news@UK

The Newsletter of UKUUG, the UK's Unix and Open Systems Users Group
Published electronically at `http://www.ukuug.org/newsletter/`

Volume 13, Number 4 ISSN 0965-9412 December 2004

Contents

News from the Secretariat

Jane Morrison

At our October Council meeting Ray Miller was elected as UKUUG Chairman and James Youngman agreed to continue as UKUUG Treasurer.

Charles Curran was granted an Honorary membership of UKUUG to thank him for his efforts over the last six years.

Ray has also been working extremely hard putting together the programme for the Winter 2005 event. You should find the full information booklet and booking form enclosed with this mailing. There is a strong programme this year and we hope you will be able to attend.

The date and venue for the LINUX 2005 conference is now confirmed. The conference will be head at the University of Wales, Swansea between the 4th and 7th August 2005.

The call for papers will appear on the website very soon. We are currently looking for sponsors for the event and if you know of any company who would like to get involved please let me know.

Do you need to buy some books? Don't forget that your UKUUG membership subscription allows you to receive a discount of 27.5% on O'Reilly publications – and we also have GNU Press titles in stock – see the UKUUG web site for a list of titles available and prices.

The annual subscription invoices will be sent out in January, please look out for your invoice and as always prompt payment will be gratefully received!

Unbelievably it is that time of year again and I would like to wish you all a very Happy Christmas and a Peaceful New Year.

The Secretariat will be closed 21st December to January 4th 2005.

Please note the copy date for the next issue of the Newsletter (March 2005) is 18th February 2005.

UKUUG Winter Conference 2005

Ray Miller

UKUUG's next Winter Conference will take place at the Paragon Hotel, Birmingham, on Thursday 24th and Friday 25th February 2005. We have a packed programme lined up, with talks on the theme of "Security and Networks".

On Thursday morning, Allison Randal, president of the Perl Foundation and project manager for Perl 6, will deliver a Perl 6 Workshop. The main conference begins in the afternoon with a keynote by Wietse Venema, author of TCP Wrappers and the Postfix MTA. We also have speakers from the OpenBSD and FreeBSD projects; Oracle; Hewlett-Packard; University of Oxford; University of Cambridge; and many more – see the conference web site for full details and up-to-date information:
`http://www.ukuug.org/events/winter2005/`

A conference dinner has been organised for Thursday evening, at Birmingham's Chung Ying Gardens restaurant, only a ten-minute walk from the hotel. Conference talks continue on Friday morning, and are followed by a series of lightning talks in the afternoon.

This conference is a must-attend event for system and network administrators. As well as the technical talks, the conference provides a friendly environment for members to meet, learn and enjoy lively debate on a host of subjects. See the newsletter insert or the conference web site for more information and booking details.

We still have a few slots left in the lightning talks programme. There are also opportunities for sponsoring some aspects of the conference, and for table-top displays at the venue. Please contact us to discuss the possibilities: `winterconf@ukuug.org`

Announcement: UKUUG Award 2005

Applications are invited for the UKUUG Award 2005.

The value of the prize shall be £ 500 plus a pass to the Open Source Convention.

The closing date for submissions is Saturday, 2 April 2005.

The UKUUG Award (previously known as the UKUUG Open Source Award) is given annually (if submissions of sufficient merit are received) for a significant contribution to free and open source software; this might be in the form of an article or paper, software product, or other contribution.

The purpose of the Award is to encourage and foster developers and other contributors in or connected with the British Isles. We are looking for a work or project that is, or might be, significant in the world of free and open source software. The Award is not limited to (recent) UK students although special encouragement is given to entries therefrom and the best of those entries will be recognized.

Initially, a one-page summary (accompanied by an abstract and a short biography of about 250-500 words) of the work should be submitted via www.ukuug.org/osa. If the work is part of a joint project, the personal contribution should be stated clearly.

The judging panel will include representatives from UKUUG, UK computer science departments, and the wider community. If the judges are unable to distinguish between the merits of the best candidates, the prize shall be divided accordingly. No person may be awarded the prize more than once.

UKUUG are giving a prize of £ 500, to which O'Reilly UK Ltd (www.oreilly.co.uk) are generously adding both a pass to the Open Source Convention (conferences.oreillynet.com) and monies towards travel and accommodation.

