH's tools and utilities fail to function properly with a kernel which does not have RH's patches. Likewise, some third-party patches will conflict with the contents of the Red Hat kernel, as they will be expecting a clean kernel tree.

Unless you're in need of specific kernel capabilities which render a Red Hat kernel useless, due to incompatibility, it's often good to stick with distribution kernels. Of course, if you don't need any of the patches within the distribution's kernel, then a vanilla kernel tree should work perfectly well.

FAQ I'VE GOT MY KERNEL TAR FILE. NOW HOW DO I COMPILE IT?

The general build location is /usr/src/linux, so you need to **cd** into /usr/src then untar it with:

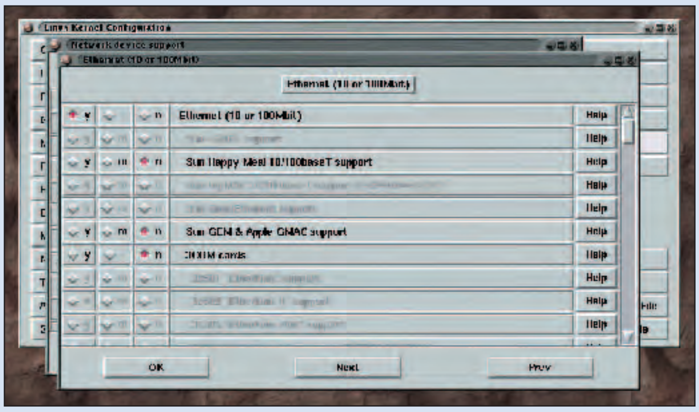
```
tar xzf ~/linux-2.4.25.tar.gz
```

Once the kernel tarball is unpacked, you will have a 'linux' directory which contains the source code. The general routine to configure, build and install a kernel is:

```
cd linux
make mrproper xconfig dep clean
bzlilo modules modules_install
```

First, you will need to configure your kernel. Generally, keeping the defaults will result in something that will boot, but it is likely that you will need to select other options for hardware devices, such as sound cards, SCSI interfaces and everything else which everyone won't be using. As long as there is a backup kernel available, accessible via *LILO*, then it should not be a major issue to try different options to decide which supply the required capabilities for the system.

bzlilo will build the kernel, copy it to /vmlinuz, then run **lilo** to install the new kernel. This will require lilo to be modified to use /vmlinuz as the image, rather than /boot/vmlinuz, although one can hack the Makefiles to install the kernel into /boot/.


Linux Kernel Configuration
Network device support
Ethernet (10 or 100Mbit)
Ethernet (10 or 100Mbit)
y n Ethernet (10 or 100Mbit)
y m n Sun Happy Meal 10/100baseT support
y m n Sun GEM & Apple GMAC support
y n 3COM cards
Help
OK
Next
Prev

Compiling the kernel with specific configuration options for our hardware makes for a more efficient kernel and a quicker system.

The last two **make** options build the loadable modules and plonk them in the right place for the kernel to use. Once you have a kernel built with loadable module support, you can just go in and reconfigure the kernel and install the modules.

FAQ CAN I ADD CAPABILITIES TO THE KERNEL SOURCE TREE? AND IF SO, HOW?

There are many patches available for the Linux kernel which enable it to do other things. One of the main projects which distribute combined kernel patches is WOLK, the Working Overloaded Linux Kernel. Rather than having to apply individual patches one at a time, WOLK provides a single patch which adds a whole bunch of different patches.

WOLK is available from **http://sf.net/projects/wolk/**

There is also the *-ac* kernel series, distributed by Alan Cox. These kernel contain many updates which have not yet made it into the mainstream kernel, along with some basic fixes for things broken in existing kernels. Many people use *-ac* kernels instead of the regular kernel tree, although as with distribution kernels, other kernels may object to being applied to an *-ac* tree.

FAQ ARE THE 2.6 LINUX KERNELS STABLE ENOUGH TO USE?

2.6 kernels are certainly stable, although if you're planning on using a 2.6 kernel in a production setting, then you will want to test it prior to rolling it out. Certainly if you're running CPU-intensive processes, you will noticeably benefit from the improved scheduling and changes to the SMP process queue. For general desktop use, you should also notice an improvement in responsiveness, due to the pre-emptive kernel patch and the removal of many locks caused by I/O operations.

« I have set aside an old machine to experiment with Linux before installing it on my main machine. The spare machine is a Pentium MMX 233MHz, 128MB RAM, 8GB HDD and so is within the minimum recommended. It has only the one disk and it is totally Linux.

Using Mandrake 9.2, with only the packages provided on the Mandrake disks, and without installing GNOME or any games *etc.* it takes 6 minutes 22 seconds to boot up! (Fedora was quicker.) I have tried formatting the disk using both ext3 and latterly ReiserFS, but it's dreadfully slow whatever I do. Within the bootup process it takes 2 minutes 20 seconds alone to check module dependencies!!! and another 20 seconds to check for new hardware. Please could you tell me:

1 Which is the fastest of the main Linux distributions?

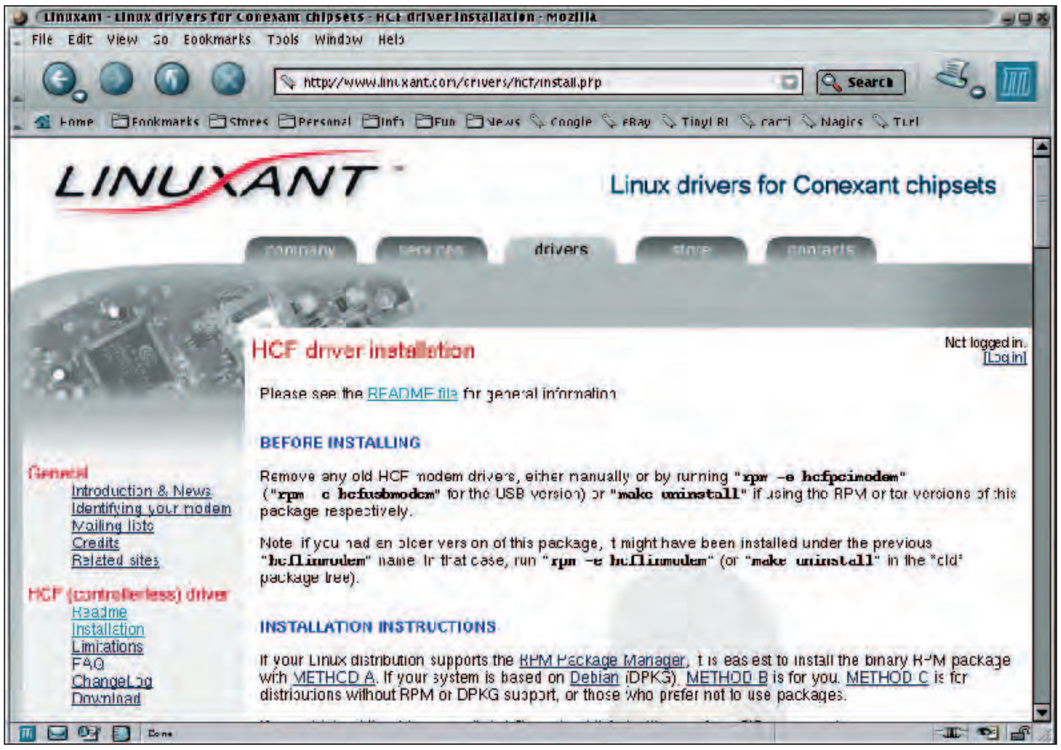
2 Which is the best formatting system to use for speed ... ext3 or ReiserFS; or maybe the new ReiserFS that is due out?

3 Just how big should the swap partition be for best speed performance... 1.5x RAM, 2x RAM, 3x RAM ? and would this have any affect on bootup speed?

4 Is there any practical way to cut down on the bootup time?

Vic Woolnough, *via email*

A Waiting over two minutes to check module dependencies is ridiculous! However, it may well be performing another function during this time. Running **depmod -a** will rebuild the module dependencies for you, so comparing the time to perform the two functions should indicate the cause of the problem. It could very well be a problem with the disk itself causing slow reads, or the CPU is becoming swamped



Linuxant - Linux drivers for Conexant chipsets - HCF driver installation - Mozilla
File Edit View Go Bookmarks Tools Window Help
http://www.linuxant.com/drivers/hcf/install.php
Search
Home Bookmarks Stores Personal Info Fun News Google eBay TinyURL cart Nagios Tux
LINUXANT
Linux drivers for Conexant chipsets
company services drivers store contacts
Not logged in.
[Login]
General
Introduction & News
Identifying your modem
Mailing lists
Credits
Related sites
HCF (controllerless) driver
Readme
Installation
Limitations
FAQ
ChangeLog
Download
Done

HCF driver installation

Please see the README file for general information

BEFORE INSTALLING

Remove any old HCF modem drivers, either manually or by running "rpm -e hcfpcimodem" ("rpm -e hcfusbmodem" for the USB version) or "make uninstall" if using the RPM or tar versions of this package respectively.

Note: if you had an older version of this package, it might have been installed under the previous "hcflinmodem" name. In that case, run "rpm -e hcflinmodem" (or "make uninstall" in the "old" package tree).

INSTALLATION INSTRUCTIONS

If your Linux distribution supports the RPM Package Manager, it is easiest to install the binary RPM package with METHOD A. If your system is based on Debian (DPKG), METHOD B is for you. METHOD C is for distributions without RPM or DPKG support, or those who prefer not to use packages.

Conexant has kernel modules available for its popular USB ADSL modem, so we can use it to connect to the Internet from Linux.

performing another function while it is working out module dependencies.

The speed of a distro depends upon a number of factors, including the kernel you run, the different services you run in the background, and how efficiently you make use of your resources. As you don't have much memory or CPU, running a very graphically intensive Linux distribution is going to make the system run very slow, as would running everything from a mail server to a web server on the system, gobbling up valuable memory.

ext3 is a good all-round filesystem to use with Linux, and is particularly quick for both reads and writes. ReiserFS is not as quick for writes, however it is much faster for reads. ext3 is generally a good choice, as one can mount it ext2 for emergency system recovery processes.

Fedora or Debian are good distros to install on a slower system – either of these distros should run quite happily on the system you have. Both allow you to build a fairly efficient installation with recent software and updated security patches.

The amount of swap space is entirely dependent upon the apps you run, although double the amount of RAM is usually a minimum. More swap space is not going to help your boot time at all, as your system will not be consuming enough memory to even touch the swap space at that point of the process.

Red Hate

Q I installed Red Hat 9 to try out, as I would like to get rid of Windows, but I can't seem to get my Etec USB ADSL to work. Red Hat 9 knows it is there, as it shows up in the hardware listings. It use the Connexant chipset and to get it working under Windows, you first open a browser, then type in 10.0.0.2, then the password, then type in your details from your ISP; but it's not the same with Linux. Please help if you can – would I be better of buying a router to get to work that way?

George Cameron, *via email*

A Drivers for your ADSL modem are not a standard part of the Red Hat Linux distribution, so you will have to download the kernel modules from **http://linuxant.com/**. They install as standard RPMs, so they should be fairly easy to get working and there is documentation available at **www.linuxant.com/drivers/hcf/install.php** explaining what exactly you need to do in order to get them installed. In addition to the kernel modules, there is a command-line application that is used to configure the modem and ensure that it can connect to your DSL service provider correctly. Of course, you will require a functional USB system that currently detects the DSL modem connected to your system prior to installing the modules.

Connect!

Q Please can you answer a stupid question: I have an existing Internet Service Provider – CIX. Access to the Internet is currently through its own desktop, Ameol, via Windows. I am sick and tired of it and have been playing about with Linux for a while; it's time to get serious. Can I install Linux (Xandros) and still access the Web by using the connect number and passwords of my ISP?

If not, any recommendations as to who I could use?

Michael Threlfall, *via email*

A You can dial-up to any ISP that uses *SLIP* or *PPP* for connectivity. If CIX offers this form of connectivity, then you will be able to use it for Internet access from within Linux. If you are unable to connect to CIX using *PPP* or *SLIP*, then you will have to search for an alternative ISP. Almost all popular ISPs – other than AOL – will permit you to connect to them through Linux, although it is likely an unsupported method of connection.

rackspace
MANAGED HOSTING

Samba share

Q **I regularly copy binary files from my windows machine to a *Samba* share on my Linux server. There are 300+ files, and the problem is they all get copied over in uppercase letters. Is there a way with command-line that I can move them to small caps? The Cobol program I run needs them to all be small and renaming each file is getting quite cumbersome.**
Jay, *via email*

A There are quite a few different ways of doing this. Probably the quickest and simplest way would be to use the *tr* utility coupled with some simple shell syntax:

```
#!/bin/bash
for file in *
do newfile=`echo $file|tr [A-Z] [a-z]`
  echo "Moving $file to $newfile"
  mv $file $newfile
  echo Done
done
```

This script will replace all the uppercase characters in any filename in the current directory to lowercase. I've not built in any testing to see if the filename is already lowercase, so this script will spit out an error if it is, but it should continue to run through all the other files.

Do It Yourself

Q **I'm actually also a Rackspace customer, but I thought I would send this question to you via the *Linux Format Answers* pages as I don't need a resolution right away, and I'm sure others would find a use for the answer. I am running Red Hat 7.3. My server only has one network card and I need to add some extra IP addresses to the card. How is this done? I know Rackspace has done this for me previously, but I would like to try to do this myself on my staging server at my office.**
A Reader, *via email*

A Yes this is possible and quite easy to do. If for instance, your network card is eth0. You can check this by typing

```
ifconfig -a
```

On Red Hat 7.3, all network configuration scripts are stored in /etc/sysconfig/network-scripts, and depending what that interface is named there will be a corresponding file called 'ifcfg-????'. In this case the file would be 'ifcfg-eth0'. To create a new sub-interface, you need to create a new file. I would recommend copying the file for eth0 as a template, but it's just as good to create it from scratch. The syntax to do this is as follows:

```
# cd /etc/sysconfig/network-scripts
# cp ifcfg-eth0 cp ifcfg-eth0\:1
```

The filename for any sub-interfaces should be called ifcfg-eth0:? – where ? is the number of the subinterface. In our example, this is eth0:1. Next, edit this file using your favourite editor. It should look something like this:

```
DEVICE=eth0:1
ONBOOT=yes
BOOTPROTO=static
IPADDR=192.168.0.1
NETMASK=255.255.255.0
```

Once the changes have been made, simply use **ifup** to bring up the interface. In this case **ifup eth0\:1**. You can check if the interface is up by running **ifconfig -a** again.

Remote virtual host

Q **I have set up a virtual host on my server and I can access it from any computer's browser on the network. (pointing the browser to the web page: subdomain.domain.com or simply subdomain). The problem arises when I try to access this subdomain site from outside the network – from a dialup connection at home. How would I go about having this virtual host viewable from outside my private network without registering it with my domain registrar?**

Is there any way that I could set up the server to handle and resolve requests to this virtual host? It does not make sense to me to have this virtual host registered with a name server when my server could handle it directly!

I hope I've made my question clear enough for you to understand!
Hajo, *via email*

A If I understand your question correctly, then you are asking if it is possible for the server that domain.com points to, to be able to supply DNS information for any domains or hosts under it. Unfortunately this is not possible without telling the authoritative name servers (*eg* those of your registrar) that there are other nameservers who are able to provide resolution for other parts of this domain.

For instance - the NS records for your domain probably look something like this:

```
domain.com.    IN    NS
ns1.registrar.com.
domain.com.    IN    NS
ns2.registrar.com.
```

In order to add a subdomain over which you have control you will need to get a new NS record added as in the next:

```
subdomain.domain.com.          IN
NS          192.168.1.1
```

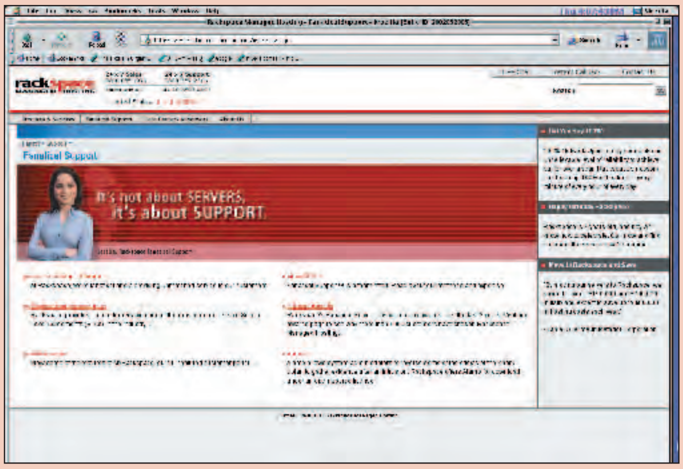
Here we have said that 192.168.1.1 is an authoritative nameserver for subdomain.domain.com. We can now set up a DNS server on 192.168.1.1 and give it records for subdomain.domain.com *eg*:

```
subdomain.domain.com.
IN          A          192.168.1.100
subdomain.domain.com.
IN          MX
subdomain.domain.com.
another.subdomain.domain.com IN
CNAME subdomain.domain.com.
```

and so on. If you are not able to get records added to the authoritative nameservers though, there is very little you can do – this is for security reasons and is not a limitation of DNS.

ALL ABOUT RACKSPACE

When it comes to managed hosting Rackspace is unique. We are the only hosting company to guarantee 100 per cent network uptime and we will even pledge to replace faulty hardware (let's face it, it does happen) within an hour. As you would expect from Red Hat's only Advanced Hosting partner you will receive Fanatical Support™ and instant emergency response from fully qualified level three technicians, available 24-hours a day, 365 days a year. No answering machines, no diversions, no silly on-hold messages. All our customer platforms are housed in state-of-the-art, secure data centres and can be fully customised to meet any requirement. A dedicated account manager is allocated to you at the outset and will remain on your account. All Rackspace employees are committed to ensuring customer satisfaction, and will not rest until a client problem is resolved. Since Rackspace was established in 1998, it has specialised in managed hosting – nothing else. We're not an ISP, we don't sell domain names, and we won't offer website design – all we do is managed hosting. As a result, we do it very well and 97 per cent of our customers would happily recommend us. For information on how we can improve your web hosting, please contact us at www.rackspace.co.uk and ask any questions you may have about the ultimate managed hosting service.



WIN an Archos AV140 Video Recorder WITH RACKSPACE MANAGED HOSTING

www.rackspace.co.uk

Every month, the best question related to Systems Administration that a *LXF* reader sends in wins a prize: this month, you have the chance to win the awesome Archos AV140 Video Recorder.

The AV140 combines a modular MP4 video player and recorder, MP3 music player and recorder, digital camera and camcorder, digital photo wallet and data storage in a compact device that fits in the palm of your hand and is compatible with Linux. Based on a 40GB hard disk that is completely compatible with Linux, Windows and Mac OS, all you have to do is connect it to your computer and it will be mounted as an additional hard disk – on which you can store *any* type of data. USB 2.0 auto-connection ensures fast data transfer to and from your PC.