The Award for 2004 was made to Julian Field of the University of Southampton for his work in creating, developing, and supporting MailScanner, the highly respected e- mail security system.

UKUUG Strategy

Ray Miller

Picking up on discussions at the recent AGM, we will be holding a strategy meeting after the Winter Conference in Birmingham. This will provide an opportunity for members to discuss UKUUG's aims and objectives, and strategies for meeting those targets.

This will start at 1545 on Friday 25 February, at the Paragon Hotel, Birmingham - immediately after the Winter Conference. This is open to all UKUUG members, not just conference delegates. Please come along and help influence the direction we take over the next 12-18 months.

UKUUG Apple Special Interest Group

A new Apple Special Interest Group within the UKUUG, coordinated by Graham Lee has just been created. The scope of the SIG is Mac OS X, Darwin, and anything involving using, developing for or maintaining Macs within a UNIX environment. It is hoped that we will be able to organise regular events of interest to the Apple community but also to other UNIX users in general, such as speaker events, BOFs and anything else you would like to see.

This leads us onto the definition of "anything else you would like to see". A mailing list, apple-sig@lists.ukuug.org has been set up for the SIG and anyone with an interest (a Special Interest, no less) is encouraged to subscribe and discuss potential events for the SIG. A BOF will be organised for the early new year, at which we can thrash out a more concrete plan for the group. I look forward to seeing you there and on the mailing list (which is of course open to any Apple-related discussion).

You can subscribe to the SIG mailing list via the web page
http://lists.ukuug.org/mailman/listinfo/apple-sig

or by sending an empty message to apple-sig-subscribe@lists.ukuug.org

Please subscribe to the list or drop Graham a line (leeg@ukuug.org) if you would like to get involved.

Knoppix 3.6 CDs

Ray Miller

UKUUG recently teamed up with the JISC Open Source Advisory Service, OSS Watch, to produce and distribute Knoppix 3.6 CDs. This version was remastered by OSS Watch to include software especially relevant to UK higher and further education; see
http://www.oss-watch.ac.uk/resources/osswatchknoppix.xml

for more information. The ISO is also available for download from the OSS Watch site:
http://www.oss-watch.ac.uk/downloads/

You should have received a copy of the CD with the minutes of the UKUUG AGM distributed in October, and we still have a few copies going spare. If you would like one for a friend, or to distribute at a meeting, please contact the UKUUG Secretariat.

The UKUUG jobs mailing list

Ray Miller

From time to time, UKUUG receives notice of employment opportunities in areas relating to Unix, Open Source, and Free Software. These are posted to the jobs@ukuug.org mailing list. To subscribe, send an empty message to jobs-subscribe@ukuug.org , or visit the web page:
http://lists.ukuug.org/mailman/listinfo/jobs/

If you are an employer and would like to notify members of an opportunity in a relevant area, please post a brief announcement to jobs@ukuug.org . This is a moderated list, so your post may be held for a short time pending approval by one of our moderators.

UKUUG Diary

Sam Smith

UKUUG has put together a diary of events in the UK which may be of interest to the UKUUG membership:
`http://www.ukuug.org/diary/`

It is not limited to UKUUG events, and is open to anything which may be of interest to the UKUUG membership (or those who share the UKUUG interests of UNIX or Open Standards). Please submit any events you are aware of that are missing.

iCal, XML and RSS event feeds are available for use and reuse of the data.

Software Patents in the EU

European Open-Source Luminaries

Appeal to EU Council Against Software Patents

European nationals who created Linux, MySQL and PHP issue joint statement ahead of EU Council meeting – "The proposed software patent directive is deceptive, dangerous, and democratically illegitimate"

Munich, Germany (23 November 2004). The three most famous European authors of open-source software have issued an appeal against software patents on NoSoftwarePatents.com. Linus Torvalds (Linux), Michael Widenius (MySQL) and Rasmus Lerdorf (PHP) urge the EU Council, which will convene later in the week, not to adopt a draft directive on software patents that they consider "deceptive, dangerous, and democratically illegitimate". They also call on the Internet community to express solidarity by placing NoSoftwarePatents.com links and banners on many Web sites.

This announcement comes after an eventful week on the software patent front. The Polish government clarified that it does not support the legislative proposal in question, and Microsoft warned Asian governments that they could face patent lawsuits for using the Linux operating system instead of its Windows software.

The open-source programs that were created by Linus Torvalds, Michael Widenius and Rasmus Lerdorf form three of the four parts of a technology stack commonly referred to as LAMP by the first letters of its components. The combination of Linux (operating system), Apache (Web server), MySQL (database) and PHP (programming language) is an industry standard that powers millions of Internet servers worldwide. Linus Torvalds and Michael Widenius, the Finnish Software Entrepreneur of the Year 2003, are Swedish-speaking Finns. Greenland-born Rasmus Lerdorf is a famous Dane according to Google.

The joint statement stresses that software authors are well protected by copyright law while software patents establish the law of the strong, which creates more injustice than justice. The draft legislation on which the EU Council reached a disputed political agreement on May 18th is called deceptive because it leads laymen to believe that software is excluded from patentability while actually containing a number of passages that would legalize software patents in the EU, the broadest one of which is its article 5(2).

Particular emphasis is given to the fact that an adoption of the proposal without a formal vote, as a so-called A item, would lack democratic legitimacy. Under the Act of Accession, new voting weights apply in the EU Council from this month on, and the countries that supported the proposed legislation on May 18th fall short of a qualified majority on today's basis. Additionally, the national parliaments of two of the supporting countries (Germany and the Netherlands) have spoken out against the proposed legislation.

In separate public surveys of IT companies by the European Commission and the German government, a vast majority (94% and, respectively, an estimated 99%) of respondents opposed the patentability of software. However, a small group of multinationals and the patent system push for an extended scope of patentability.

The full text of the statement is available on the Internet in (initially) 11 languages.

About the NoSoftwarePatents.com Campaign

The NoSoftwarePatents.com campaign was launched on October 20th in initially 12 languages and is supported by three IT companies (1&1, Red Hat, and MySQL AB). More information on the campaign is available on the campaign Web site.

Contact Information

For further information concerning this announcement or the NoSoftwarePatents.com campaign, please contact: Florian Mueller
Campaign Manager, NoSoftwarePatents.com
telephone +49 (8151) 651850
`press@nosoftwarepatents.com`

Announcement: First Australian UNIX Developer's Symposium

We have received a call for participation for AUUG's 2005 UNIX Developers' Conference, which will be held in Adelaide on 8 and 9 April, 2005.

Full details are available at
`http://www.auug.org.au/events/2005/auds/`

SANE 2004 Conference Report

Ray Miller

September 30th 2004 saw the opening of the 4th International System Administration and Network Engineering Conference (SANE) at Amsterdam's RAI conference centre. The conference was organized by the Netherlands UNIX User Group (NLUUG), co-sponsored by Stichting NLnet, with cooperation from USENIX, the Advanced Computing Systems Association.

A SANE conference has been held every two years since the first was organized in 1998 "...to strengthen the European ties between the National UNIX User Groups and their members", in the spirit of the former EUUG/EurOpen. I had attended SANE 2000, held in Maastricht, so was delighted to receive an invitation from NLUUG to represent UKUUG at SANE 2004.

The conference itself was preceded by three days of tutorials - a very strong programme with five parallel streams throughout the three days. Topics ranged from networking (IPv6, firewalls, wireless, IP telephony), through operating systems (FreeBSD 5.2 code walkthrough, Linux 2.6 process management), to popular applications (MySQL, Postfix, OpenLDAP, Samba). Every SANE conference has also featured a Black Hats Session, which is obviously popular: this year's ("Black Hats Session IV: Developments in Security") was run on Monday and repeated on Tuesday.

Work pressures prevented me from attending the tutorials, but I arrived at RAI on Wednesday evening just as Richard Stallman was finishing his presentation, "The Danger of Software Patents". Stallman had travelled to Amsterdam earlier in the day and joined in the demonstration for innovation without software patents held in Amsterdam's Dam Square. The demonstration was organized to coincide with a high-level EU conference on future ICT policy in Europe (initiated by the Dutch government in their 2004 Presidency of the EU), also being held in Amsterdam. Enough politics (for now).