You can now enjoy 160 hours of near-DVD quality MP4 video with MP3 sound wherever you are. You can record and playback to your TV, VCR or camcorder directly in MP4 format. Playback is an AVI format file with MPEG4-SP VHS quality video and MP3 stereo sound, with the ability to record up to 2,000 hours of voice audio. See www.archos.com/products/prw_500432_specs.html.

Email all your sysadmin questions and problems to our expert system administrator Hans Huberland: sysadminqa@rackspace.co.uk

Every published query on these system admin pages wins a Rackspace T-shirt!

★Star Question – AV140 winner!

This issue's lucky winner is ***Les*** – your new portable multimedia player/recorder will be with you shortly!

I'm building a high availability cluster for personal use, and I'm using the following shell script (or variance of) which *SSHs* or *telnets* into other machines and starts processes, which then broadcast their display to my local machine (all the others are headless). These scripts are linked to icons on my desktop, so I can just point and click to start a remote application.

```
#! /bin/sh
xhost +192.168.0.2
{
sleep1
echo "username"
sleep 2
echo "password"
sleep 2
echo "DISPLAY=192.168.0.1 xterm &
sleep 1
echo "application-name"
sleep 5000 #application expiry time
}| ssh 192.168.0.2
```

The problem with this is, if there are a lot of processes running on any given machine, this slows down the responses for the commands issued by the shell script. So, for example, the script provides the password before asked, authentication fails, and the application fails to start. The only way to get around this is to set sleep to around 7 or 8. This is to compensate for when hosts are under high load.

Is there an easier and faster way to do this? There is no need for authentication, as all hosts on the network are trusted. Your thoughts and suggestions would be truly appreciated!

Les, *via email*

A Hi Les, thanks for a very interesting question. In order to simplify the authentication process, I would recommend using Public Key Authentication for *SSH*. To do this you will need to generate a public/private key pair on your client. The public key on your client needs to be saved onto the server. To enable public key authentication make sure that the following lines exist in /etc/ssh/sshd_config:

```
PubkeyAuthentication yes

AuthorizedKeysFile
.ssh/authorized_keys
```

Generating the public/private key pair varies from system to system. In Red Hat Linux, you can use the **ssh-keygen** command. In your case, as you are on a secure network, I will go without recommending that you set up a passphrase as strongly as I normally would. Remember though, if you have no passphrase and somebody accesses your workstation and steals the private key, they will have as much access to your server as you do. If you still want to eliminate the requirement for an interactive password, then you can use the command bellow to generate the keypair:

```
ssh-keygen -t dsa
```

This will create a .ssh directory with a public and private key. Once you have generated the keys you need to put the public key into the .ssh/authorized_keys file in the home directory (on the server) of the remote user you would like to **ssh** in as. Assuming you want to log in as root, you should now be able to just use:

```
ssh -l root 192.168.0.2
```

Find and replace

Q I host a website with technical information for the industry I work in. It has recently been brought to my attention that I'm linking to certain pages 'illegally' (it is still questionable whether linking to another site is illegal, but this is a whole other kettle of fish). In the meantime, to avoid causing any unnecessary hassle, I'd like to remove all links to this site and replace them with a DELETED keyword or something similar – that way I could fairly easily add them back at a later date when the legality issues have been resolved. The problem is that I probably have hundreds of pages with links to this site. Do you have a simple syntax I can use to globally find and replace all these strings in all these files?

Rob, *via email*

A This is a fairly straightforward job for Perl, with a little help from regular expressions. With the following command-line syntax, you should be able to change all the files in a single directory:

```
perl -pi -e "s/original
string/replacement string/g"
```

Outbound SMTP

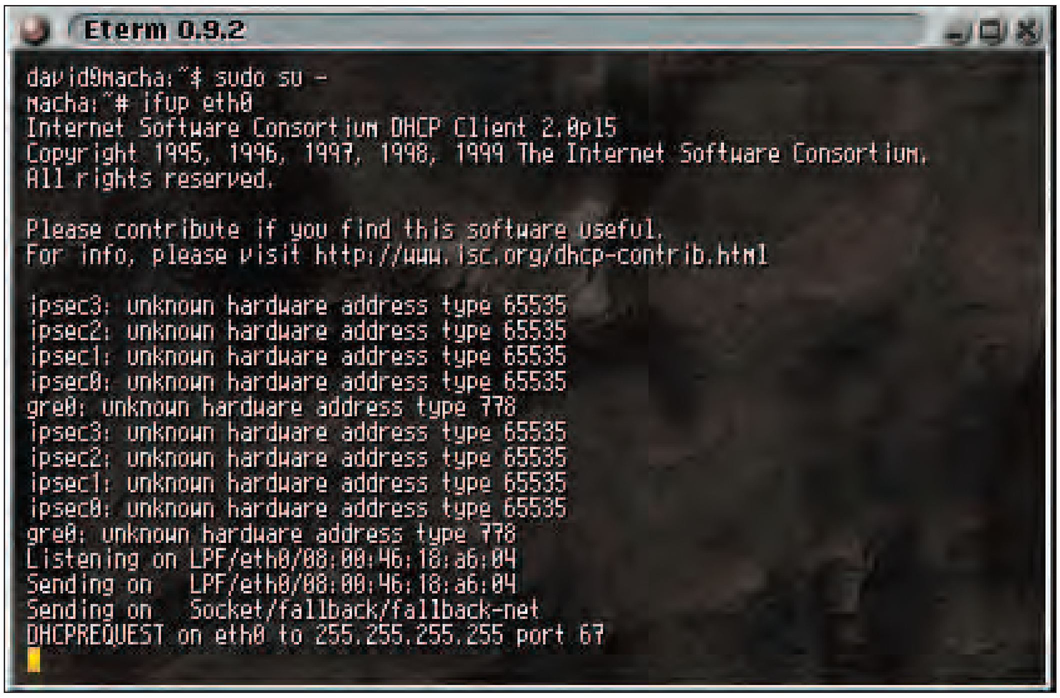
Q **I have recently installed Tiscali broadband using a Sagen F@st800 modem on an Athlon XP 2100+, 512MB RAM machine running Mandrake 9.2. Now the broadband connection works fine; using *Mozilla1.4*, I can surf the Internet, collect my email via the POP3 server but I can't send email. Before the broadband connection I was using a dialup connection via a 56KBps modem using *KPPP* to manage my dialup and this worked fine but since I started using the broadband connection I just cannot send email.**

Just to prove my point to myself I ran up the modem connection again and – lo and behold – I could send email but going back to broadband no joy. The broadband connection is using eth0 and ppp0 and as I can surf, so I must be able to send and receive on the Ethernet port – it cannot be a permissions thing; and as I can send email using the modem connection, I'm also assuming it's not *Mozilla*'s fault. So what else is there? Is there a *Mozilla* or Linux configuration file I need to change that tells it which device or iNet address to use to send mail?

Any help you could give me to solve this annoying mystery would be greatly appreciated.

David M Swain, *via the LXF forums*

A ISPs permit relaying through their SMTP servers based upon the IP address that you are using. If you have an IP from the Tiscali network, then you will be able to relay through its SMTP server and send mail, but if you try to send through Tiscali's server from Freeserve, then you will find that it denys relaying for you. The only way around this issue is to adjust your settings in *Mozilla* to reflect your new ISP and put their SMTP server in the configuration for outbound SMTP. This, of course, makes it somewhat difficult if you are switching from one ISP to another; however, you can run an SMTP server locally and relay directly to the destination. One downside of this is that you may find that your connections are denied if they are using a dial-up 'black hole' list, but modifying your SMTP server's configuration based upon which ISP you are connected with should be a trivial script as part of the scripts which bring the applicable network interfaces up.



```
david@macha:~$ sudo su -
macha:~# ifup eth0
Internet Software Consortium DHCP Client 2.0pl5
Copyright 1995, 1996, 1997, 1998, 1999 The Internet Software Consortium.
All rights reserved.

Please contribute if you find this software useful.
For info, please visit http://www.isc.org/dhcp-contrib.html

ipsec3: unknown hardware address type 65535
ipsec2: unknown hardware address type 65535
ipsec1: unknown hardware address type 65535
ipsec0: unknown hardware address type 65535
gre0: unknown hardware address type 778
ipsec3: unknown hardware address type 65535
ipsec2: unknown hardware address type 65535
ipsec1: unknown hardware address type 65535
ipsec0: unknown hardware address type 65535
gre0: unknown hardware address type 778
Listening on LPF/eth0/08:00:46:18:a6:04
Sending on   LPF/eth0/08:00:46:18:a6:04
Sending on   Socket/fallback/fallback-net
DHCPREQUEST on eth0 to 255.255.255.255 port 67
```

Obtaining a DHCP address for cable Internet requires the modem to sync up to the cable network first, which can take a while. Patience is a virtue!

Laptop Linux

Q **I have an old laptop, a 486 SX20, that I was thinking of installing Linux on. For some time now, the only thing I have used it for was word-processing when travelling. Nowadays, it's too slow and small for any web access; I have been wondering if there is a Linux I can use on it?**

I would like to suggest that it would be useful to readers if you put a page on your website with links to recommendations for older systems, and what distro should be used on them. This could be a permanent fixture, with updates only necessary when there is a significant change that becomes available. I don't expect that would be too often on an old system like mine!

Bob Klahn, *Sylvania, Ohio, USA*

A You can certainly install a distro such as Debian on your laptop, although depending up much memory you have available, you may not be able to run X on it. Certainly, you will be limited as to what you can run on the device, as a word processor such as *StarOffice* or *OpenOffice.org*, as there will be insufficient disk or memory available for such a process. The best you are likely to get is a minimal X installation, through which you can run command line processes, or a command line system. While it is certainly true that Linux works better than most other operating systems on older hardware, it does depend what your end goal is as to if it's actually worth the effort.

DHCP woes

Q **I am running SUSE 9 – when I boot up I get a message that says: "No IP address yet backgrounding" so I can't use the network or the cable modem; but if I reboot, then it's always alright, but never on the first boot. Any ideas?**

A Reader, *via the LXF forums*

A It can take a cable modem upwards of a couple of minutes to sync up to the cable network, allowing it to pass IP traffic onto your cable provider. Until the cable modem is fully synced, you will be unable to DHCP yourself an IP address. So, you may want to leave the cable modem turned on all the time, so when you boot your PC up, the cable modem is already connected to the cable network and you can reliably get an IP address every time. LXF

Posting to the forum

The *LXF* online community

Not only do our popular forums at **www.linuxformat.co.uk** have sections dedicated to your technical queries, hardware, programming languages and general help; but also there's always a lively Linux discussion going on!

Submission advice

We are happy to answer all sorts of Linux-related questions. If we don't know the answer, we'll find out for you! But in order to give you the best service, it helps a lot if you read the following submission advice.

- Please be sure to include any relevant details of your system. *"I can't get X to work"* doesn't really mean anything to us if we don't know things like what version of X you are trying to run, what hardware you are running on.
- Be specific about your problem. Things like *"it doesn't work"* or *"I get an error"* aren't all that helpful. In what way does something not work? What were you expecting to happen? What does the error message actually say?
- Please remember that the people who write this magazine are NOT the authors or developers of Linux, any particular package or distro. Sometimes the people responsible for software have more information available on websites *etc.* Try reading the documentation!

We will try to answer most questions. If we don't answer yours specifically, you'll probably find we've answered one just like it. We regret that we can't really give personal replies to all your questions.

WRITE TO US AT:
***Linux Format*, Future Publishing, 30 Monmouth Street, Bath BA1 2BW** or
email: **lxf.answers@futurenet.co.uk**

PRIORITY ORDER FORM:

□ **I would like to subscribe/renew to *Linux Format* (please delete as appropriate): Select CD or DVD option.**

	UK	Europe/N.America	Rest of World
□ **CD**	£54.90	£78.90	£91.90
□ **DVD**	£59.90	£83.90	£96.90

YOUR DETAILS

Title________ Initials________ Surname ______________________________

Address ____

____

Postcode ______________ Country ______________________

Email address______________________________ □ business □ home

Telephone number ___________________________ □ business □ home

Mobile ____

Your subscription will start with the next available issue

We will use the contact details supplied to communicate with you about your subscription. Indicate here if you're happy for *Linux Format*, Future Publishing and its sister companies to contact you by: □ email □ mobile If you're also happy for us to pass your details on to carefully selected companies indicate here: □ email □ mobile

PAYMENT DETAILS [please tick]

I understand that I will receive 13 issues during the next 12 months.

□ I enclose a cheque for __________________ made payable to Future Publishing Ltd

I wish to pay by

□ Visa □ MasterCard □ American Express

□ Switch: issue no./valid date ______________________________

Card number

□□□□□□□□□□□□□□□□□□□□

Expiry date______________

Signature____________________________________ Date______________

□ Direct Debit (UK only – double CD Edition) – I wish to pay £13.73 every 3 months

or □ Direct Debit (UK only – DVD Edition) – I wish to pay £14.98 every 3 months

Please complete the form below

Instruction to your Bank or Building Society to pay Direct Debits

future publishing
Media with passion

DIRECT Debit

Originator's Identification Number

7	6	8	1	9	5

Future Publishing Ltd. Cary Court, Somerton. Somerset. TA11 6BR

Please send this form to address shown

1. Name and full postal address of your Bank or Building Society branch

To: The Manager

____Bank/Building Society

Address____

________________________________ Postcode ______________

2. Name(s) of account holder(s)

3. Branch sort code
(from the top right-hand corner of your cheque) □□□□□□

4. Bank or Building Society account number □□□□□□□□

5. Instruction to your Bank or Building Society

Please pay Future Publishing Direct Debits from the account detailed on this Instruction subject to the safeguards assured by the Direct Debit Guarantee. I understand that this instruction may remain with Future Publishing and if so, details will be passed electronically to my bank or building society.

Signature(s)________________________________Date______________

Ref No (Office use only)________________________________

Banks and Building Societies may not accept Direct Debit Instructions for some types of account.

You may photocopy this blank form and then complete it. You cannot fax or email Direct Debit forms.

UK READERS: *Linux Format*, Subscriptions, Future Publishing Ltd, Freepost BS4900, Somerton, Somerset, TA11 6BR

OVERSEAS READERS: *Linux Format*, Subscriptions, Future Publishing Ltd, Cary Court, Somerton, Somerset, TA11 6TB, United Kingdom

Please indicate if you do NOT wish to receive relevant information about special offers or products/services from *Linux Format*, Future Publishing or any of its sister companies by □ post or □ telephone. Please indicate if you do NOT wish us to pass your details on to other carefully selected companies to enable them to contact you about their products/services by □ post or □ telephone.

CLOSING DATE 9.06.04 **OFFER CODE** LXFXP54

LINUX FORMAT

» 3 EASY WAYS TO ORDER:

1 PHONE 0870 444 8645

2 POST Future Publishing Ltd, Freepost, TA11 6BR

3 FAX this completed form to: 01458 271146

OVERSEAS CUSTOMERS CALL: +44 870 444 8645

SUBSCRIBE AND SAVE!

SAVE 29% OFF THE COVER PRICE

That's 13 issues for the price of 10
Don't miss out, call today

- ✔ Save over £24* on the price you'd pay in the shops
- ✔ FREE delivery direct to your door
- ✔ Never miss an issue!

LINUX FORMAT BRINGS YOU:

1 Essential reading for all Linux users
2 20-page *Linux Pro* supplement with every issue
3 Three CDs or a DVD packed with Linux software
4 Latest news, latest reviews and the best Linux tutorials around

Taking out a subscription or giving a gift is easy. Simply call **0870 444 8645** or complete the order form opposite.

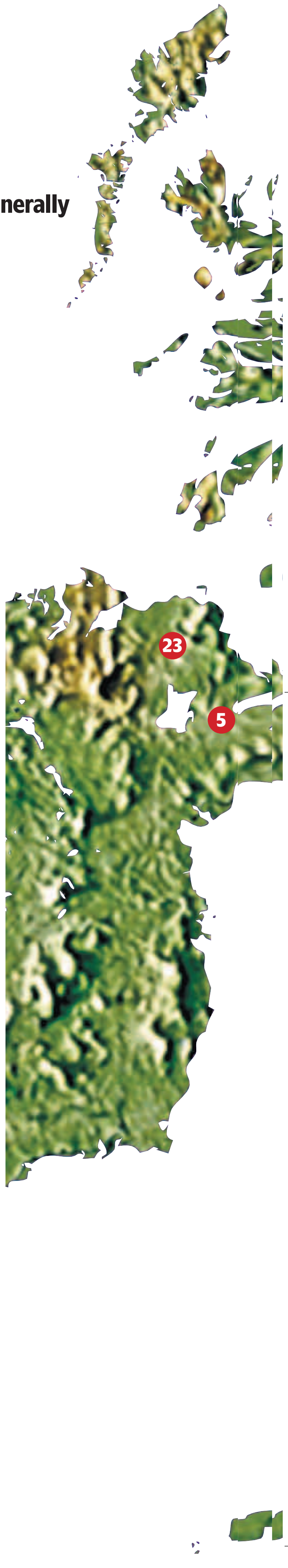
LINUX FORMAT
MAKING MOV
TURN OVER FOR YOUR FREE LINUX PRO MAGAZINE
MANDRAKE 10.0
BUILD THE
BOX!
STOP SPAM!
PLUS PREVENT VIRUSES • HALT HACKERS • GET PROTECTED
ON THE DISCS:
LIVE DISTROS!

*£24 saving based on price of DVD issue.

ORDER HOTLINE ☎ 0870 444 8645

User Groups

LUGs worldwide are full of members keen to help with your problems, discuss ideas, and generally natter about all things Linux. You can find lots more information online at www.lug.org.uk



1 ABERDEEN
www.aberdeen.lug.org.uk/

2 ANGLIAN LUG
www.alug.org.uk/

3 AYLESBURY
drbond@educational-computing.co.uk/

4 BEDFORDSHIRE
www.beds.lug.org.uk/

5 BELFAST & N. IRELAND
bgalbraith@gmx.net/

6 BERKSHIRE & THAMES VALLEY
www.sclug.org.uk/

7 BEVERLEY
www.beverley.lug.org.uk/

8 BIRMINGHAM (SOUTH)
www.sb.lug.org.uk/

9 BLACKBURN
matt@consultmatt.co.uk/

10 BLACKPOOL
www.blackpool.lug.org.uk/

11 BORDERS (ENGLAND/SCOTLAND)
welby@mcrobers.com.kg

12 BRIGHTON
www.brighton.lug.org.uk/

13 BRISTOL & BATH
www.bristol.lug.org.uk/

14 BRIXTON
blug@communitytechnology.org.uk/

15 CAMBRIDGE
mark@taurine.demon.co.uk/

16 CHESHIRE (SOUTH)
www.sc.lug.org.uk/

17 CHESTERFIELD & DISTRICT
www.spirelug.org.uk/

18 CHICHESTER HARBOUR – MARITIME GEEKS
hugh@mjr.org

19 CLEVELAND OPEN SOURCE GROUP
www.cleveland.lug.org.uk/

20 COVENTRY
www.covlug.org.uk/

21 CUMBRIA
www.cumbria.lug.org.uk/

22 DERBYSHIRE (SOUTH)
www.sderby.lug.org.uk/

23 DERRY
www.derry.lug.org.uk/

24 DEVON & CORNWALL
www.dclug.org.uk/

25 DONCASTER & SCUNTHORPE
www.doncaster.lug.org.uk/

26 DORSET
www.dorset.lug.org.uk/

27 DUNDEE & TAYSIDE
www.dundee.lug.org.uk/

28 EDINBURGH
www.edlug.org.uk/

29 EXETER (UNIVERSITY)
nick@urgusabic.net/

30 FALKIRK
www.falkirk.lug.org.uk/

31 GLASTONBURY
www.glastonbury.lug.org.uk/

32 GLOUCESTERSHIRE & COTSWOLDS
www.gloucs.lug.org.uk/

33 HAMPSHIRE
www.hantslug.org.uk/

34 HEREFORD
www.hereford.lug.org.uk/

35 HERTFORDSHIRE
www.herts.lug.org.uk/

36 HERTFORDSHIRE (EAST)
www.ear.lug.org.uk/

37 HUDDERSFIELD
www.hud.lug.org.uk/

38 HULL
www.hull.lug.org.uk/

39 ISLE OF MAN
www.iom.lug.org.uk/

40 ISLE OF WIGHT
www.iow.lug.org.uk/

41 KENT
www.kent.lug.org.uk/

42 LANCASTER
www.lancasterlug.org.uk/

43 LEAMINGTON SPA
ian@siim.info

44 LEICESTER
www.leicester.lug.org.uk/

45 LINCOLNSHIRE
www.lincs.lug.org.uk/

46 LONDON (GREATER - GLLUG)
www.gllug.org.uk/

47 LONDON (LONIX)
www.lonix.org.uk/

48 LONDON (NORTH-WEST)
www.northwestlondon.lug.org.uk/

49 LONDON (SOUTH)
www.sl.lug.org.uk/

50 LOUGHBOROUGH
www.loughborough.lug.org.uk/

51 LUTON & ST ALBANS
www.lust.lug.org.uk/

52 MALVERN
www.malvern.lug.org.uk/

53 MANCHESTER
www.manlug.mcc.ac.uk/

54 MANSFIELD
www.mansfield.lug.org.uk/

55 MENAI
kevin@dotmon.com

56 MILTON KEYNES
www.mk.lug.org.uk/

57 MORAY
Stu@forres.co.uk

58 NORTH EAST ENGLAND (DURHAM)
www.ne.lug.org.uk/

59 NORTH LONDON
jason@voyagercomputers.co.uk

60 NORTHANTS
www.northants.lug.org.uk/

61 NOTTINGHAM
www.nottingham.lug.org.uk/

62 NUNEATON
www.nuneaton.lug.org.uk/

63 ORMSKIRK
www.ormskirk.lug.org.uk/

64 ORPINGTON
www.orpington.lug.org.uk/

65 OXFORD
www.oxford.lug.org.uk/

66 OXFORD BROOKES UNIVERSITY
www.obaoss.lug.org.uk/

67 PETERBOROUGH
www.peterboro.lug.org.uk/

68 PRESTON & LANCASHIRE
www.preston.lug.org.uk/

69 RUGBY
www.rugby.lug.org.uk/

70 SALFORD (UNIVERSITY)
http://linsoc.ussu.salford.ac.uk/

71 SCARBOROUGH
www.scarborough.lug.org.uk/

72 SCOTTISH
www.scottish.lug.org.uk/

73 SHEFFIELD
www.sheflug.org.uk/

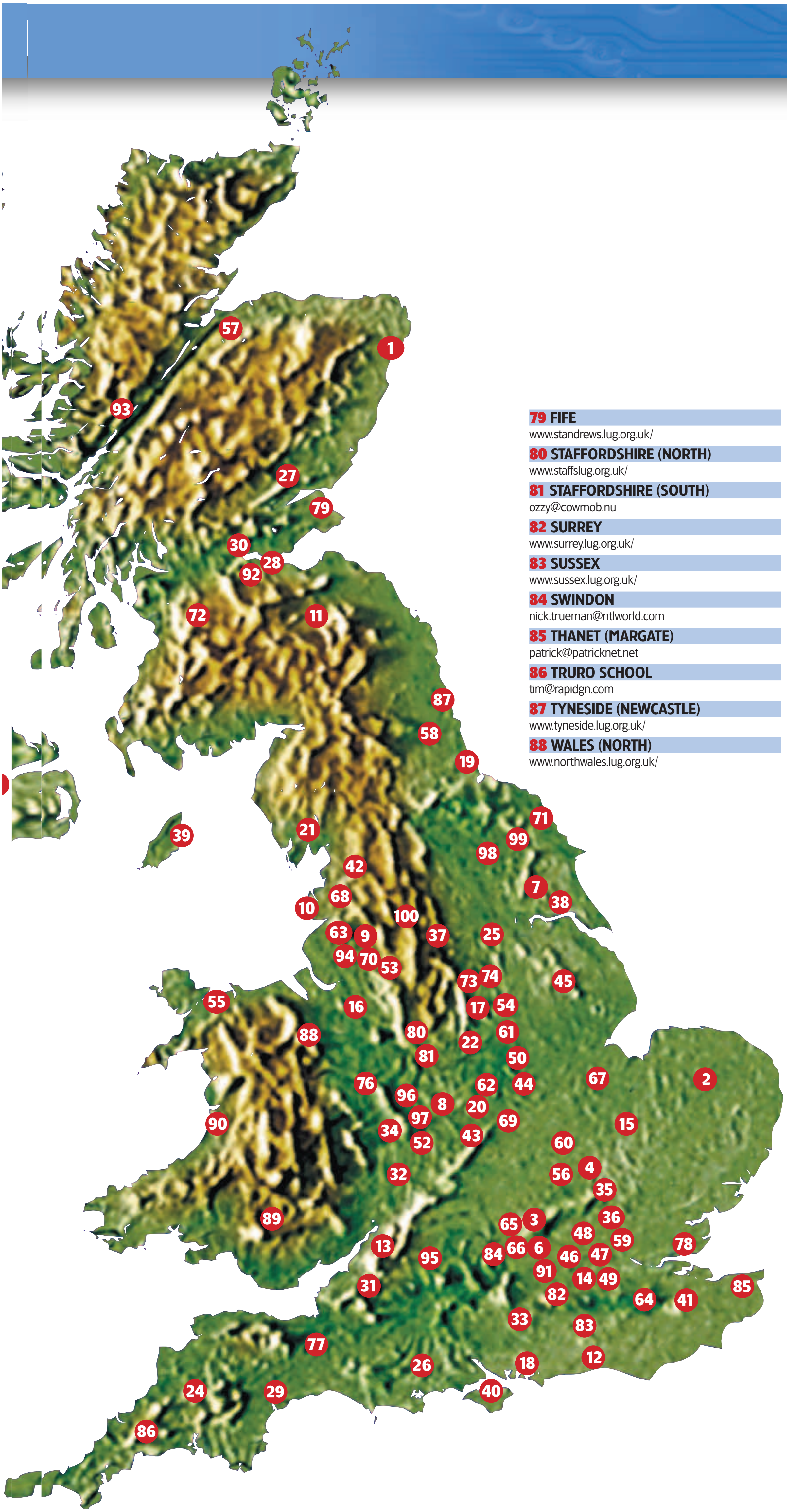
74 SHEFFIELD LINUX & BSD USER GROUP
www.slugbug.org.uk/

75 SHETLAND
www.shetland.lug.org.uk/

76 SHROPSHIRE
www.shropshire.lug.org.uk/

77 SOUTH WEST
www.southwest.lug.org.uk/

78 SOUTHEND-ON-SEA
derek@techsys.co.uk



79 FIFE
www.standrews.lug.org.uk/

80 STAFFORDSHIRE (NORTH)
www.staffslug.org.uk/

81 STAFFORDSHIRE (SOUTH)
ozzy@cowmob.nu

82 SURREY
www.surrey.lug.org.uk/

83 SUSSEX
www.sussex.lug.org.uk/

84 SWINDON
nick.trueman@ntlworld.com

85 THANET (MARGATE)
patrick@patricknet.net

86 TRURO SCHOOL
tim@rapidgn.com

87 TYNESIDE (NEWCASTLE)
www.tyneside.lug.org.uk/

88 WALES (NORTH)
www.northwales.lug.org.uk/

89 WALES (SOUTH)
www.swlug.org.uk/

90 WALES (WEST)
www.westwales.lug.org.uk/

91 WALTON ON THAMES
rael@freeuk.com

92 WEST LOTHIAN
ben@150bpm.co.uk

93 WEST OF SCOTLAND
woslug@uk2.net

94 WIGAN AND ST. HELENS
www.all-the-johnsons.co.uk/wishlug/

95 WILTSHIRE
www.wiltshire.lug.org.uk/

96 WOLVES
www.wolveslug.org.uk/

97 WORCESTERSHIRE
www.worcs.lug.org.uk/

98 YORK
www.york.lug.org.uk/

99 EAST YORKSHIRE
sharkonline@whatemail.com

100 WEST YORKSHIRE LUG
www.wylug.org.uk/

OTHERS

BENELUX
www.b-lux.org/

BUSINESS LUG
www.businesslug.org/

CHANNEL ISLANDS
myles.hartley@db.com

FUENGIROLA GNU/LUG
www.fuengirola.lug.org.uk/

SCHOOLS
www.schools.lug.org.uk/

STUDENTS
www.students.lug.org.uk/

TR LINUX
www.trlinux.lug.org.uk/

UK UNIX USER GROUP (UKUUG)
www.ukuug.org/

YOUNG LINUX USER GROUP
www.young.lug.org.uk/

LUG OF THE MONTH

Iraqi Linux User Group

Hasaneen writes: Supported by AL-Nahrain University's Computer Engineering Department, the Iraqi Linux User Group gave its first seminar aimed at educating people in Iraq about Open Source and Linux concepts in Baghdad on 17 February 2004 , from 10:00 AM to 12:30 PM. Presented by Ashraf Tariq, our LUG introduced the audience to:

- Open Source, GNU and GPL license.
- Brief History of Linux
- Why Linux?
- The Value of the source
- Some Core Features of Linux.
- Linux & Security (by Sarmad).
- Linux Q&A by Ashraf and Hasanen.
- Practical presentation – installing Mandrake 9.2 on a workstation.

We distributed Mandrake CDs to the audience, most of whom were students; also there were two professors from the college and other interested people.

Some aims of Iraqi LUG are:

- Educate Iraqis and businesses on the principles and benefits of Open Source.
- Cultivate IT culture in Iraq to join the global outsourcing community.
- Provide solutions for schools, colleges, universities, Open and distance learning IT setups.
- Provide programming resources covering Java, C++, and web.
- Provide solutions for SME (Small to Medium Enterprise) IT setups.
- Offer a registration service for people and organisations inside Iraq for Linux consultancy and services.
- Repository of information, software, and links to latest Linux, Arabic Linux and Open Source components.
- Open and distance learning.

Our website is mostly in English, and we have an international membership, with members in the UK, Canada, Germany – as well as those inside Iraq – championing the adoption of Open Source in Iraq wherever possible.

If you as an individual or group think that you could help us acheive our aims – or that we could assist with yours, please get in touch through our website at **www.iraqilinux.org/index.php**

WORLDWIDE LINUX USER GROUPS

Help needed from global Free/Open Source Software users

Our brand-new UK map marks the beginning of *Linux Format*'s efforts to get more involved with Linux User Groups communities, be it in the UK, or worldwide. We have one problem though: we can't agree on how we can do this! Following issue 52's very brief appeal at the end of *The Great 2004 LUG Explosion*, we've had several good ideas already, that we repeat this month as an example of the kind of suggestions that we want:

- Mumbai reader **Vishal Kumar** suggested that some worldwide LUGs – particularly in Asian countries like Thailand where Linux adoption is being fostered by local governments – might be able to benefit from having the opportunity to be subscriptions agents and/or distributors of our magazine in any areas of the world where *Linux Format* isn't readily available on the news-stand. We're exploring ways of making this happen as you read this: if your LUG would be interested in taking part in such a scheme, please do get in touch as soon as possible – we'd like to hear from you so we can get a very rough idea of numbers.

- **Adler Milstrey** emailed us from Berlin with the suggestions that user groups in wealthier states could practice some sort of 'twinning' scheme with LUGs in developing areas. If you've got any ideas as to how *Linux Format* could help facilitate this, let's hear them!

- **Dougal Greenwood** from New South Wales, Australia maintains that a major factor in holding back Linux adoption for many computer users is that they aren't aware of the fact that you can run Windows and Linux side-by-side. He suggested that a good way of publicising the existence of LUGs to people in your local area could be to co-ordinate a 'Windows Amnesty Day', whereby members of the public bring their PCs along and are helped through making their own setup dual-boot with both Windows and Linux by LUG gurus.

LXF reckons that this idea gives some much-needed branding and direction to the otherwise limited appeal of the 'Linux Install Day' idea – if maybe this were organised on a national or international footing, and perhaps sponsored by Linux software vendors, we suspect that publicity would be quite easy...

So now it's down to you! Got an event happening? Need another LUG to send you some distro CDs from overseas? If you don't send the details in, we can't publicise them! If you have any ideas about what *Linux Format* can do to help LUGs, *eg* suggestions for topics, regular sections or events to be covered on these pages, you don't have to be a LUG member to contribute! Email us at **lxf.lugs@futurenet.co.uk** or if enough of you request a new LUG Forum on our website, we'll start one straight away.

Infopoint scheme

LUGs can get a free stand at computer fairs to evangelise Linux, thanks to **Jono Bacon** & Northern Computer Markets...

The problem with LInux advocacy is that it is often targeted at limited audiences. This is a problem the Infopoint project sets out to quash: the aim is to create a Linux information point that will be present at the many computer fairs dotted around the world. These computer fairs are typically visited by computer users who are looking to buy new equipment and software. An Infopoint will be another table at one of these fairs, but will provide info and demonstrations of Linux and Free software. This will create an opportunity for the public to see Linux in action and what it has to offer.

Northern Computer Markets (NCM) (**www.computermarkets.co.uk/**), one of the UK's most popular organisers of computer fairs, has offered to help kick off the Infopoint project. NCM runs computer fairs across Northern England including: Wolverhampton, Birmingham, Leeds, Bradford, Manchester, Hull, Doncaster, Stoke-on-Trent, Derby, Nottingham and Burton-on-Trent.

Northern Computer Markets is donating a free table once a month at each computer fair for the purpose of an Infopoint. This table can be used to demonstrate Linux to visitors and give out info and software. Computer fairs are visited by hundreds of people and an Infopoint will be a real opportunity to inform these visitors that there is an alternative in the form of Linux. This practical method of advocacy could have a real impact in spreading awareness of Linux and Free software.

Thanks to the generosity of Northern Computer Markets, I need YOUR help to organise the different Infopoints available. This is an ideal opportunity for LUGs or just a group of friends to get involved. Interested? See **www.jonobacon.org/infopoint/**. *Linux Format* will be featuring the dates of computer fairs where an Infopoint is present on these LUG pages, and we hope to feature much more about these Infopoints in future issues.

Infopoint forums for discussion are at: **www.jonobacon.org/forums/** As ever, please send comments, advocacy stories and suggestions to the address below and **spreadingtheword@jonobacon.org**.

Linux User Group organisers

If you're not listed here, or we have your details wrong, please contact us at: **LUGS!, *Linux Format*, 30 Monmouth Street, Bath, BA1 2BW**
or email your details to: **lxf.lugs@futurenet.co.uk**

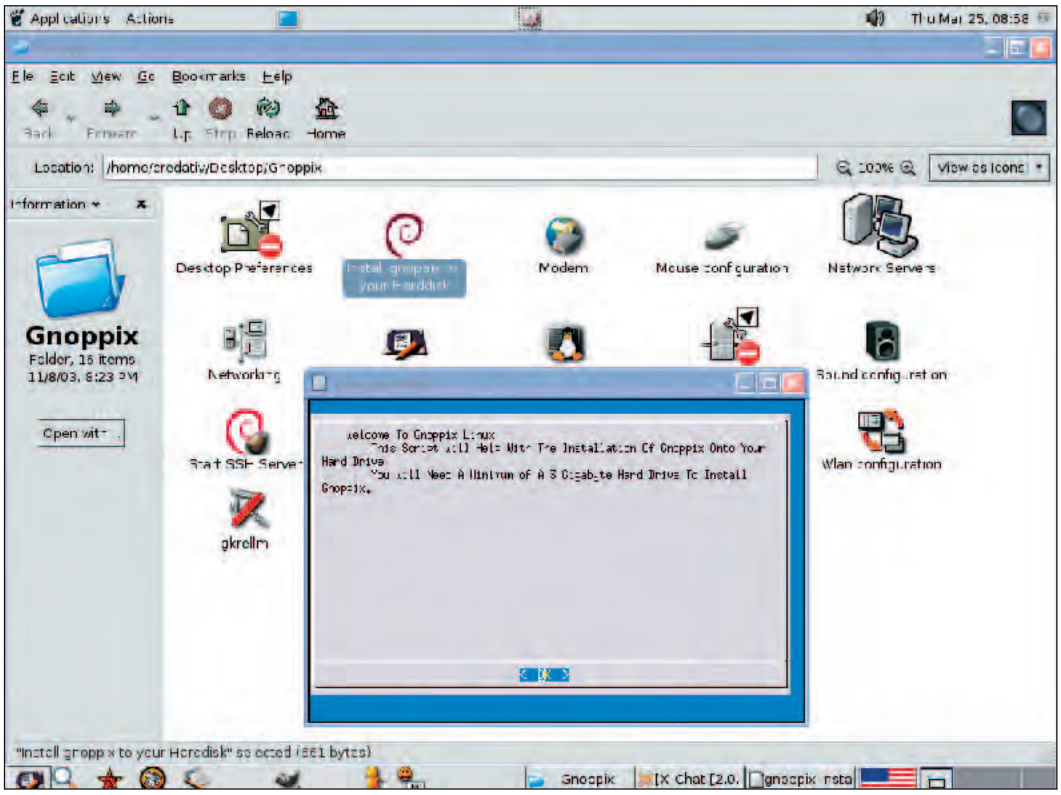
Coverdisc

Neil Bothwick is your guide through the wonders of this month's jam-packed *Linux Format* DVD. There is partitioning and boot info here that CD readers will find very useful too!