Wednesday evening also saw the SANE Free Software Bazaar, a free event open to non-delegates. Here you could meet and chat informally with developers from the Debian project, OpenBSD, FreeBSD, CAcert, and many others. Birds-of-a-feather sessions covering Samba, KDE, MMBase, KeyWorx, and VIM were also held on Wednesday evening.

The conference proper started on Thursday morning with a keynote by Paul Kilmartin of eBay, Inc, "eBay through the eyes of the Systems Administrator". This was a very interesting talk about the challenges of managing the IT infrastructure behind a (rapidly) growing company, where downtime means losing real money (eBay currently transacts business worth more than USD 1000/second). The most important point I came away with was this: when you are planning for high availability, you do not want to be at the bleeding edge, you want to be doing what other HA sites are doing. Unfortunately for eBay, this is not always possible: they are, after all, one of the world's largest online retailers.

Another important point from Kilmartin's talk was that they are never under the illusion of having solved a problem: while a new system might handle today's workload, eBay's growth is such that the lifetime of any solution is strictly limited. Kilmartin ended his talk with a section entitled "Why I Hate Vendors". Anyone who has dealt with a vendor support desk more interested in closing a trouble ticket than actually solving a problem will have a lot of sympathy with him.

After the keynote, the conference split into two streams: refereed papers, and invited speakers. I stayed with the invited speakers for the rest of the morning.

The first of these was Arjen Lentz of MySQL AB, with "MySQL Roadmap - What we have now and where we are heading". He covered some history of the MySQL project, their development procedures and release schedule, and MySQL's current (and planned) features. Whenever he was talking about a feature, he said a few words about the developer behind it: their background, where they are in the world, and how they came to be involved with the project. This added a personal dimension to what might otherwise have been a dull list of features, and also emphasized the global bazaar nature of MySQL development.

Next was Wietse Venema's "Open Source Security Lessons". He began his talk with some history, taking us back to the time when Eindhoven University in the Netherlands was first connected to the Internet. One "unofficial" user of their systems was causing problems for system administrators: they cleaned up after their activities with "rm -rf /". In an effort to track down this intruder, Venema wrote the first version of what we now know as "TCP wrappers".

He went on to talk about the press response to his and Dan Farmer's release of SATAN, the network security vulnerability scanner: "It's like distributing high-powered rocket launchers throughout the world, free of charge, available at your local library or school" (San Jose Mercury). As it turned out, the release of SATAN did not result in an increase in reports of computer break-in activity, and SATAN proved a useful addition to the system administrator's toolbox for many years.

He then talked about Postfix, and the role its release had in bringing open source software to the attention of IBM's senior management. Finally, he came to the debate about open versus closed source software and security, where he thinks the protagonists are missing the point: "...when a system isn't built to be secure, then it will be like Swiss cheese no matter how many security patches you apply". He pointed out that this is not a new insight, and quoted a 30-year-old paper saying essentially the same thing.

After lunch, I moved to the other lecture room for the refereed papers: "Lambda Networking in

NetherLight" by Erik Radius of SURFnet; then "Traffic shaping for large-scale web services" by Angelos Varvitsiotis of the Greek Research and Technology Network.

The first of these was a technical talk about using different wavelengths of light (lambdas) to transmit multiple data channels over a single optical fibre (dense wavelength division multiplexing). As well as the technical aspects, Radius talked about NetherLight's global connectivity (which includes StarLight in Chicago, and UKLight in London), and potential uses for the technology (for example, high-bandwidth GRID computing).

From high bandwidth to low: Varvitsiotis's talk was about traffic shaping for web servers with an uplink bottleneck. He used an Apache module, `mod_mimetos`, to set the IP type-of-service value according to the MIME type, file size, directory, etc. of the content being delivered, in conjunction with a class-based queuing (CBQ) scheme and a set of filters to map ToS values to particular queues, implemented using the Linux kernel's advanced routing and traffic control mechanisms. He also updated Apache's `mod_mime_magic` module to bring it into line with the latest "file" code.

Varvitsiotis then used data gathered from his University's cache logs to generate driver data for a simulation, and ran different workloads against an uplink-throttled web server. The results of these experiments are detailed in his paper.