DISTROS

GNOPPIX

Gnoppix is another Debian-based Live CD distro. Despite what the name might suggest, this is not simply a modified Knoppix, with GNOME replacing KDE. While both distros are based on Debian 3.0 and have similar goals, Gnoppix was created separately from Knoppix – although the developers admit that it was inspired by it. Because it concentrates on GNOME, you will find it rather different to Knoppix in use. Which is better? That's a personal decision only you can make. The good news is that you have both distros sitting in front of you, ready for you to try them and come to your own conclusions.



```
    welcome To Gnoppix Linux
        This Script will Help With The Installation Of Gnoppix Onto Your
Hard Drive
        You will Need A Minimum of A 3 Gigabyte Hard Drive To Install
Gnoppix.
```

Looking every inch a GNOME system, this is Gnoppix beginning the hard drive installation process. Installation is not needed to run Gnoppix, but it is a good way to install a GNOME-oriented Debian system.

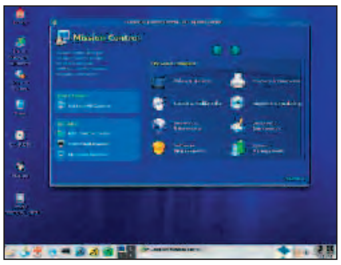
PCLinuxOS

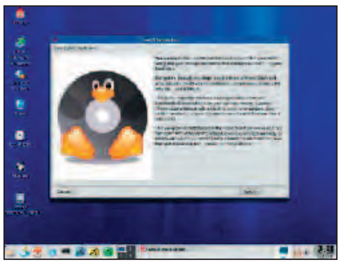
This is another Mandrake-based Live CD. This one is unofficial and uses many of the Texstar RPM packages that are often released for Mandrake. In contrast to many of the other Live CDs that are available, you need to login with PCLinuxOS. You can login either as **root** or **guest**, and the password for each is the same as the login name. Which you choose depends on what you want to do, and how much you trust yourself not to accidentally break things.

Unlike Mandrake's own Live CD offering, this one comes with a hard disk installer; although Mandrake's own installer is so good anyway, you may wish to consider using that if you want a hard disk Mandrake system.

Mandrake Move

This was on the CD but not the DVD a few months ago in *LXF51*. It was the first, and only, time we have included a major package on the CD but not the DVD. Since then, we have received numerous emails from readers of the DVD edition asking when they will have a chance to try it. The answer is *"right now!"* As with so many of these





These screenshots betray PCLinuxOS' Mandrake origins, but this is definitely *not* the same as Mandrake Move

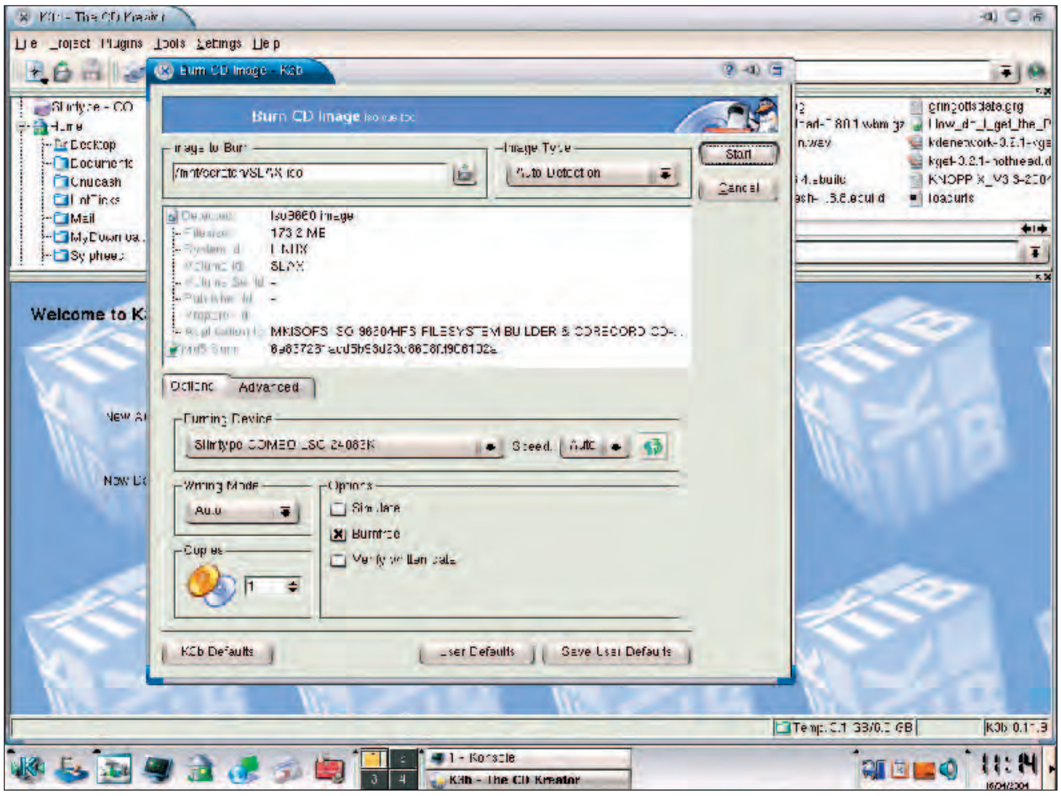
On the DVD

Wherever you see this logo it means there's related stuff on the DVD

IMPORTANT NOTICE

Before you even put the DVD in your drive, please make sure you read, understand and agree to the following:

The *Linux Format* DVD is thoroughly tested for all known viruses, and is independently certified virus-free before duplication. We recommend that you always run a reliable and up-to-date virus-checker on ANY new software. While every care is taken in the selection, testing and installation of DVD software, Future Publishing can accept no responsibility for disruption and/or loss to your data or your computer system which may occur while using this disc, the programs or the data on it. You are strongly advised to have up-to-date, verified backups of all important files. Please read individual licences for usage terms.



Burn CD Image - K3b

Image to Burn: /mnt/cdrom/SLAX.iso
Image Type: Auto Detection
Detected: Iso9660 Image
Filesize: 173.2 MB
System Id: LINUX
Volume Id: SLAX
Volume Set Id: -
Publisher Id: -
Preparer Id:
Application Id: MKISOFS ISO 9660/HFS FILESYSTEM BUILDER & CDRECORD CD-...
Md5 Sum: 8a83723?acd5b93d23c8638f49c6102e

Burning Device: Slimtype COMBO LSC 24082K   Speed: Auto
Writing Mode: Auto
Copies: 1
☐ Simulate
☒ Burnfree
☐ Verify written data

***K3b* – on several of this month's Live CDs – being used to burn a SLAX ISO image to disc.**

WE WANT YOUR CD & DVD REQUESTS!

WHAT WE WANT:

- **Requests for Free/Open Source Linux software that you'd like to see featured on the *LXF* coverdiscs.**
- **Constructive criticism of the software we've included on recent issues.**
- **Posts in the CD/DVD Requests section on our forums at www.linuxformat.co.uk**

WHAT WE DON'T WANT:

- **Technical questions – please direct those to support@futurenet.co.uk if you have defective discs, or alternatively lxf.answers@futurenet.co.uk if you have an application-related problem.**
- **Details of commercial apps or time-limited trial versions – *unless* you are a company wanting to offer *Linux Format* some sort of deal for including a version of your product on the LXF coverdiscs.**

WRITE TO US AT:
***Linux Format*, Disc Editor, Future Publishing, 30 Monmouth Street, Bath BA1 2BW** or email: **linuxformat@futurenet.co.uk with 'Coverdisc Request' as the subject-line.**

Live distros, the instructions are basically

1 Burn the ISO file to a CD
2 Boot and enjoy!

There is no hard disk installation option with this distribution. If you want that, installing Mandrake 10.0 from last month's DVD is your best bet. Mandrake Move is not intended as an evaluation version of Mandrake – it is a separate product intended to fill the specific needs of those who use Live CDs.

Writing ISO images to CD

Whether you have created an ISO image using the method detailed here, or you simply want to create a disc using one of the other Live CD distros that are only on the CD or DVD as .iso files, you will need to be able to write the image to a CD-R or CD-RW disc.

An ISO image contains the data as it would be written to the CD, it is basically a dump of the CD's contents to a file. Since the data is already formatted, burning it to disc requires a different approach to copying files from a filesystem. The standard command line tool for this is cdrecord and the method is covered in our regular *Essential Disc Information* section on page 111.

If you prefer to use a GUI tool, you have several choices. *K3b* is probably the most popular such program. Start the program and select Tools>CD> Burn CD Image... from the menu. Type the path to the ISO image file where it says 'Image to Burn' or use the file selector button – *K3b* should work out all the relevant settings for you; so, just click the Start button and let *K3b* get on with it.

If you are trying Linux for the first time and want to burn the disc under Windows, you will find instructions for several CD-writing programs at

»

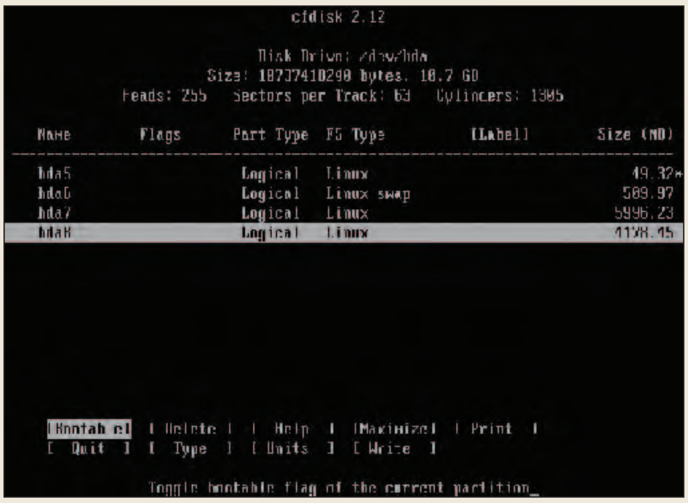
PARTITIONING YOUR HARD DISK

More than one way to root and swap

A hard disk is divided into partitions, each of which is accessed as a separate data storage area. A standard Windows setup has a single partition occupying the whole drive, so you may not have even noticed that is was partitioned before. But adding a second operating system will require that each has its own portion of the disk. Unlike Windows, Linux installations generally use two or more partitions, the exact number and layout depending on exactly what the machine is being used for.

Ask ten Linux users how you should partition your hard disk, and you will probably get ten different answers! The simplest setup has a swap partition and a root partition. The swap partition should be roughly twice the size of the computer's RAM, but probably not more than 512MB. It is used for temporary storage of data to free up memory. The root partition (/) contains everything else.

You can also have separate partitions for other sections of the filesystem hierarchy, /usr, /var/, /home and /usr/local are the most common candidates for this. However, this normally only complicates the situation. It is generally worth having /home on its own partition. This contains your own data and configuration files, making it a separate partition means you won't lose it if you reinstall. It also means you can share one /home partition between more than one distribution, if you like to experiment with the different distributions on the *LXF* coverdiscs.



```
                              cfdisk 2.12

                         Disk Drive: /dev/hda
                  Size: 10737418240 bytes, 10.7 GB
          Heads: 255   Sectors per Track: 63   Cylinders: 1305

    Name        Flags      Part Type  FS Type          [Label]        Size (MB)
 ------------------------------------------------------------------------------
    hda5                    Logical   Linux                             49.32*
    hda6                    Logical   Linux swap                       509.97
    hda7                    Logical   Linux                           5996.23
    hda8                    Logical   Linux                           4178.45

      [Bootable]  [ Delete ]  [  Help  ]  [Maximize]  [ Print  ]
      [  Quit  ]  [  Type  ]  [ Units  ]  [ Write  ]

                 Toggle bootable flag of the current partition_
```

« **www.linuxiso.org/viewdoc.php/howtoburn.html** For example, with Nero you use the File>Burn CD image menu item, select the ISO image file and follow the prompts. The defaults are usually correct.

Of course, as always with Linux, there is another alternative. You can boot from the Knoppix CD or DVD, which contains all the software you need to burn CDs. The only proviso is that you cannot do this if you boot from your CD writer, because it will be locked by Knoppix. But if your machine has two optical drives, you can boot from the reader and burn the discs on the writer.

Damn Small Linux has a boot option that should let you eject the disc after booting, freeing up the drive to use for burning. It didn't work quite as advertised, but you can use it to burn a CD with *cdrecord* without having Linux installed, even on a single drive machine. Here's how.

Burning CDs with DSL

Boot from the CD, typing

```
knoppix toram
```

at the boot prompt. When everything has booted, right-click on the desktop and select XShells>Root Access from the popup menu. If the ISO image you wish to burn is on your hard disk, mount the relevant partition now. For example, if it is on the first partition of your first drive (/dev/hda1), the normal location of a Windows partition, type

```
mount /mnt/hda1
```

Now you need to remove the link to the boot CD and unmount it with:

```
rm /cdrom
umount /dev/cdrom
```

SLAX makes this easier, by typing

```
slax copytoram eject
```

at the boot prompt, but you'll need 256MB of RAM to use this as SLAX is so much bigger than Damn Small Linux.

The next step is to find out the device address of your CD writer, with

```
cdrecord -scanbus dev=ATAPI
```

This should give something like

```
Linux sg driver version: 3.1.25
Cdrecord-Clone 2.01a19 (i686-pc-linux-gnu) Copyright (C) 1995-2003 J9rg Schilling
Using libscg version 'schily-0.7'
scsibus0:
                0,0,0       0) 'Slimtype' 'COMBO LSC-24082K' 'JKU5' Removable CD-ROM
                0,1,0       1) *
                0,2,0       2) *
                0,3,0       3) *
                0,4,0       4) *
                0,5,0       5) *
                0,6,0       6) *
                0,7,0       7) *
```

Now you can burn the ISO image to a disc with one of

```
cdrecord dev=ATAPI:0,0,0 /mnt/hda1/whatever.iso
cdrecord dev=ATAPI:0,0,0 blank=fast /mnt/hda1/whatever.iso
```

replacing **0,0,0** with the address you got from **cdrecord -scanbus**. The first command simply writes the ISO image to a blank disc. Use the second if you are writing to a CD-RW disc that already contains data, to erase it first. You can also add **-eject** to either of these commands to have the disc pop out when it is done. LXF

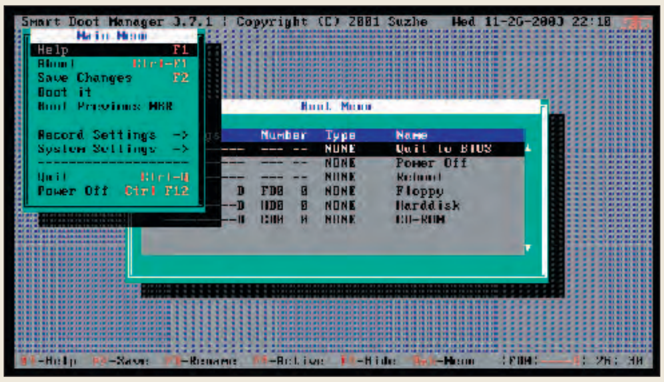
BOOTING FROM CD OR DVD

Smart Boot Manager sorts out your problems

Although many of our coverdiscs – including this month's – are made to be bootable, some people do find that they have problems booting from them. If this affects you, the first thing to check is that your computer is set to boot from CD (this includes DVDs) before the hard disk. Your computer's or motherboard's manual should give details on how to do this; it usually involves pressing a key (often **Del** or **F2**) during startup to access a menu. Watch the messages when you switch on the computer, it will prompt you which key to press.

Some BIOSes include a boot menu that allows you to select the boot device each time by holding down a key. This will sometimes work when a straight boot-from-CD fails, because it gives the system longer to read and recognise the disc.

If your computer is correctly set up, but still refuses to boot from these discs, it is most likely that there is an incompatibility between your BIOS and the *isolinux* system used by many CDs. The simple solution to this is to use a boot floppy like *Smart Boot Manager*. This is in the Essentials directory of the discs each month. Copy the image from sbootmgr.dsk to a floppy disk, using *dd* from Linux or *rawrite* from Windows. Booting from this disk gives a menu that should enable you to select and boot from the CD or DVD. In case your computer has no floppy drive, we have made a CD ISO image, called sbootmgr.iso. Burn this to a CD-R and boot from that. When you see the boot menu, eject the CD and replace it with the cover disc before selecting the relevant device (usually CD0).



Use *Smart Boot Manager* to persuade a reluctant computer to boot from your *Linux Format* CD or DVD.

DVD CONTENTS AT A GLANCE

Magazine

HotPicks	All the programs covered in this month's HotPicks section.
RoundUp	LiveCD distros covered in this month's Round Up
SDLProgramming	Source code for Trout Wars
Security	Programs covered in our feature on security
Storix	Try Storix for yourself
TheGimp	The GNU Image Manipulation Program.

Desktop

GNOME	The latest release of the GNOME Desktop
GNOMEInstallationGuide	A guide on how to install GNOME from scratch.

Distros

DamnSmallLinux	A 50 MB Live CD with a functional desktop.
Gnoppix	A Live CD Distribution using the GNOME Desktop.
Knoppix	A bootable CD with a collection of GNU/Linux software.
MandrakeMove	Move with Internet, Office & Multimedia in your pocket!
PCLinuxOS	Live CD based on Mandrake 9.2 that runs from a bootable CD
RPMLiveCD	Build rpm based Live Linux CDs
SLAX	MiniCD distro based on Slackware and Linux Live scripts

Essentials

Allegro	Multi-platform game library
ALSA	An alternative implementation of Linux sound support
Avifile	Library to read and write compressed AVI files
CheckInstall	A &amp;quot;make install&amp;quot; installations tracker
CSV	Comma separated index files of the cover discs
GLib	The GLib library of C routines
glibc	The C library used in the GNU system
GTK	A library for creating graphical user interfaces
gtkmm	A C++ interface for the popular GUI library GTK+
Guile	An embeddable library implementation of Scheme
Jigdo	Ease the distribution of very large files over the internet
Kernel	The latest kernel source and patches.
lesstif	LGPL'd re-implementation of Motif
libESMTP	A library for posting Electronic Mail
libmcrypt	A library to access various encryption algorithms
Libsigc	A callback framework for C++
libstdC++3	The GNU Standard C++ Library
libXML	A library for manipulating XML and HTML resources
Mesa	3-D graphics library which uses the OpenGL API
ncurses	Text-based interface creation library
OggVorbis	Open, professional audio encoding and streaming technology
RAWRITE	Write images to floppy disk with Windows
SDL	Portable low-level access for multimedia
SmartBootManager	OS-independent and full-featured boot manager
SVGAlib	Provides VGA and SVGA modes in a console

Essential disc info

Read this important information before you use your *Linux Format* coverdisc – CD or DVD. We've collated some helpful info to help you get the most from these jewels of data!

FINDING THE ESSENTIALS

MISSING SOMETHING?

As many of the programs on our discs are the very latest releases, they are often built on the very latest libraries and may depend on other packages your current Linux setup does not contain. We try to provide you with as many of these important supporting files and libraries as possible, though obviously we don't have space to include absolutely everything.

In many cases, the latest libraries and other packages you might need will be included in the "essentials" folder on the disc, so if you are missing dependencies, this is the first place to look.

PACKAGE FORMATS

Wherever possible, we try to include as many different types of package for an installation as possible, whether that be distribution specific RPMs, debs or whatever. Please bear in mind that we can only do this where space permits and when the packages are available.

We will, apart from exceptional or legally restricted situations, include the source files for any package, so that you can build it yourself.

DOCUMENTATION

These pages provide helpful information on how to install and use some of the packages on the CD. Please note that many of the applications come with their own documentation, and there are additional notes and files in the relevant directories.

WHAT ARE ALL THESE FILES?

If you are new to Linux, you may find the profusion of different files and extensions confusing. As we try to give as many packages as possible for compatibility, there will often be two or three files in a directory covering different types of Linux, different architectures and usually source and binary versions – so which do you install? They can be identified by their filenames, and usually just by the file extensions.

Someap-1.0.1.i386.rpm – This is probably a binary rpm, designed to run on x86 systems.

Someap-1.0.1.i386.deb – The same, but a debian package.

Someap-1.0.1.tar.gz – This is usually source code.

Someap-1.0.1.tgz – Same as the above, tgz is abbreviated form of tar.gz

Someap-1.0.1.tar.bz2 – Same, but uses bzip2 compression instead of zip

Someap-1.0.1.src.rpm – This is also source code, but supplied as an rpm to make it easier to install

Someap-1.0.1.i386.RH7.RPM – A binary, x86 RPM designed specifically for Red Hat Linux

Someap-1.0.1.ppc.Suse7.rpm – A binary RPM designed specifically for SuSE7.x PPC Linux.

Someap-devel-1.0.1.i386.rpm – A development version.

INSTALLING FROM TARBALLS

A tar ball is a two stage archive. First the files are archived into a single file with tar and then compressed with *Gzip* or *Bzip2*. To unpack, **cd** to the directory you want to unpack it, usually your home directory and type one of the following two lines:

```
tar xzvf /mnt/cdrom/Desktop/progname/progname-2.1.0.tgz
tar xvf --bzip2 /mnt/cdrom/Desktop/progname/progname-2.1.0.tar.bz2
```

Use the first for Gzipped files, those ending in .tar.gz or .tgz, and the second for Bzipped files, ending in .tar.bz2 or .tbz2. Naturally, you change the paths to suit the location and name of the archive. and replace **/mnt/cdrom** with whatever is applicable to your system (*eg* **/cdrom**). This normally unpacks the archive into a directory of the same name, enter that directory with:

```
cd progname-2.1.0
```

To compile and install the software, type the following three commands:

```
./configure
make
su -c "make install"
```

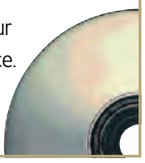
The last line will prompt you for the root password, as this stage must be run as root. If you are already logged in as root, just type **make install**. This will give you a default installation. If you want to change any aspect of the install, type **./configure —help** to see the options available. For example, you are usually able to change the default location with the PREFIX argument. When you have finished installing, you may remove the source files with:

```
cd ..
rm -fr progname-2.1.0
```

You should also log out as root, before you do anything you may later regret.

DEFECTIVE CDs

In the unlikely event of your disc being defective please email our support team (**support@futurenet.co.uk**) for further assistance. If you would prefer to talk to a member of our reader support team please call **01225 822 743**.



CREATING INSTALL CDS WITH CDRECORD

The quickest way to burn an ISO image to CD is with *cdrecord*. You need to be root to do this. First find the address of your CD-writer with

```
cdrecord -scanbus
```

This will show the devices connected to your system. The SCSI address of each device is the three numbers in the leftmost column, say 0,3,0. Now you can burn a CD with

```
cdrecord dev=0,3,0 -v
 /path/to/image.iso
```

You can simplify the command by saving some default settings in /etc/default/cdrecord. Add a line for each CD writer on your system (usually one) like this

```
Plextor= 0,3,0 12 16M
```

The first item is a label, after the SCSI address you put the speed and the buffer size to use. You can now replace the SCSI address in the command line with the label, but it gets even easier if you add

```
CDR_DEVICE=Plextor
```

Now you can burn an ISO image to disc with

```
cdrecord -v/path/to/image.iso
```

If you really don't want to use the command line, *gcombust* will do the job for you. Start it as root, select the "Burn" tab and the "ISO 9660 Image" gadget near the top of the window. Put the path to the image file in the gadget and press "Combust!". Now put on the kettle while the CD is created for you.

Other OS?

You don't have to use Linux to burn the ISO to a disc. All Linux-specific bits are already built into the image file. Programs like cdrecord simply dump it to the disk. If you don't have a CD-writer, find someone who has one, and a DVD drive, and use the CD burning software on their computer. It can be Windows, MacOS, AmigaOS whatever.

No CD burner?

What if you have no CD writer? Do you know someone else with one? You don't have to use Linux to burn the CDs, any operating system that can run a CD-writer will do the job (see above).

With some distributions it's possible to mount the images and do a network install, or even a local install from a disk partition. The methods often vary between distributions, so check vendors websites for more info. LXF

NEXT MONTH

ISSUE 55 ON SALE THURSDAY 10 JUNE

LEARN TO LOVE CODE

Scared of wrting programs? Don't be – everyone should be able to write simple scripts – make the computer work for you rather than the other way around! Plus, scripts can unlock the hidden power of your software. Join us next month and prepare to program!

Filesystems

Which is best? What's the difference? Major filesystsems, and some a little more obscure, explored, explained and compared next issue

On test

GNOME 2.6 , Xandros, Maya 6, the new VIA M2 mini-ITX board, games, the latest servers and more

PLUS

DON'T MISS NEXT MONTH'S *LINUX PRO!*

Dedicated supplement with pages of real-world Linux advice and case studies for IT professionals. Includes a look at *DB2* and *MySQL* databases, the PeopleSoft phenomenon, portable storage and more!

DON'T MISS YOUR COPY OF *LINUX FORMAT!*
Get it delivered to your door every month – subscribe on page 98

The exact contents of future issues are subject to change

EDITORIAL
Editor Nick Veitch *nick.veitch@futurenet.co.uk*
Reviews Editor Paul Hudson *paul.hudson@futurenet.co.uk*
Art Editor Julian Jefferson *julian.jefferson@futurenet.co.uk*
Production Editor Matt Nailon *matthew.nailon@futurenet.co.uk*

Editorial Contributors Mike Saunders, Chris Brown, Hoyt Duff, Jono Bacon, David Cartwright, David Coulson, Andy Channelle, Michael J Hammel, Hans Huberland, Pamela Jones, Dave Cross, Marco Fioretti

COVER CD PRODUCTION
CD Editor Neil Bothwick

ART CONTRIBUTORS
Art assistance Colin Nightingale
Photography Amanda Thomas, James Wilson, Alamy, Digital Vision, Eyewire, Photodisc
Illustration Paul Bateman, Chris Winn, Shane Collinge

ADVERTISING SALES
Senior Sales Executive
Andrew Hudson *andrew.hudson@futurenet.co.uk*
(01225) 442244 Ext 2441

MARKETING AND PROMOTIONS
Marketing Manager Fiona Tully
Subscriptions Executive Marie Lock

PRODUCTION
Production Co-ordinator Diane Ross
Group Production Manager Clare Tovey
Commercial Buyer Mike Thorne
Advertising Production Co-ordinator Jo Crosby

MANAGEMENT
Publisher Kelley Corten
Publishing Director John Weir
Managing Director Robert Price

DISTRIBUTION AND CIRCULATION
Trade MarketingManager Jason Comber
jason.comber@futurenet.co.uk

Distributed by Marketforce (UK) Ltd, 5th Floor, Low Rise Building, Kings Reach Tower, Stamford Street, London, SE1 9LS **Tel** 020 7633 3300
Overseas Distribution by Future Publishing Ltd.
Tel 01225 442244.

Overseas Licences
International Licensing Director Simon Wear
simon.wear@futurenet.co.uk **Tel** +44(0)1225 822798
Fax +44 (0)1225 788105 **Cell phone** +44(0)777 5641493

Contact Details
Linux Format, 30 Monmouth Street, Bath BA1 2BW
Tel +44 (0)1225 442244 **Email** *linuxformat@futurenet.co.uk*

Subscriptions and Mail Order
Phone +44 (0)1458 271178. See page 98
Email *linuxformat.subs@futurenet.co.uk*

All CD-ROM demos and reader submissions are supplied to us on the assumption they can be incorporated into a future covermounted CD-ROM, unless expressly stated to the contrary.

Disclaimer All tips in this magazine are used at your own risk. We accept no liability for any loss of data or damage to your computer, peripherals or software through the use of any tips or advice.

Printed in the UK by Midway Clark (Holt) and Southern Print
© Future Publishing Ltd 2004

Future Publishing Ltd. is part of The Future Network plc.
The Future Network produces carefully targeted specialist magazines for people who share a passion.
We aim to satisfy that passion by creating titles offering value for money, reliable information, smart buying advice and which are a pleasure to read. Today we publish more than 90 magazines in the UK, US, France and Italy. Over 80 international editions of our magazines are also published in 28 other countries across the world.
The Future Network plc is a public company quoted on the London Stock Exchange (symbol: FNET).
Non-executive Chairman Roger Parry
Chief Executive Greg Ingham
Group Finance Director John Bowman

Tel +44 (0)1225 442244

BATH • LONDON • MILAN •
NEW YORK • PARIS •
SAN FRANCISCO • WROCLAW
www.thefuturenetwork.plc.uk

future publishing
Media with passion

ONLY AVAILABLE WITH LINUX FORMAT MAGAZINE

LINUX PRO

FROM THE MAKERS OF LINUX FORMAT

JUNE 2004

BETTER THE DEVIL YOU KNOW?

Free software for BIG BUSINESS

Is there really common ground for commerce and Open Source? Our expert analysis and industry comment uncovers the realities

PLUS

Open Source Java debate
Should Sun release Java? Will it ever? The experts speak

Meet the regulations
Your corporate security policy should address your legal obligations...

The Zen of Zend
PHP development with professional tools – give your business a performance edge

Managing the grid
Oracle's grid-based solutions address management issues

PRACTICAL LINUX SOLUTIONS FOR I.T. PROFESSIONALS

Welcome

TWENTY PAGES OF REAL-WORLD LINUX FOR IT PROFESSIONALS

One of the continual threads of discussion in the world of Linux is Open Source and its suitability for particular tasks. To many it seems, the value or quality of the software produced is secondary to the way it is produced. What is most surprising though, is the amount of sheer nonsense and erroneous attributes that are applied to Open Source.

Most infamously, there was Steve Ballmer's "cancer" comment, which (incorrectly) stated that by using the GPL, a company would never again be able to sell software. Recently, we have seen a similarly inept comment from Dan O'Dowd of Green Hills Software, which accuses Linux of being a security risk. The basis of his argument is that everybody can contribute code to Open Source, which would include terrorists and the ominous "foreign-nationals" that he seems so concerned about. Apparently, *"Linux in the defense environment is the classic Trojan horse scenario – a gift of 'free' software is being brought inside our critical defenses. If we proceed with plans to allow Linux to run these defense systems without demanding proof that it contains no subversive or dangerous code waiting to emerge after we bring it inside, then we invite the fate of Troy..."*

This rather pre-supposes two things – firstly, that all contributed code is blindly accepted. Secondly, that nobody else ever looks at any of the code. Er, both of which are more true of closed-source development. If the alternative is a closed-source system that turns out to have so many easily exploitable holes because it hasn't been looked at by enough people, that seems to be OK. Maybe commercial software houses should vet their programmers?

Nick Veitch Editor
nick.veitch@futurenet.co.uk

THE CONTENTION SEEMS TO BE THAT ANYONE CAN CONTRIBUTE TO OPEN SOURCE CODE, INCLUDING TERRORISTS AND OMINOUS-SOUNDING "FOREIGN NATIONALS"

CONTENTS

10

18

CONTACTS

EDITORIAL

Editor
Nick Veitch *nick.veitch@futurenet.co.uk*
Reviews Editor
Paul Hudson *paul.hudson@futurenet.co.uk*
Art Editor
Julian Jefferson *julian.jefferson@futurenet.co.uk*
Production Editor
Matt Nailon *matthew.nailon@futurenet.co.uk*

Publisher
Kelley Corten

Photography
James Wilson, Powerstock, Digital Vision Getty, Corbis

ADVERTISING

Senior Sales Executive
Andrew Hudson
andrew.hudson@futurenet.co.uk
(01225) 442244 Ext 2441

LinuxPro is a part of LINUX FORMAT magazine

Published by future publishing
Media with passion

30 Monmouth Street, Bath BA1 2BW
Tel +44(0) 1225 442244

THE ZEN OF ZEND

DOES YOUR BUSINESS USE THE *REAL* POWER BEHIND PHP?

PAUL HUDSON investigates Zend's PHP-based product range and how it can help you get more value from the Web...

At the core of PHP 4 and now PHP 5 lies the *Zend Engine*. As the founders of Zend and also lead developers on the PHP language itself, both Zeev and Andi have many years of combined experience working with the system in a variety of environments, and so it's no surprise to see that Zend's product list is a veritable toolkit for PHP users who want the most out the language. The company being named by the combination of "Zeev" and "Andi" is a blunt testament to how integral Zend is to the life of PHP.

Apart from the *Zend Engine*, Zend's Linux product-line is primarily made up of three distinct parts: *Zend Studio*, for developing PHP; *Zend Performance Suite*, for caching and optimising PHP code; and *Zend Safeguard Suite*, for encrypting and licensing PHP scripts externally. All of them are absolutely standalone – you don't need to use any special combination unless you specifically want to.

DEVELOP

Irrespective of how fast your servers are or how much time your admins spend tuning configuration scripts, the basic performance of any website is nearly always defined by the quality of the code behind it. Although many developers still think that *Kate* is a suitably powerful text editor for them, *Zend Studio* continues to be the IDE for professional PHP developers. As it's written in Java, you can use it as smoothly on Linux as you can on Mac or Windows, which is great for developers who 'hot-desk'.

"SUPPORTING LINUX IS A STRATEGIC DECISION FOR ZEND, AND I'M HAPPY TO SAY WE DO THAT NOT ONLY WITH OUR SERVER PRODUCTS, BUT ALSO WITH THE DEVELOPMENT TOOLS."

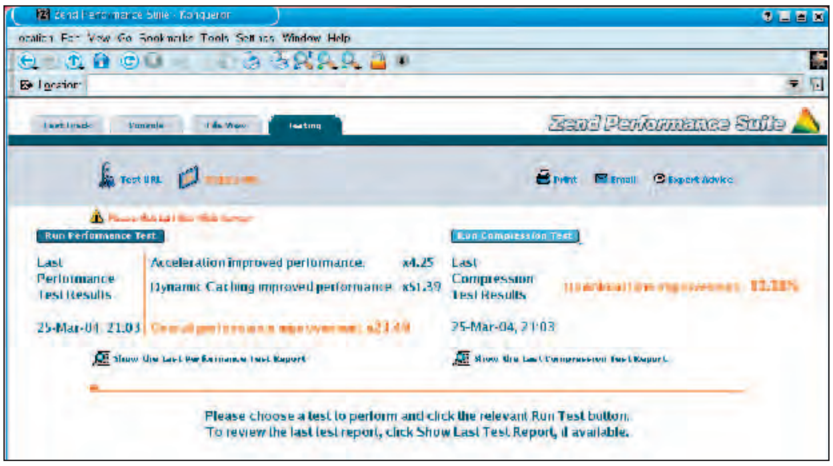
ZEEV SURASKI, CTO ZEND

However, comparing *Zend Studio* and *Kate* – and believe us, we really do like *Kate* a lot and respect all the work the developers have done there – is like comparing chalk and cheese. *Zend Studio* is integrated very deeply with PHP – you can interactively debug your scripts, read the contents of output buffers, run profile operations to see how well the script performs, and it also contains a powerful program linter (semantic checker) that statically scans your source code and points out possible improvements. *Zend Studio* also features powerful code completion as you type – function names are filled in, parameters are automatically listed, and snippets of the PHP manual are shown as you type to help you maximise your time spent programming.

Despite the exemplary nature of the other features, it's the debugger and profiler that steal the show. If you've previously programmed in other languages such as Java or C++ and used a fairly modern IDE, you'll be used to the idea of step-through debugging – marking lines of code as breakpoints where execution should pause, and the IDE will automatically stop when appropriate and let you inspect all the variables to see what their current values are. Compare to this to the manner of debugging most PHP programmers use, which is to print out messages such as **In function** and **somefunc() called**, and you'll find a huge increase in productivity.

The other show-stealer is the profiling mechanism, which runs back the entire script and monitors how much time was spent in each function call and each script, then breaks it down into pie charts and statistics. Without this information, optimising a script is a matter of guesswork – the adage is that 90 per cent of a program's time is spent in 10 per cent of the code: if you choose to spend your time optimising code that has little impact on performance you might as well be doing nothing.

Unless you have unusual requirements, the chances are you'll be able to work entirely in *Zend Studio* – it has a full CVS system to allow you to version-control your files, and support for FTP is seamless in that you can save and load remote files just as if they were local. One big advantage is that *Zend Studio 3* was the first IDE released to have full support for the new syntax in PHP 5.


Zend Performance Suite - Konqueror
ocation Edit View Go Bookmarks Tools Settings Window Help
Location:
Testing
Zend Performance Suite
Test URL
Print
Email
Expert Advice
Run Performance Test
Run Compression Test
Show the Last Performance Test Report
Show the Last Compression Test Report

Last Performance Test Results
Acceleration improved performance: x4.25
Dynamic Caching improved performance x51.39
25-Mar-04, 21:03
Last Compression Test Results
25-Mar-04, 21:03

Please choose a test to perform and click the relevant Run Test button.
To review the last test report, click Show Last Test Report, if available.

***Zend Performance Suite's* online management GUI is a quick and easy way to see how much return you're getting on the investment.**

SCALE

A fundamental aspect of system administration that we try to emphasise in this magazine is that more companies need to be flexible enough to scale up without having to drastically rethink their entire IT strategy. In web serving terms, too many companies still think the solution to the scaling issues is just to throw more hardware at the problem: *"Want to serve twice as many web pages? Buy twice as many servers!"* To a point, that strategy works, particularly if you buy a strong load-balancing solution. But, a much easier option is to use *Zend Performance Suite* on your web servers, as it works to dynamically boost the speed they can serve up web pages.

The technique behind *Zend Performance Suite* (*ZPS*) is simple, but clever: the first time a new script is requested, it is parsed by PHP and, before it is actually executed, it gets cached in RAM by *ZPS* so that subsequent accesses can skip the whole parsing process. Depending on the size of the scripts being used, this can result is a massive speed up – usually between two to two-and-a-half times.

One particularly clever feature yet to be seen in other caching solutions is smart content compression: compressing HTML output saves bandwidth by sending less data than if it were uncompressed. However, one problem that compression suffers from is that often it's not worth the effort: if a page compresses by just 10 per cent, then extra time that the CPU spent compressing it was worthless. *ZPS* works around this by remembering how well each script compresses, and, if the ratio was too low, skips the process next time – something that is simply not possible with other solutions.