The next refereed paper, "TCG 1.2 - fair play with the Fritz chip?", was presented by Rudiger Weis of Vrije University. This was an entertaining (but nevertheless worrying) look at the latest proposal from Microsoft and other members of the Trusted Computing Group (TCG).

The concept of trusted computing is to place an especially trusted observer, or "Fritz chip", into information-handling devices, to prevent even the device owner from carrying out certain operations: the owner gives up some control of their device in return for the ability to verify a device's "trustworthiness".

While the proposed architecture will offer only limited protection against worms and viruses, it offers a lot of features that can be used to protect a personal computer against its owner, especially in the field of Digital Restrictions Management (in the words of Ron Rivest, "...you are putting a virtual set-top box inside your PC. You are essentially renting out part of your PC to people you may not trust").

Cryptographers and privacy organizations have pressurized the TCG into modifying their proposals, and the recent TCG 1.2 specification does address some of their concerns. There are, however, still worries about backdoors, potential compatibility problems between Trusted Computing and Free (GPL-licensed) Software, and patent issues (an official Microsoft statement reads "...Much of the next-generation secure computing base architecture design is covered by patents, and there will be intellectual property issues to be resolved. It is too early to speculate on how those issues might be addressed.").

The final talk of the day was a choice between the invited speaker, John Nelson on "Special Effects on the Movie 'I, Robot' ", and Clifford Wolf's refereed paper on "Distributed Software Development using Subversion and SubMaster". I opted for the latter.

Some of you will already know Clifford Wolf as the project leader for ROCK Linux. Just over a year ago, the ROCK Linux project decided to switch from CVS to Subversion. In the first half of his talk, Wolf covered the basics of revision control systems and introduced Subversion itself. He then moved on to discuss SubMaster, and it was here that the talk started to get interesting.

Like CVS, Subversion is a centralized revision control system, where only privileged project members have commit access to a central repository. Other developers must submit patches via a mailing list, where they can easily be overlooked.

SubMaster, developed by the ROCK Linux project, is an attempt to address this problem and provide for a distributed development model. SubMaster provides scripts that make it easy for developers to create and manage their own branches (in their own local Subversion repository), keep them synchronized with the central repository, and send patches upstream. It also provides a CGI script to manage patch submission, collect feedback, make regression tests, and apply patches to the main tree.

But a conference is about more than just technical talks, and SANE is no exception. There are opportunities to chat informally with peers during the refreshment breaks, but there's nothing like being thrown together on a boat with an unlimited supply of beer to break the ice.

The SANE 2004 social event on Thursday evening began as something of a mystery tour, with three "bendy buses" setting off across the city, attempting a three-point turn on a dual carriageway, then dropping us in the middle of nowhere. After a short walk through a residential then industrial area, we arrived at a boat yard and boarded a boat for the evening's cruise. Entertainment was provided by the Bucket Big Band (I counted seven saxophones, a clarinet, trombone, two trumpets, two guitars, a drummer, and a very energetic conductor). As well as unlimited drinks, a buffet provided plenty of Indonesian food, making for a very enjoyable evening. Better still, by the time we docked, the bus drivers had found the boat yard, so there was no need to repeat the walk.

The first invited speaker on Friday morning was Geoff Halprin of The SysAdmin Group, with "The Changing Face of System Administration". Halprin discussed the challenges facing modern-day system administrators and the often conflicting priorities: troubleshooting, user support, infrastructure projects, keeping our skills up-to-date. He stressed the importance (to system administrators as well as managers) of measuring how much time is spent on each of task, and of maintaining the correct balance (learning and infrastructure projects should not lose out to short-term objectives).

I switched to the refereed papers stream for the next two talks, "High Available Loadsharing with OpenBSD" by Marco Pfatschbacher, then "Deployment of Worldwide IDS Networks" by Matthias Hofherr. Both of these speakers work for GeNUA mbH, a German IT security consultancy.

Pfatschbacher presented a paper describing work carried out as part of his diploma thesis about High Availability VPNs. In a traditional load balancing setup, the load balancer is a single point of failure unless a second, redundant, load balancer is introduced. As with many HA solutions, this introduces extra complexity. Pfatschbacher came up with a nifty idea to provide HA and load balancing without this complexity.

He implemented a new kind of network interface in OpenBSD, a virtual Ethernet interface, or veif. The veif can be assigned an arbitrary MAC address, effectively providing two network interface cards in one. Thus two hosts on the same network can share a common MAC and IP address without changing the MAC addresses of their physical interfaces. Each host remains individually addressable, while packets sent to the common address are seen by both hosts.

Of course, this presents problems on a switched network, so his next trick is to make a switch behave like a hub. To achieve this, veif never sends any packets with its virtual MAC as a source address (think proxy ARP), so the switch never learns the whereabouts of the common MAC address.

The next step is to ensure that, although all packets are seen by both hosts, each packet is only processed by one host. Pfatschbacher introduced an option to OpenBSD's pf to filter packets based on a hash of the source and destination IP addresses and ports. One host is configured to drop all packets in one half of the hash space, and the other host to drop all packets in the opposite half.

OpenBSD 3.5 introduced support for CARP (Common Address Redundancy Protocol), which utilizes virtual MAC addresses to enable multiple machines on the same local network to share a set of IP addresses, while ensuring that these addresses are always available. Pfatschbacher used CARP for monitor and failover of the pf-hash configuration: if one host fails, its hash range is migrated to one of the remaining CARP hosts.

In the next talk, “Deployment of worldwide IDS networks”, Hofherr presented a case study featuring a fictional company, BigCorp, who wanted to employ a network intrusion detection system in their offices across the globe.

Hofherr described a hierarchical solution, with IDS sensors analyzing traffic and generating alerts that are fed upstream to a “Central”. The sensors and the central communicate over a dedicated management network, both to lessen the burden on the production network, and to reduce the likelihood of an attacker analyzing the IDS data. The solution was based on the open source IDS Snort, with a central server running PostgreSQL. Administration is over https to an Apache server, using client certificates for authentication.

Hofherr discussed the different possibilities for traffic capture, their chosen solution (Ethernet Tap devices), the problems this introduced for Snort (and how they solved them), and the protocol for communication between the sensors and central servers. He also discussed security, availability, and monitoring of the IDS infrastructure itself.

He concludes that, although installation of a single network intrusion detection system is well understood and documented, implementing a distributed IDS presents new problems. While there are no out-of-the-box open source solutions, the software components do exist and the challenge is in coming up with a robust, secure, and conclusive design.

A meeting of national Unix User Group board members had been called for Friday lunchtime. The Netherlands (NLUUG), Norway (NUUG), Denmark (DKUUG), United Kingdom (UKUUG), and Croatia (HrOpen) were all represented here. Discussion focused on how the national groups might work together, for example, reciprocal agreements enabling members to attend national UUG events at the local members’ rate. DKUUG is planning to revitalize the defunct EUUG/EurOpen and put the content of old EUUG magazines online, and NUUG has digital video footage of some of its talks available.

It was interesting to meet with the other UUG board members and to see the common challenges we are facing. The meeting engendered an excellent spirit of cooperation, and I came away feeling quite optimistic. The challenge remains in turning ideas into concrete actions, and following through on those actions.

I returned to the invited speakers for the remainder of the conference. This stream started off after lunch with a talk on “Dutch Law Enforcement vs High Tech Crime” by Pascal Hetzscholdt, a policy advisor to the Dutch National Police Agency. Hetzscholdt is currently involved in setting up a High Tech Crime Centre in the Netherlands.

He talked about the challenges faced by the police in tackling the new “cyber crime”, and the links between high tech crime (phishing, fraud) and organized gangs often involved in drug trafficking and arms trading. These links can make it hard to decide which agency should tackle the problem: fraud investigators, because of the financial aspects of phishing? “cybercops” for their technical expertise? drug enforcement agencies when the money is used for drug trafficing?

Fighting IT crime is not seen as a “cool thing” - sitting in front of a computer screen is not as exciting as a high-speed car chase. And shouldn’t priority be given to more shocking crimes like murder, rape, kidnapping? In the Netherlands, these priorities are decided by the public prosecutor who often does not recognize the significance of computer crime, but knows that it can be costly to find the IT expertise required to fight it.