If you want to a get a free performance boost, you could try out the *Zend Optimizer*, which isn't a code caching solution like *Zend Performance Suite*, but it can still make a fairly sizeable speed difference – around 10 per cent depending on what you're doing in the script. The *Optimizer* works by re-arranging parts of the code to work faster, and therefore it's quite an advanced product. As a result, it had some analysts likening it to brain surgery, and, if you had a choice, who would rather have perform brain surgery on your scripts?

PROTECT

One of the inherent problems with scripting languages is that they are distributed in source format. Although this is fine if the source code itself is licensed under an Open Source licence such as the BSD or GPL licences, but if your company has just spent hundreds of man-years developing a new web-based project, the chances are you'll want some return. How, then, to distribute your scripts without letting the world read your source code?

This is where *Zend Safeguard Suite* comes in. By converting your source code to a special, encrypted format, it becomes essentially illegible to people, but can still be executed on servers. Unlike some other "encoders", *Safeguard Suite* isn't merely an obfuscator that mashes up variable and function names. Instead, the source file becomes literally meaningless as it is 100 per cent scrambled. As a result, the script can no longer be read by a vanilla install of PHP, as the script needs to be "decoded" to get the original meaning - and decoded is there in quotes because it doesn't actually have to be *decoded* as such, and it doesn't slow things down at all. In order to process the encrypted file, the decrypter needs to be installed, and Zend cleverly chose to bundle their decrypter inside the Zend Optimizer, which means that most systems have it installed already, and those that don't can install it easily.

SafeGuard Suite does more than just scramble scripts, however - it's a full licensing solution at the same time. You can, for example, set expiry dates on your scripts so that they stop working after a trial period has expired. You can also limit what machines can run the script, whether by IP address, hostname, or other uniquely identifiable information. The latest feature to be added was that you can limit the number of concurrent users, which allows you to package your scripts so that they can be used by a maximum of 5 people or 50 people at a time - essentially letting you provide different tiers of functionality at different prices, all through the same script.

WIN

Each of the Zend products is able to standalone or can easily plug together to provide a combined and increased return on investment. Thousands of companies already get substantial value out of the Zend offerings – a list of Zend users reads like a Who's Who: AMD uses *Zend Performance Suite*, HP uses *Zend Safeguard Suite*, and Nortel uses *Zend Studio*.

Both *ZPS* and *Zend Studio* are about working smarter: buying new hardware is much more expensive than buying *ZPS*, and equally buying your existing developers *Zend Studio* is much cheaper than throwing more manpower at a resource. Both times you save money as well as getting higher efficiency, which shows that Zend really does live up to the high expectations placed on it. ■■■

SMALL BUSINESSES WELCOME

ONE OF THE SMARTEST moves that Zend made was to offer a special Small Business program, which offers *Zend Studio, Zend Encoder* (a cut-down version of *Safeguard Suite*) and *Zend Accelerator* (cut-down version of *Zend Performance Suite*) available for small businesses for a one-off fee of just $295 per licence. Zend's definition of a small business is quite broad: so long as your company makes less than $250,000 a year in revenues, you quality for the program.

As both *Zend Encoder* and *Zend Accelerator* are cut-down versions of standard products, they do offer a little less functionality; but the core is still there: *Encoder* does all the source code protection, but has none of the extra licensing functionality. *Zend Accelerator* caches PHP code, but doesn't have some of the more advanced functionality such as smart compression.

At $295 per licence this is an absolute bargain, and should be well within the grasp of every small business.

LET MY LANGUAGE GO!

PAUL HUDSON looks at the reasons for and against an Open Source Java...

As any Java programmer can tell you, "write once, run anywhere" is the long-standing mantra for the Java programming language. Sure, at first it was "write once, crash anywhere," as cross-platform compatibility issues plagued the early versions, but Sun Microsystems has worked hard to remove these niggles and Java 1.5 shows just how far the platform has advanced since it was launched.

But despite being promoted as 'ubiquitous,' Java adoption continues to be stunted far from its potential – particularly under Linux – and the primary reason is often said to be that Sun has kept too much control over Java for it to be used freely. Sun's Java implementation is the 'standard' – it implements all the features as defined in the documentation, and remains on the cutting-edge as new features are added.

However, there is no Open Source equivalent of this: the closest is the *GNU Compiler for Java (GCJ)*, which lags some way behind the official implementation. As a result, developers who wish to write truly free Java software must either work full-time with *GCJ* and live without the advanced features of the official implementation, or code for the 1.5 spec and try to backport their code to *GCJ* later. Both are difficult options that developers would rather avoid altogether.

Scott McNealy, Sun Microsystems CEO, on open-sourcing Java: *"We're trying to understand what problem does it solve that is not already solved..."*

If Java was open – truly open so that anyone could download it, change it, compile it, and redistribute it – the situation would be entirely different. But would it also have negative repercussions for Java in the enterprise?

"DO IT", SAYS IBM

Apart from Sun itself, few companies prefer Java to be closed-source. IBM has been the most vocal, having offered to help Sun open up Java. Of course, this offer may not be quite what it seems – not only have the two traded blows about how Linux on the desktop should be done, but IBM also has the largest market share in the Java application server arena, which gives it a vested interest in having a degree of control over the development of Java. Control that Sun, no doubt, would rather *not* give up.

Of course, IBM by no means stands alone in the debate. Eric Raymond argues that Sun stands to gain from allowing Java to be distributed freely with Linux. As it stands, few distros bundle Java, largely as a result of the lack of openness. Distros such as Debian will never be able to include Java in the system as they strive to be 100 per cent free. On the other hand, SUSE bundles Java with its default 9.1 Pro system, which shows that it *is* possible, if some compromises are made.

If Java was open, it would likely become as ubiquitous as *gcc* – every distro would be able to run Java apps as standard, and as a result the number of apps available for the platform would rise exponentially. If anyone has any doubt how much benefit Open Source can bring to Java developers, they need only look at *Ant*, the Java build tool. This simple tool, licensed under the *Apache* licence, reads an XML file that describes how to build a project. Despite its apparent simplicity, it is now the standard build tool for Java developers, with most if not all IDEs supporting Ant compilation by default. Even Microsoft has taken note of the success of *Ant*, and has produced *MSBuild* – a tool so similar to *Ant* that would you be forgiven if you couldn't tell the difference.

At the same time that *Ant* makes huge strides forward for developer productivity, Sun continues to reject offers to join the Eclipse project – the Open Source Java IDE backed by Fujitsu, HP, IBM, Intel, MontaVista, Novell, Oracle, Red Hat, SAP, SUSE, Sybase, and other industry leaders. Could the contrast be any bigger?

"WE WON'T DO IT", SAYS SUN

Sun's defence focuses largely around the threat of diversification – the idea that if Java was opened up entirely, Microsoft would be able to create its own version of it that tied developers to Windows. This happened previously with the creation of the Windows Foundation Classes for Java, and Sun sued Microsoft successfully because it had created an incompatible version of Java. If Java was open-sourced, a licence such as the GPL would give a company like Microsoft every right to create an incompatible release of Java that tied developers to its platform. This argument is valid, and Sun is right to be blaming Microsoft as the source of its unrest. But, many have said that the problem is less about legal issues and more about technology issues – specifically that Microsoft's .NET platform, though less mature, is catching up because it is faster-moving, cheaper, and, in its *Mono* implementation, much more free as well.

WOULD MAKING JAVA AVAILABLE UNDER THE GPL ALLOW A COMPANY LIKE MICROSOFT THE CHANCE TO TAKE OVER AND LOCK IN DEVELOPERS?

One thing that *is* clear is that not everyone connected with Sun believes that Java should remain closed. Even James Gosling, the inventor of the Java programming language, said, *"I am certainly one of the people who would love to make [Java] Open Source."* Furthermore, at least some parts of Sun have a firm commitment to Open Source software, as the *OpenOffice.org* project shows. Of course, more cynical people might put Sun's release of *OpenOffice.org* down to a wish for Sun to remove Microsoft's *Office* cash cow, which would put its "support" in quite a different light.

That said, the question remains: would GPLing Java allow Microsoft the chance to take it over and lock-in developers? The likelihood of this happening is slim, for two reasons. First, Microsoft has now been declared a monopoly, which would make it hesitant to be seen to repeat its illegal actions of the past. Second, and more importantly, a GPL licence would mean that any changes Microsoft *did* make to Java would also need to be released under the GPL – a licence Microsoft has branded as "viral", and has been avoiding at all costs. While it's unlikely that MS would dare go near GPL software, it would also mean that any additions it made could be taken by the community and implemented elsewhere, effectively 'untying' developers.

If Java was to fragment, it would be more likely to happen as people created their own, incompatible releases – if IBM Java was different to Sun Java, which itself was different to Red Hat Java, and so on. Thankfully this is again unlikely – the Java specification might as well have the first eight letters in bold, as it's the most *specific* specification you're likely to find anywhere. As long as the majority of vendors stuck closely to the spec – which would only be in their interests – Java would continue its cross-platform strength.

THE DUAL-LICENCE OPTION

Perhaps the solution to the problem is for Sun to dual-license Java, in the same way as *OpenOffice.org* and *MySQL*. In this situation, it would become available under the GPL (or similar licence) for people who want to use it freely, but Sun would also licence it under a proprietary, non-free licence that allowed licensees to make changes without the need to distribute them to the community.

At first this might seem silly: who would have the non-free version when the free version is readily available? You'd be surprised: many large companies would happily pay for the privilege to have unrestricted development, as it would give their products a commercial advantage over everyone else. And, with the free version helping make Java as ubiquitous as we've always been told it is, the non-GPLed version would be an attractive product for Sun to sell and support.

Why, then, does Sun want to cling on to Java? A possible reason is that Sun envisages an era of Java as a brand rather than Java as a technology. The first instance of this is already available: the Java Desktop System uses Linux, X Windows, GNOME, and *OpenOffice.org* – all written using C or C++ rather than Java. Despite the bundling of a few Java apps, the 'Java' name here is just marketing – it's arguable that more people recognise the Java brand than the Sun brand.

If Java was free, this stamping might no longer be possible. When you think of Mickey Mouse, you think of Disney, and when you think of Java you think of Sun. Would that continue in an Open Java world? We think so. In fact, we think that an Open Java would strengthen the platform immensely, allowing the language to piggyback on the success of Linux. Java hasn't been bundled with Windows for years, but with Linux, Sun yet again has the chance to have Java on every desktop... ■■■

WHAT THEY SAY...

The two sides weigh in with their points of view

"We need an absolutely official Open Source implementation of Java." Bob Sutor, IBM's Director of WebSphere Infrastructure Software

"Go Open Source with DB2 and then you can tell me what to do with my assets!" Scott McNealy, Chairman, President and CEO of Sun Microsystems, responding to IBM

"Sun's insistence on continuing tight control of the Java code has damaged Sun's long-term interests by throttling acceptance of the language in the Open Source community, ceding the field (and probably the future) to scripting-language competitors like Python and Perl" Eric S Raymond

"IBM is a strong supporter of the Open Source community, and we believe that a first-class Open Source Java implementation would further enhance Java's position in the industry by spurring growth of new applications and encouraging new innovation in the Java platform." Rod Smith, VP of Emerging Technologies, IBM

"Sun has contributed more to Open Source than anybody else bar Berkeley. We understand Open Source better than anyone else. IBM is just wrapping itself in the flag, but it still behaves like an old-fashioned systems company – Sun is actually taking the risks." Simon Phipps, Chief Technology Evangelist, Sun.

WHY LINUXWORLD 2004 IS THE MOST IMPORTANT UK LINUX EVENT

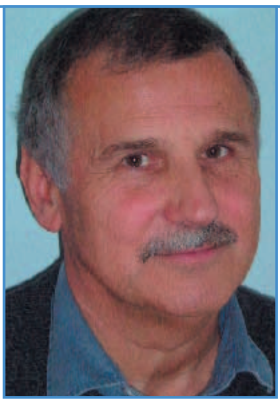
LET'S GET TOGETHER

Meeting with your peers and exchanging information is a vital way to keep up-to-speed in the Linux World. **PAUL HUDSON** spoke to **BOB DENTON**, Event Director of LinuxWorld UK.

As the UK's longest-running Linux event and also the highlight of our yearly calendar, the Linux Expo in London is finally getting a facelift. Now branded under the global title of LinuxWorld, the conference this year is going to be twice as big as last year's successful event. As the CEO of the company organising the conference, Bob Denton knows what it takes to bring enterprise Linux and community Linux together – we spoke to him about the importance of networking with your peers, who should attend, and why it's not all just about techies...

***LINUX PRO:* What's so important about people meeting up at an exhibition/conference like this one?**

BOB DENTON: The main reasons for attending Linux World are to get up-to-date with what's happening in the market; to go hands-on with new systems, services and solutions; to check out compatibility; to try before you buy and to get good advice directly from the manufacturers, resellers and Linux gurus – and perhaps just as importantly to mix with like-minded people and learn from their experiences. The combination of the exhibition, free keynotes and seminars, the free daily *Great Linux Debate* and the master classes ensure that visitors can choose between a quick once-over at a subject, or going for full immersion.



"ALREADY FOR 2004, WE HAVE THE MAJOR PLAYERS AS PLATINUM SPONSORS – NOVELL, HP AND SUN. EXPECT MORE OF EVERYTHING THIS YEAR!"

BOB DENTON, LINUXWORLD UK

***LXP:* How does LinuxWorld Expo relate to last year's Linux Expo?**

BD: Linux Expo has been the leading UK Linux event for the last five years, and the show manager is still Gary Cunningham, which ensures continuity. But this year, we got together with IDG World Expo to become under licence a part of the LinuxWorld global series of events. This allows the major global exhibitors to negotiate and plan for a global presence across events in the US, UK, Germany, Italy, China, Japan and a growing number of other locations.

What it means to the visitors and delegates is that we have – on tap – a direct connection to the key seminar and masterclass presenters from around the world, with good feedback on their performances so that we can create a stimulating and valuable series of conference streams. Plus the global approach will progressively enable us to attract even more international exhibitors.

For 2004 we already have the major players (Novell, HP and Sun) as Platinum Sponsors, and they will bring top speakers and present their latest solutions and thinking to enable visitors. Expect more of everything this year – more exhibits, more seminars, bigger debates, new master classes, and lots more still to be announced.

With *Linux Format* and *Linux Pro* as our premier media sponsors, you will be kept up-to-date as exhibitors, speakers, features and other news breaks!

***LXP:* How important is the .ORG village?**

BD: This will be the biggest .ORG village ever seen in any UK Linux event. The importance of the .ORG community is

DATE FOR YOUR DIARY

LINUXWORLD EXPO 2004 UK is to be held on **Wednesday 6** and **Thursday 7 October 2004** at Olympia Two in West London. As per usual, *Linux Format* and *Linux Pro* will be there, with Nick's latest hairstyle usually being as controversial as what he has to say in the *Great Linux Debate*, and Paul rounding up freebies by the bagful!

immense – all experts in their own right, they attract a huge audience to the event and pride themselves on providing solutions to most of the questions visitors will have about Linux.

***LXP:* Would Linux User Groups (LUGs) benefit from attending too?**

BD: Yes very much so, and we are preparing to email all the LUGs we have identified in the UK to invite them to participate in a number of ways. There will be a LUG meeting point – where the various groups can present their contact details, meeting dates and other announcements; plus on the Thursday afternoon we are planning a LUGfest in our large meeting room for them to get together to talk about Best Practice and how they can jointly forward the interests of the Linux community.

We are currently in the process of meeting and contacting LUGs to work with them to define how best they can use these facilities. Any *LXF/LXP* readers who are LUG members are welcome to get in touch with us to let us know what subject matter they would like to see included in the LUGfest session – it's planned as a 90-minute session on Thursday afternoon, and we want to cover a variety of subjects.

***LXP:* What kind of company would benefit from exhibiting at LinuxWorld UK?**

BD: This year, the show spreads across two floors at Olympia Two, with more space for the conference sessions and some new features – like our Email Corner where visitors can pick up their messages (adjacent to the catering area). Exhibitors can opt to take just the space and build their own stands, or we offer 'turnkey' stands – where exhibitors turn up with the stand already built and ready for business: furniture in place, space for graphics and promo material, and an electrical package all ready for plugging equipment into.

A host of sponsorship opportunities exist from advertising in the *Linux Format*-produced show guide to branding carrier bags and badges. A number of exhibitors (AMD, EsEda, LDRA, and Sybase, for example) have already booked sessions on our two Product Education Theatres where they can present their products, services and solutions in 45-minute sessions – free to delegates.

Space is disappearing fast and we have already have reserved stands for AMD, Cyclades, Equinox, EsEDA, HP, Linux Format, Novell, Optism, Sun, Sybase, Thawte and UKLinux.net, amongst others. Watch this space through the summer via our website which is when most of our exhibitors budgets come free!

To cut a long story short, anyone who has a product, development, application, solution, service, accessory, training course, book or an idea that relates to Linux in particular or Open Source in general should be exhibiting at LinuxWorld – Gary Cunningham is available on **0845 408 0555**.

***LXP:* Is the Expo just for techies, or would managers benefit from visiting also?**

BD: No – last year was not just for techies either. 25 per cent of last year's attendees were directors of their organisation, 86 per cent had an active role in the purchase of IT products and services for their organisation, and 72 per cent stated that they play an active role in the strategic process of applying new technologies within their organisations.

Linux is coming of age, and while the notion of open source has clearly come from a technical route, we are now seeing massive strides in its implementation as a tool for those who don't understand the technicalities. This year's show will be a must-see for corporates who want to evaluate or implement Linux for their systems, in particular the financial sector and public sectors appear to be the fastest-moving areas and in education the LEAs are starting to become very serious about adopting Linux. The move towards the Linux Desktop will be one of the highlights of the show. We expect the attendees this year will include IT directors and managers, and an increasing proportion of executives and managers who merely want to see what Linux can deliver and have limited interest in how it does it!

***LXP:* What speakers can we expect to attend?**

BD: It's a little early to give our final line-up of speakers, but the topics covered in the LinuxWorld masterclasses are, Linux in Education, Security and Linux, Linux in the Public Sector, Open Source development, Desktop Linux, Making a move from Windows to Linux; plus we have some top case studies being presented by large organisations that have got the very most out of adopting Linux.

We have some very senior individuals coming to present the HP, Novell and Sun keynotes – and the Great Linux Debate will have a broad and stimulating line-up of panel members to answer the visitors' questions.

***LXP:* Do you think we're going to see an increase in the number of Linux events in the UK?**

BD: By aligning ourselves with the global family of LinuxWorld events, we expect that our show will continue to dominate the UK with an ever-growing line-up of exhibitors and presenters. We would expect that the exhibition may grow into having several zones that will reflect either the technical aspects of Linux and OpenSource or the various solutions that it can deliver. The conference will become broader with technical and solutions streams – wherever your interest lies, you can find it at LinuxWorld.

76 per cent of the 6,000 visitors to LinuxExpo UK 2003 said that they will visit us again this year, and with an ever-broadening series of media partnerships we expect to have a show and conference that is just plain unmissable – see you there! Prospective visitors can register now for a free ticket on **www.linuxworldexpo.co.uk** ■■■

ABOUT BOB DENTON

BOB DENTON'S BEEN involved in many aspects of IT from the early 1970s, when he was a key player in cash registers going electronic. By the late 1970s, he had been Sales Director at Dixons distribution company Ace – where he launched the Mattel Intellivision games console – and earlier at Texas Instruments he had been their Personal Computer Manager defining the likely applications for home and business computers. In the early 1980s, he was involved in the launch of the Dragon 32 and other early home computers; and Bob was also the developer of Micronet 800, and the distributor in UK, France and China for the Sinclair ZX81, ZX Spectrum and QL.

In recent years, he has been an exhibition organiser, and today his company, XPO Events, runs MacExpo in the UK and Germany, Adobe Live and LinuxWorld.

FREE FOR ALL?

NICK VEITCH examines the clash of cultures between Open Source and business. Are they really mutually exclusive?

The Open Source phenomenon has profoundly changed the world of IT, but it seems that while many are keen to embrace the new opportunities it offers, still more are confused, afraid and cynical about the benefits of something 'free'. In the same way that passers-by would look at you oddly and cross the road to avoid you if you started handing out five-pound notes in the street, there definitely seems to be some lack of understanding in the business community about the goals and objectives of Open Source.

But here's the great thing – you don't *have* to understand it! It helps, but there's no reason to 'buy into' the Open Source philosophy to benefit from it, at least at the customer level.

WORTH WHAT YOU PAY FOR IT?

There is still very much an idea amongst the business community that software is worth what you paid for it. Obviously *MS Office* is better than *Star Office*, because it costs more? If free software *was* better than commercial software, then nobody would buy any? And so on. Of course, in the real world, the economics of IT are a lot more complicated than that. The real issue for corporate IT isn't the licensing cost, but the idea that support will cost more, or the software will be unreliable and cost more in downtime. In the real world, the ideology of Open Source goes out the window. What the typical IT manager wants to know about software is:

- Does it work?
- What happens if it breaks?
- How much does is it going to cost me?
- Am I going to get fired from my job if/when it all goes wrong?

Choosing to deploy Open Source software used to be 'making a brave decision', though thanks to a huge increase in the number of companies running Open Source solutions nowadays, that isn't anywhere near the barrier that it once was.

Choosing to open-source your own software – well, some would have you believe that demonstrates an ignorance of economic factors, rather than a special kind of courage. And yet, we see – time and time again – that the companies which have done so typically climb extremely quickly up the league tables of their own sectors. In the rest of this feature, we're going to take a quick look at some of these success stories, and the ways in which making money and Open Source software can go hand-in-hand.

LINUX: AN EVIL CANCER?

Surprisingly, a large number of people still seem to believe the 'cancer' argument. Microsoft's CEO Steve Ballmer once infamously displayed a complete lack of understanding of the whole Open Source issue by erroneously describing the GPL licence as a "cancer" that would eat up the value of IP. The contention was that once you started developing software using Open Source, all the software you developed would somehow be 'tainted' by the connection, and companies or individuals would have to give away all their code. This just isn't true. In fact, some high-profile organisations that do the exact opposite by maintaining a 'dual-licence' system, either to divide platforms or levels of support.

TROLLTECH AND Qt

In the Linux developer community, Trolltech is probably one of the best-known examples. The *Qt* GUI toolkit was a bone of contention amongst Linux users for quite a while. It was chosen by original KDE developer Matthias Ettrich simply because it offered the best features; but as it wasn't covered by the GPL, this caused the whole KDE project to be famously excluded from Linux distributions such as Red Hat. In effect, this effectively demonstrates Ballmer's risible cancer argument in reverse!

The situation changed when Trolltech adopted a dual-licence strategy, by allowing the use of the libraries under a commercial licence or a free licence. For commercial development – *ie* to produce a closed-source product – the commercial licence is required, but for Open Source developers, the libraries are obviously simply subject to the terms of normal Open Source software. Trolltech calls this approach *"fair exchange"*. Basically, if you make money out of closed-source products using Trolltech's software, you need to pay the company some money back. Alternatively, the Open Source license allows users to develop, modify and distribute their software freely.

Why does Trolltech adopt this approach? Because it gives the company the best of both worlds. The commercial licence offers a revenue stream, and enables the company to invest in the staff and resources to continue development, as well as make a profit. The Open Source license makes Trolltech a part of the Open Source community, and allows the company to benefit from all of the work done by the rest of the community. On a few occasions, this may be contributed code, but more often than not, it is the legion of Open Source developers who do the quality testing of the libraries, and provide much of the inspiration for new features and functionality. Developers around the world participate in the release cycles, testing beta versions and providing feedback. This is a two-way street for developers, who in return get to see bugs fixed, new features implemented and the freedom to use the software in their own projects. Few of them participate purely to help out Trolltech, but many do because their own projects rely heavily on the *Qt* libraries.

Trolltech and others call this approach a 'virtuous cycle' (see illustration, over page). In effect, the benefits of 'giving' to the community are repaid manyfold. Trolltech has also realised what some other companies have not: the army of Open Source developers – whether they be bedroom geeks, kernel hackers or coders working on large corporate projects – are an almost priceless resource when it comes to thoroughly testing new releases: without them, new versions of *Qt* would be few and far between, and the revenue stream from commercial licensing would be that much more fragile.

WITH A 'VIRTUOUS CYCLE' APPROACH TO DEVELOPMENT, THE BENEFITS OF 'GIVING' TO THE OPEN SOURCE COMMUNITY ARE REPAID MANYFOLD

THE PROOF IN THE PUDDING...

IF YOU'RE STILL WAVERING over whether dual licensing works or not, cast your eye over this list of companies who licensed MySQL under the commercial licence...

- Alcatel
- AOL
- The Associated Press
- Caterpillar
- Cox Communications
- DaimlerChrysler
- Dow Jones
- EarthLink
- Enercon
- Ericsson
- Google
- Hoover's Online
- Hewlett-Packard
- Lucent
- Lufthansa
- NASA
- Nortel
- NYSE
- Omaha Steaks
- Sabre Holdings
- Siemens
- Texas Instruments
- Time Inc.
- UPS
- US Census Bureau
- Yahoo!

MYSQL

Another company that operates on a dual-licence basis is MySQL. The company originally open-sourced its database product in 2000. At the time it wasn't particularly well-known, and didn't have much of a following outside its native Sweden. Open-sourcing the software changed all of that. There are now estimated to be over four million active installations of MySQL worldwide, making it the most popular Open Source database by some considerable length. MySQL is almost certainly the biggest Open Source software vendor (if that isn't a contradiction in terms) too.

Co-founder 'Monty' Widenius explained his notions of why the company had been so successful in a recent interview with SWBusiness:

"We had the courage to try to create something new, and we succeeded. We were ahead of our time – a lot of other companies are only now contemplating this type of business. We're exceptional in the sense that our operations are dispersed all over the world. R&D is based in Finland and sales are based in the USA. The majority of our coders all around the world work from home, just like I do."

In a recent press release from MySQL and Trolltech to promote a speaking tour, both companies espoused the merits of the dual-licensing system, and roped in some expert opinion from elsewhere.

"Dual-license companies combine the software quality and distribution benefits of an open source model with the software licensing revenue model of a traditional commercial software vendor," said Dan Kusnetzky, vice president for System Software Research at industry analyst firm IDC. *"This appears to be a successful business model. It offers customers seeking a traditional packaged software model what they want in terms of documentation and ongoing support. For those seeking freely available software and the open source model of community development and support, they get what they want."*

»

TOO COSTLY TO SUPPORT?

But there are plenty of organisations that also provide specific support services for pure Open Source software. In many ways this is a more obvious and transparent transaction – the customer pays for support and other services, not for the software itself. And thanks to the lack of lock-in, changing your services supplier needn't mean changing the software stack – there is no reason why many different organisations can't offer pure support services for the same software.

One of the most high-profile of these companies is the JBoss group. The JBoss software is already the number one Java application server solution for Java developers and ISVs, and is slowly becoming more widely accepted as a final delivery solution as well as being regarded as just a development and testing server. With big-name users such as AMD, BASF, McDonald's, Siemens and Motorola, it can hardly be described as niche software only used by hardcore developers.

The JBoss Inc part of the group was set up to promote what the company likes to call *"Professional Open Source"*, support services that would be of value and use to large corporate customers. Far from being a weak business model, this separation is believed by JBoss to be one of the key strengths. Sacha Labourey, JBoss's European General Manager explains JBoss's position:

"With a lot of software, what the customer pays and the relationship to the vendor is complex. You might pay so much for a license, an extra 20 per cent for a maintenance contract, perhaps other support options. This is complex for the customer, because even if they are fed up about the service they receive, that's the way it is , and it's just part of the agreement they have. In our case it's really different because we remove from this model the license renewal, and only deal with the service.

"For the customer this is really clear – they are paying you to provide service, and the day you don't deliver that they are going to stop. They haven't lost any upgrade path.

"For us, we must have credibility in support - if we don't then we have no business. We absolutely need to hire the key developers. We aren't interested in support for Open Source in general – we only support projects where we have developers as employees, and that makes sense for us."

Even software for which there is no commercial license or direct support available doesn't exist totally without options. *Samba* is an incredibly mature and reliable project for file-sharing with Windows machines (or other Linux or Mac desktops) over a network. Although now with a large development team, there is no commercial support available directly from *Samba*.

However, there are several organisations that support *Samba* indirectly, because they form part of other solutions offered by these companies. It's no surprise that coincidentally, most of the *Samba* authors also work for these organisations.

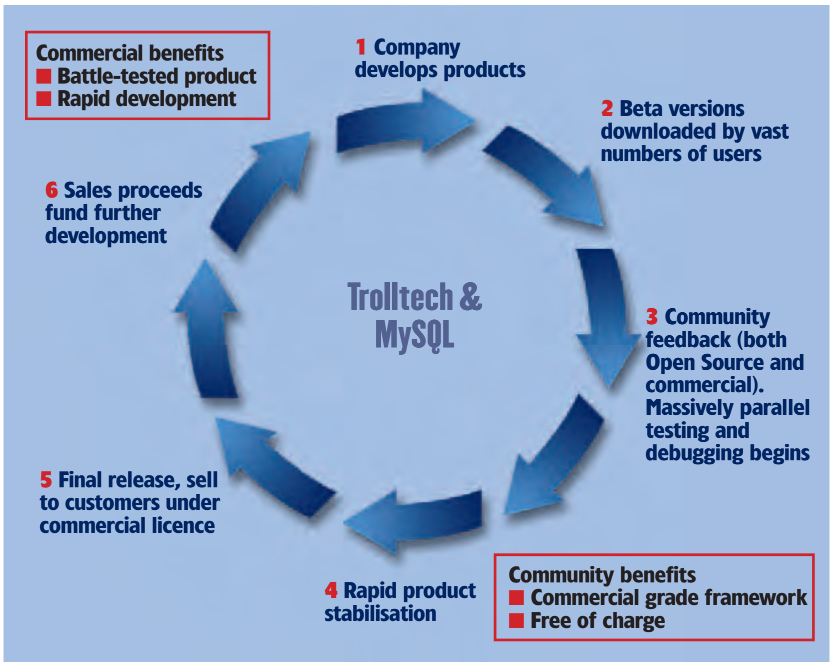
WHAT DO THE EXPERTS THINK?

Bruce Perens and Tim O'Reilly: a difference of opinion

AS MANY OF THE NEW FEATURES TO BE seen in Red Hat Enterprise Linux 4 are first tested in Fedora, Red Hat is also making good use of Open Source for its business model. However, Bruce Perens doesn't think the business model is quite so good in Red Hat's specific case, famously postulating: *"The Open Source community is supposed to produce Fedora so Red Hat can put a stamp on it and charge lots of money."*

On the flip-side, some companies are increasingly finding that even if they don't want to distribute their software back to the community, they can still help – Amazon makes large-scale use of the *Mason* project, which allows Amazon to embed Perl code directly inside HTML. Having gained so much from the community, Amazon now gives back: about half the changes in *Mason1.21* and *Mason1.22* (mostly performance improvements) came directly from Amazon's contributions. Furthermore, as Tim O'Reilly stated in last month's *Linux Pro*, *"If you had the Amazon code you still wouldn't have Amazon"*, meaning that Amazon opening its database up through the Amazon API was as useful to the community as MySQL opening up its code, and didn't compromise Amazon.

THE VIRTUOUS DEVELOPMENT CYCLE


Commercial benefits
■ Battle-tested product
■ Rapid development
1 Company develops products
2 Beta versions downloaded by vast numbers of users
6 Sales proceeds fund further development
Trolltech & MySQL
3 Community feedback (both Open Source and commercial). Massively parallel testing and debugging begins
5 Final release, sell to customers under commercial licence
4 Rapid product stabilisation
Community benefits
■ Commercial grade framework
■ Free of charge

A FUTURE TOGETHER?

From what we have seen here, there is no conflict between Open Source and business in terms of the pragmatism of existing in the real world. Both in the spheres of developing Open Source software, and deploying it, the needs of business are not diametrically opposed to the notion of free software. Far from it, Open Source is actually a key enabler of many businesses. While we might imagine that NASA could have struggled on somehow without MySQL, it is unlikely that many of today's web-based enterprises would exist in their current form without the prevalence of LAMP (Linux, *Apache*, *MySQL*, and PHP/Perl/Python) technology. Nor has open-sourcing software like *MySQL* eroded the value of the IP investment in it – if anything, thanks to the effectively free beta-testing, it has actually *increased* the inherent value.

There may well be cases where Open Source doesn't make sense, but it is certain that Open Source has become an option not only to be used by business, but to actually *generate* businesses. ■■■

BACKUP MADE EASY

With Storix moving from success to success, **PAUL HUDSON** asked CEO **ANTHONY JOHNSON** why backup has never been more important...

When you already have redundant systems waiting in case of system failure, what's the point in backup? If fault-tolerance were this easy, administrators would have fewer sleepless nights, but the reality is that many companies have faulty – or, worse, nonexistent – backup and restore solutions, which mean that no number of redundant systems will help you keep your data safe. The solution, as always, is to call in the experts. In this case that's Storix – long-time Unix and Linux backup solution providers, and makers of *System Backup Administrator (SBA)*. We spoke to Storix's CEO, Anthony Johnson, about the big issues in the backup industry and how Storix is helping companies tackle them...

***LINUX PRO:* What are the key challenges that backup administrators have to deal with?**

ANTHONY JOHNSON: It's hard to narrow down to a few, but here are the top issues that backup administrators bring up regularly:

1 The biggest challenge for backup administrators is full system recovery, also referred to as 'bare-metal restore'. This means the system is completely inoperative and the OS itself must be restored. After the system has been in production for a while, it's usually had various applications installed, undergone numerous patches and updates, and has had users, networks, filesystems and various hardware and software configured and reconfigured a dozen times. Reinstalling the system from the original Linux distribution media can be the start of a very long recovery.

2 Preserving the integrity of database data while backing up live data is another big challenge. In today's online marketplace, you can't simply shutdown the system for backups (at least not for long). A frequent mistake made by many administrators is backing up live 'relational' data. All appears to go well until you have to restore data. Then you find that the data first restored from the backup is often several hours older than the last data restored. This can only be overcome by taking a 'snapshot' of the data at a specific 'point-in-time'.

3 Systems configured for higher I/O performance and availability present complex obstacles to system recovery. There are huge advantages to be had regarding careful placement of disk partitions, implementation of journaled filesystems, logical volume management (LVM) and software RAID devices. But you also need a recovery plan for rebuilding your system as quickly as possible, which is further complicated by even small changes in the target system or disk configuration.

***LXP:* Do you find that many companies still rely on old or poorly maintained backup solutions?**

AJ: Fortunately, most administrators have some form of data backup. But without a system recovery solution, a data backup is not of much value when the system crashes. A question I often ask backup administrators is: *"If your system were to disappear AT THIS VERY MOMENT, is your current backup sufficient to get a new system in operation – quickly, and identical to the first?"* I'm usually met with either blank stares or a look of sudden horror.

Being able to restore your files is important. But equally as important is having somewhere to restore them. If a single disk fails, the system may remain operational. But it will be necessary to recreate a partition map, logical volumes or software RAID devices, then remake filesystems – all before any data can be restored. Few administrators I speak with even know the size of the filesystems. When a failure occurs, they just keep rebuilding them over and over until the data finally restores without running out of space. This is a lot of unnecessary downtime.

Many administrators say they don't do backups because they have RAID. Eek! *NEVER* rely on RAID as a backup medium. Any changes to data in a RAID configuration are instantly recorded on all copies. RAID5 provides single-disk recovery, but is no help in a dual-disk or complete system failure. Most restores are the result of files deleted or corrupted by user and application errors.

***LXP:* What do Storix's products do to make backup easier – and safer – for users?**

AJ: Unlike more general backup products, our largest focus is on system recovery. In the real world, administrators can't be expected to track every change to their system environment, and recovery onto the exact same hardware can't be assumed. *Storix SBA* not only backs up and restores your data, but tracks your system configuration and can automatically rebuild your entire system from the ground-up. To do so, *SBA* must understand your system configuration as well as the operating system it resides on.

With this knowledge at hand, *SBA* can not only rebuild the system as it once was, but can also be used to tailor the system recovery in many ways. For example, you can migrate your partition-based filesystems to LVM or software RAID, change filesystem types, move data between disks, *etc*. This is done using a simple user interface that walks the user through the process.

LXP: **You launched v5.1 of *System Backup Administrator* recently. What does that add?**
AJ: Our first focus of *SBA Version 5* was to add more boot and installation media options. For Linux, this includes the ability to create a network boot image or make a local hard disk boot to the SBA system recovery process. Version 5 also adds many new features for commercial users, including LVM Snapshot backups, use of spare hard disks as backup media, and the ability to copy backups between media or servers. For the home Linux user, we've also added a very inexpensive *SBA Desktop Edition*, which provides most of the data and system backup/recovery features needed in a personal system environment. Other new features include automatic verification of backups upon completion, better hardware support, and improvements to the user interface.

LXP: **Were the new features requested by customers?**
AJ: The new *Desktop Edition* was strongly requested by personal system users. Most of the other features are based on user surveys and other feedback we've received. We provide what we think an administrator will need – and what they tell us they need. *SBA5* was rigorously tested by many of our users, and more changes came from feedback we received during the test period.