Hetzscholdt appealed to the system administrators and Internet service providers in the audience for their help: the police need our expertise. But he was not given an easy time during audience questioning: many are unhappy with legal requirements imposed on ISPs to collect logs and data about their users activities and meet the costs of storing this for long periods of time.

Next came my favourite talk of the conference, Sjoera Nas of Bits of Freedom on “The Multatuli Project: ISP Notice & Take Down”. Under the European directive on electronic commerce, Internet service providers risk liability for hosting apparently illegal content from their customers. This is quite different from the situation in the United States, where the DMCA provides a safe harbour for service providers.

In 2003, three researchers from the Oxford Centre for Socio-Legal Studies conducted a small experiment with notice and take-down, to see if the different legal frameworks made any difference in practice. They published an article (an extract from John Stuart Mill's "On Liberty", about freedom of speech) on a homepage in the UK and one in the USA. This was clearly marked as dating from 1869, and belonging to the public domain.

They then sent a fake complaint to the two ISPs, using an anonymous Hotmail address. The UK provider removed the homepage within 24 hours, while the US provider insisted that the complainant declare they were acting in good faith (this is one of the safe harbour provisions in the DMCA). Not wanting to risk the next (fraudulent) step, the researchers stopped there.

Bits of Freedom organized a similar experiment this summer, involving ten Dutch ISPs. They uploaded some text by the famous author Multatuli (Eduard Douwes Dekker), dating from 1871. Again, their homepage clearly attributed the text and stated that it was in the public domain.

Seven of the ten providers took down the homepage, one within 3 hours of receiving the fake complaint. Only one provider showed any distrust about the origin of the complaint, and only one demonstrated that they had actually looked at the page in question. In one case, the customer was not even informed of the complaint, and in another, the customer's personal details were forwarded to the complainant. Two of the ISPs did not reply at all to the email sent to their official abuse addresses.

Nas concludes "It only takes a Hotmail account to bring a website down, and freedom of speech stands no chance in front of the Texan-style private ISP justice".

The final talk of the conference was by Peter H. Salus, the famous USENIX bookworm. His talk "UNIX and the ARPAnet/Internet at 35; Linux a teenager; still in court", gave a historical perspective on the SCO Group's attack on Linux through the court system. Salus interspersed his many slides of penguin photos with copies of legal documents from the SCO Group court cases, giving a light-hearted view of the proceedings.

Throughout the conference, more than a dozen technical posters were on display in the lobby: an alternative method for authentication, authorization and accounting for Windows 2000/XP systems; PPTP must die; CAcert; and more. The prize for best poster was awarded to John Borwick of Wake Forest University for his poster on "LDAP for Systems and Network Engineering". This described a method for storing DNS and DHCP configuration data in an LDAP database, and using Perl scripts to retrieve the data and generate configuration files.

There was also a prize for best paper, which was awarded to Luca Deri for his paper "Improving Passive Packet Capture: Beyond Device Polling". Deri proposes a new approach to passive packet capture which, combined with device polling, allows packets to be captured and analyzed at (almost) wire speed on Gbit networks using a legacy PC.

After presentation of the prizes and 'thank you's to the many volunteers who helped to make the conference run so smoothly, Quiz Master Kelvin Henney took over with the inSANE quiz. Two teams were drawn "completely at random" from the business cards solicited earlier in the day, and pitted against each other and the Quiz Master's "completely fair scoring".

You really had to know your geek culture to do well in this quiz - but that alone was not enough. There was audience participation too, with each team having to guess how the audience would respond to "yellow or green" questions. For example, the Quiz Master would shout "Yellow - Python, Green - Perl", the teams would have to write down their answers ("yellow" or "green") before the audience voted by holding coloured cards in the air.

After one team had been eliminated, the three members of the remaining team contended with each other for prizes of books, posters and T-shirts. The quiz was a fun way to end a very enjoyable conference.

I was impressed both by the professionalism of the organization, the quality of the talks, and the smooth running of the event. RAI offered excellent facilities, and the organizers had provided wireless networking throughout the conference area, as well as a terminal room with Internet access for those of us traveling without laptops.