LXP: **How do you see the new features benefiting end users?**
AJ: LVM Snapshot backups solve the relational data backup issue I talked about earlier. Since any type of data can be stored in logical volumes instead of partitions, this feature can be used for regular filesystem data or any type of database application. Using spare hard disks for system backups means having the ability to backup and recover an entire system using internal or external, portable USB or SCSI drives, or SAN-attached disks. With our current solution, multiple systems can backup to the same SAN disks, then boot and install from those disks in record time with no other boot or installation media. Home users now have a simple complete product, which can backup and recover their data or entire system. *SBA Desktop Edition* has all the flexibility they need to restore individual files or an entire system, even onto different hardware.

LXP: **What tips would you give our readers if they are worried their current backup solution is insufficient?**
AJ: First of all, *always test your backup and recovery procedures*. You should at least ensure you could recover files and directories from your backups.

Many backup products that claim to perform system recovery using a 'disk image' or 'partition image' backup. This is acceptable *ONLY* if you are restoring to identical hardware. Even a change to the size of a disk can cause such tools to fail, rendering your system unrecoverable. If you don't have the resources to do a full system recovery test, at least make sure your system recovery process supports all of the operating system features you have implemented. For instance, if you created your root filesystem as a Reiser filesystem built on an LVM logical volume, be sure your system recovery tools can rebuild it the same way.

Of course, if you have any doubts, *Storix SBA* is available for free trial. If you just can't part with the $79(US), there's still a free *Personal Edition* for backup and recovery using spare disks.

LXP: **What new features is Storix working on for inclusion in the next release?**
AJ: There is one customer request, coming mainly from home users, that we've not yet provided – DVD support. There are a few Linux backup tools available for writing DVDs, but are generally simplistic, and usually based on the *cdrecord* utility. What is needed is a user-friendly backup application supporting DVDs – one that also provides multi-volume support and full system recovery. Our goal is to eventually provide the same powerful backup automation, verification, labelling and volume support for DVD that we do for tape backups.

LXP: **Where do you see the backup market going over the next few years?**
AJ: With the ever-advancing 24x7 online marketplace, the need for fast online data backups continues to grow. Government and industry regulations are now demanding more frequent backups, higher data availability, and faster recovery times. No doubt, SAN-attached disk solutions are spreading. They provide centralised data storage to multiple systems with much faster access speeds than other network solutions.

We have and will continue to focus on SAN as viable backup and recovery media. I mentioned DVD backup media, but mostly in reference to home users. This is because DVD is still quite slow. There was a time when we thought a tape drive's performance could never come close to that of a disk drive, but tape technologies have narrowed the speed gap dramatically. As DVD advances, I see this as an inexpensive alternative – but not a replacement – to tape or disk backups.

LXP: **One last question: is that *you* in the poster behind you?**
AJ: Uh, yeah, that's one of my more interesting pastimes. And, yes, I do *always* carry a backup 'chute!

Anthony Johnson, Storix CEO – *always* test your backup and recovery procedures so you won't panic should disaster befall your system.

"IF YOUR SYSTEM WERE TO DISAPPEAR *AT THIS VERY MOMENT*, IS YOUR CURRENT BACKUP SUFFICIENT TO GET A NEW SYSTEM IN OPERATION...?"

INFORMATION SECURITY

MEETING REGULATORY REQUIREMENTS

Many businesses take information security for granted, unless there's a problem with it, or an overhaul is needed to comply with regulations, says **JAY G HEISER**, Chief Analyst at TruSecure.

After years of apparent neglect, the Information Security field is suddenly subject to unprecedented levels of external attention. There are two reasons. First, the level of computer crime continues to grow at least as fast as the digital infrastructure does. Security failures are happening, consumers are legitimately concerned, and they are demanding that something be done which helps them to feel that they are less at risk. Second, both government and shareholders are increasingly anxious about the overall corporate management of operational risk, forcing public firms to become more transparent in their risk management efforts. Information security is seen as a form of operational risk that is not only growing in significance, but is managed in a relatively immature way. Corporations and other organisations holding sensitive information have not demonstrated adequate attention to information risk, encouraging most governments and many other external regulators to create a bewildering variety of new rules and regulations. What can a resource-strapped Infosec function do in order to satisfy all these new requirements?

PRIVACY LAWS

Both the number of infosec-relevant regulations and their stringency is on the upswing, but information security is by no means a new phenomenon. Computer researchers were aware in the early 1960s that information automation had a downside, and that continued collection and consolidation of information related to individuals would mean mainframes would soon represent a serious threat to personal privacy. Enacted long before there was anything like e-commerce or the world wide web, the *US Privacy Act 1974* was an early attempt to prevent use of databases from harming individuals. What's interesting about this act is that it protects US citizens from abuses by Federal government agencies. In contrast, the various European privacy regulations have been designed to protect consumers from corporations. The 1995 *EU Data Protection Directive* (subsequently been incorporated into EU national legislations) obliged corporations to take appropriate steps to protect data relating to persons. Raising the bar again, in 1996 the *US Health Insurance Portability and Accountability Act* (*HIPAA*) – though limited only to medical information – applied to both government agencies and private firms. Effective in Summer 2003, California's Privacy Legislation requires any government or corporation anywhere in the world to notify California residents if any of their personal information is lost through a security failure. Reflecting the global reach of the Internet, California lawmakers expect compliance from every e-commerce site in the world that accepts orders from California residents.

FRAUD LAWS

Regulators also feel the need to prevent losses within digital accounting and transaction systems. The *UK Companies Act 1985* aims to prevent falsification of accounting records, including "those of a computerised nature." The *Sarbanes-Oxley Act* (2002) imposed similar internal control requirements on all corporations, domestic and foreign, which file reports with the United States Security and Exchange Commission. Significantly, it makes corporate directors personally liable if they do not implement and manage systems for the reporting of both financial data and non-financial information that is material to the health of the corporation. The act mandates and implies a wide variety of controls, including controls over IT and information security.

The Bank for International Settlements (BIS) is not a government, but its ability to set requirements that apply to all financial institutions engaging in international trade makes it highly influential. It sets requirements on the amount of capital that banks must set aside to protect them against business failure. The set aside amount is a ratio that is calculated according to a bank's overall level of risk. The requirements commonly referred to as the *Basel II Capital Accords*, or just *Basel II*, address operational risk, including a short but significant reference to information security.

WHAT IS PROPER?

Of the hundreds of laws and rules that potentially apply to firms doing business on the Internet, only a very few, such as *HIPAA*, actually delve into specific security requirements. For the most part, the regulations are mandatory guidelines, providing vague expectations for 'appropriateness' or 'in proportion to risk.' However, regulations such as *Sarbanes-Oxley* are quite specific in terms of legal consequences for non-compliance. Simply put, corporate directors can be sent to jail. This has focused unprecedented levels of management attention on the infosec function.

Obviously, every organisation subject to mandatory rules needs to retain its own legal counsel and be cognisant of the specific requirements that apply to its line of business, corporate charter, and international business activities. That said, there is a basic strategy that corporations can follow to position themselves to meet any number of current and future requirements for the management of infosec risk. Corporations must implement information security management systems that both provide and demonstrate a more than adequate level of infosec transparency.

BEING GOOD ISN'T GOOD ENOUGH

Transparency implies two conditions:

1 Relevant risk issues are reliably identified and information on their significance is communicated to the necessary external parties

2 External auditors are satisfied that this is being done.

It isn't enough for a firm to have a relatively low level of information security failure–it must be able to explain why it has a low rate of failure, and it must provide compelling evidence that it is on track to maintain a well-understood level of risk for the foreseeable future.

When requirements are vague and regulators are looking for blood, the safest approach is to do a better job of internal controls than other firms in your sector. Internal control is accomplished by identifying risks, assessing their significance, and managing risks that are deemed most significant. Borrowing several of the most important lessons from Total Quality, risk management is most effective when using a lifecycle approach. This ensures that currently identified risks are being addressed, and that new and unrecognised risks are eventually identified. When internal and external requirements change quickly, it is reassuring to already have a systematic approach in which controls are introduced, managed, and compliance measured.

> "CORPORATIONS MUST IMPLEMENT INFOSEC MANAGEMENT SYSTEMS THAT BOTH PROVIDE AND DEMONSTRATE MORE THAN ADEQUATE TRANSPARENCY"
>
> JAY G HEISER, TRUSECURE

Every type of risk requires its own management techniques, and there are many different opinions as to what control systems provide the optimal approach to infosec risks. What is clear is that external regulators expect all forms of operational risk to be professionally managed, and when they evaluate the efficacy of information security management systems, they expect to see corporations applying control lessons that have been learned in other contexts. It is no longer acceptable for information security to be an *ad hoc* afterthought. ■■■

INFOSECURITY EUROPE 2004

TRUSECURE WAS AN EXHIBITOR AT INFOSECURITY EUROPE 2004, WHICH IS Europe's largest information security event.

Now in its ninth year, the exhibition brought together professionals interested in IT Security from around the globe with suppliers of security hardware, software and consultancy services. The show featured Europe's most comprehensive FREE education programme, and over 200 exhibitors at the Grand Hall at Olympia from 27 to 29 April 2004. If you missed the security event of the year, check out the website www.infosec.co.uk

MANAGE EASY

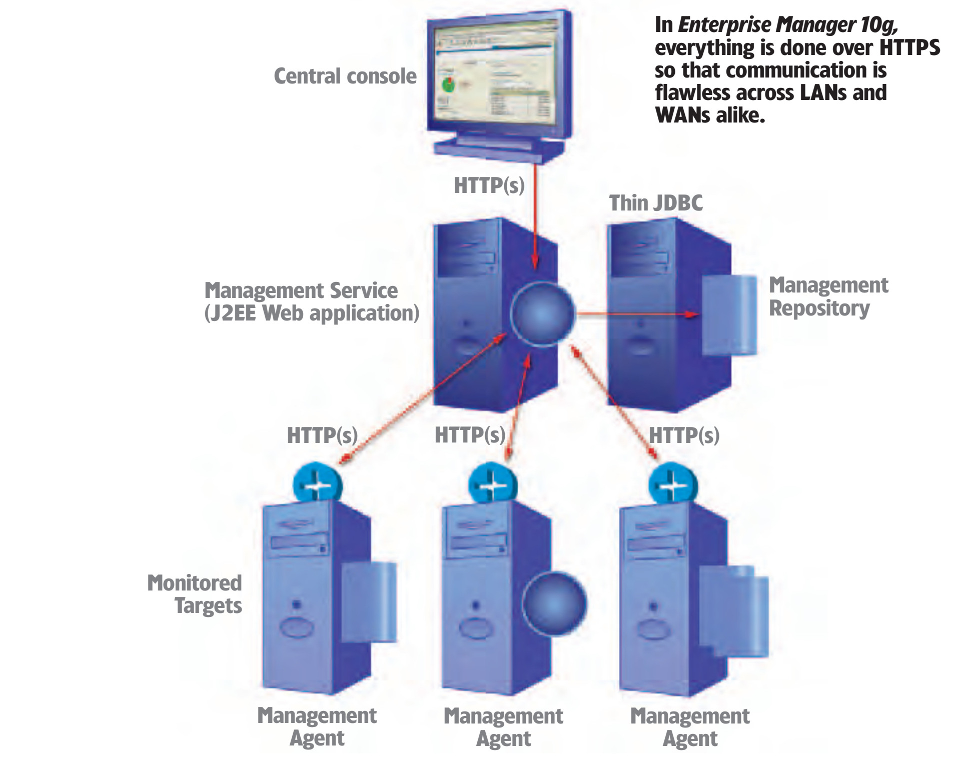
WHY MANAGEMENT IS REDEFINED WITH GRID COMPUTING

Scaling out needn't mean that your staff budget has to double with the number of computers, says **PAUL HUDSON**.

The first two parts of this series investigating Oracle's move to grid computing covered *database 10g* and *Application Server 10g* respectively, and we looked at how grid computing was totally redefining these two areas. However, one area we have yet to look at it how the management of these grid services also needs to change simply because everything is working together now. Although thousands of computers having to work together sounds like the biggest IT management headache since Windows 98, there are ways and means of keeping all your enterprise management under your thumb. Oracle's latest 10g app, *Enterprise Manager 10g (EM10g)*, does just this, and has specifically been enhanced to handle the new database and *Application Server* releases.

GLOBALLY SCALABLE

Though it might sound odd to try to cure the potential grid management problems by adding *another* grid app, *Enterprise Manager10g* doesn't run over a grid – it *manages* a grid. The most obvious thing about *EM10g* is that it runs entirely over the web (preferably through HTTPS, but it can also work over plain HTTP), making remote administration a snap. This is arranged so that administrators communicate over HTTPS with a management agent – essentially a device dedicated to monitoring grid apps, which itself communicates with an Oracle database to store and retrieve management data – where the data readouts and control interface are shown. The management agent then communicates over HTTPS with its monitored target, and is able to retrieve statistics and usage information, as well as also able to execute remote operations such as restarting services.



In *Enterprise Manager 10g*, everything is done over HTTPS so that communication is flawless across LANs and WANs alike.

There are two big advantages to running everything over HTTPS. First, it means sysadmins can work from any device: Linux, Windows, Mac, or even a mobile phone with GPRS. This added mobility means that even if a problem occurs while the sysadmin is abroad, they can get to a computer and rectify it. The second big advantage to HTTPS is that most firewalls allow it through by default, and all the others are easily configurable to do so. In combination with the fact that it's all securely encrypted as standard, this makes HTTPS flexible enough to work across a worldwide WAN with the same configuration used to work across a small LAN.

AUTOMATIC MANAGEMENT

To keep very large grids easily manageable, *EM10g* does a lot of automatic management without user interference. There are three levels to this functionality: checking itself to ensure everything is being monitored suitably, monitoring clients to ensure they have a high uptime and are available for data collection, and also monitoring performance according to requirements laid out by the administrator – if things dip below a certain level of expected performance, *EM10g* can attempt to rectify the situation, or, if that's not possible, it can notify administrators that manual intervention is required. As this is all web-based, the most common notification method is by email, although you can configure this so you get the most helpful warnings possible. Having said that, manual intervention really is a last resort in many respects: *EM10g* is able to do so much without prompting for input that the idea is simply 'set and forget'.

At first glance, you might think that *EM10g* overlaps much of the automatic management functionality introduced into

database10g and *Application Server10g*, but that's not the case. One easy way to think of it is that a grid of eight machines running *database10g* all manage themselves according to their individual needs. Each server will adjust buffers, cache amounts, network flow and other individual metrics that affect only itself, whereas the *EM10g* machine will actively monitor the grid as a whole, making sure that all the machines are being utilised properly and that maximum performance is being squeezed out of the grid. It also provides a centralised points for statistics gathering – without it, you would need to check up on each grid participant individually, which is no easy feat for large grids!

Perhaps the cleverest part of the whole system is that even the setup is automatic. When you install *EM10g* on your grid, it automatically probes the machines around it to see what's available and what can be monitored, installs all the necessary components, sets default performance requirements for the grid as a whole and for each individual machine, and also creates personalised monitoring homepages for each administrator who will work with it. The automatic target discovery functionality is again a common-sense feature: if you have a grid of a thousand computers, you don't want to have to type in a thousand IP addresses!

TO KEEP VERY LARGE GRIDS EASILY MANAGEABLE, *ENTERPRISE MANAGER 10G* DOES A LOT AUTOMATICALLY WITHOUT USER INTERFERENCE

MANAGING A GRID

As grids smoothly grow from ten computers to ten thousand, the administration tools must also be that flexible. Above and beyond the automatic management done by *EM10g* and the ability to access monitoring devices from anywhere in the world, there are three special administration features that keep grid administration manageable.

The first of these is the ability to arbitrarily split the grid up into smaller, more manageable chunks. Although this might sound like it goes flat against the idea of grid computing – that everything should work together as one mass – do remember that the splitting is only for administration tasks. The computers, irrespective of how they are split, still work together as one grid, but the arbitrary splits make division of administration much easier. For example, if you have a thousand communicating computers across the world, you might want to split them into North America, South America, Europe, Asia, Africa, and Australasia; or you might want to split them into x86, AMD64, Itanium, *etc.* For administration purposes, this makes perfect sense: breaking it down by geographical position means that specific admins in that area can be assigned control over each group so that they can respond faster to serious problems. Splitting by hardware also makes sense, as you wouldn't want an x86-trained admin to have to deal with a problem on a 16-way Itanium.

This arbitrary splitting leads directly onto the next helpful admin feature: you can assign roles and task allowances to individual administrators. Unless you run a particularly trusting company, chances are you won't want to give every administrator absolute control over all machines – you'll probably want to segment things up so that some admins get control over some machines, whereas a few 'super admins' get master control. Again, the need to work with very large numbers of machines means that role-based security is the way forward in *EM10g* – rather than having to click 10,000 checkboxes to give a new administrator control over every server, you'd just assign them the role of super admin. Similarly, you could create other roles such as geographical regions or hardware, so that you only need select one or two options to give new administrators exactly the level of access you want.

The final way that *EM10g* eases grid administration is by allowing complex jobs to be timed and executed on a regular basis automatically. As Oracle is able to get in quite deep with the control of *database10g* and *Application Server10g*, your jobs can automate complex tasks such as hot database backups, or you can optionally write your own scripts to do whatever you want on the local system as long as you have the correct permissions. Ideally large chunks of the admin work would be automated through timed scripts, although you are also able to create and store scripts that are executed on an *ad-hoc* basis – they run when you request them to run through the web management interface.

MAKING THE MOST OF IT

Although the management facilities that come as standard in *database10g* and *Application Server10g* are more than capable of handling local administration tasks, *EM10g* adds so much more to the overall management capabilities of the grid that it really is a required purchase if you have a grid of over fifty computers. The most immediate value you'll get will be a lowering in admin costs as fewer people struggle to manage the computer array; but you'll also find a healthy increase in performance as *EM10g* adjusts settings for you.

One factor you should take into account is that you have a fairly small grid (say under 25 computers) you may not get sufficient ROI through *EM10g* – we suggest you give Oracle a call and see whether it can work for you.

AVOIDING COMMUNICATION OVERLOAD WITH COLLABORATION SUITE

HERE AT *LINUX FORMAT'S* PARENT company, Future Publishing, we're about to start a pilot program using Oracle's *Collaboration Suite*, and, happily, the entire system will be running on Red Hat Linux. If you have yet to come across *Collab Suite*, it is a powerful new middleware suite from Oracle that encompasses web conferencing, automatic file versioning, email, calendaring, voicemail and fax, and wireless connectivity all in one product. In the same way that *Application Server10g* (*AS10g*) brings together numerous disparate applications on the web into one product that is much bigger than the sum of its parts, *Collab Suite* does the same for office workflow and data management. Also in the same way as *AS10g*, *Collab Suite* actually works out much cheaper than if you were to purchase all its constituent parts separately.

Many people still complain that there is no real equivalent to *MS Exchange* on Linux, but this seems set to be the biggest *Exchange*-equivalent on Linux yet, and even surpasses *Exchange* in a number of ways. We're looking forward to running a special feature on Oracle's *Collaboration Suite* once our pilot has been successful – watch this space!