Congratulations, NLUUG, on another excellent conference! I am looking forward already to SANE 2006, and heartily recommend it to anyone else with an interest in network or system administration. You can find out more about past and future SANE conferences at
`http://www.sane.nl/`

UKUUG Apple technical briefing: report

Sam Smith

A successful second Apple Technical Briefing, entitled "OS X and High Performance Heterogenous Environments" which was organised by UKUUG but sponsored and supported by Apple, took place at the Institute of Physics in London on the 1st November.

Sabah Salih from the University of Manchester spoke on his experiences of using OS X in large international physics experiments, and how his Xserve cluster integrates seamlessly with the pre-existing nodes and clusters in his group.

Sabah and his colleagues are involved in many high profile international projects where large amounts of data are produced to be stored and analysed over periods measured in years.

OS X is gaining acceptance in the scientific community for running most UNIX applications (for those that don't run as yet, work is being done so they will) while also being a friendly, responsive and stable platform on which to work. Additionally, the importance of a single, standard, high performance hardware, Unix vendor who will stand behind their systems for the multi-year lifetime of a project means a potential end to the problems of a PC dying, only for its replacement to have a different type of network card/processor etc which causes the standard image applied to all machines to fail.

Ken Tabb of the Neural Systems Group from the University of Hertfordshire provided an Introduction to Cocoa Programming. Covering briefly the heritage of Cocoa and the ways it can be used (Ruby, Java, AppleScript, Python etc), the main focus of the talk was a mixture of Xcode (Apple Developer Tools) and the ease at which you can write arbitrary Apple Aqua interfaces to sit on top of whatever you are doing.

Within the Xcode suite, Apple provide a large number of tools designed to maximise the productivity of developers by making it easier to do things that should be easy, while making hard things possible. In particular, Interface Builder was demonstrated, using standard templates and Apple's WebKit to create a simple, but fully functional web browser.

A second demonstration was preceeded by an introduction to Objective-C, including comparisons to other languages. The audience, most of whom were familiar with some variant of C, followed carefully as a simple, yet potentially very powerful, worked example of a graphical wrapper for a shell script was written; The demonstration also covered how to use Interface Builder and other parts of Xcode to produce a working Aqua application. Part of this was a brief mention of the frameworks available within OS X for reuse, and how to find more information.

After lunch, Jordan Hubbard, one of the founders of the FreeBSD project and head of BSD Engineering at Apple covered OS X, where it came from, how it got here, and where he can see it going in the future. About 60% of OS X is available as Open Source. Apple view Darwin (the Open Source UNIX/Mach core of OS X) as integral in keeping OS X as a whole tidy and manageable. As such a large part of OS X is open (including the Darwin parts sourced from Apple and others including Tcl, Perl, Python and the Konqueror sections of Safari). Apple view their relationship to the Open Source community as important, and as growing even more so.

The Xserve has also had an impact on Apple's mindset. It is no longer enough to say "it has a GUI" when talking about an interface to interact with the core system; a GUI is useless if you don't have a display connected to the machine (or even a way to connect a monitor to the machine). As a result, all of OS X has command line equivalent controls of the GUI tools.

Considering the future of OS X, Jordan commented on some of the innovations that past Unix Operating Systems (e.g. Sprite, DomainOS) have had, and whether some of their innovations could be revived to make work easier and better. There is a requirement for "Operating System Archaeology" as well as new innovation – there are a great many good ideas which people have already attempted which may be ripe for reexamination. The Open Source community, especially, are doing significant amounts of reimplementation, but original, new, innovative ideas are only happening at the periphery.

Robert Watson of MacAfee Research rounded off the day with a detailed talk on the implementation of the Mandatory Access Control framework for FreeBSD and the port to OS X, covering the differences and similarities between the two. Mac OS X poses some interesting challenges due to the exposure of the kernel Mach IPC systems to user space, and how these can be used by software to access parts of another.

While creation of policy is difficult (SELinux ships with a 600 line default policy), the MAC framework is just that – a framework on which arbitrary policies can be implemented – rather than than a fully complete policy for use in a specific environment. One outcome of the work that was mentioned was the appreciation customers have of being able to have applications such as Microsoft Office running on the same systems as that on which they require the features of Mandatory Access Control.

All in all, it was a successful event with a variety of interesting talks on multifaceted aspects of the theme. Slides from most speakers are available from the UKUUG website at `http://www.ukuug.org/events/apple04/`

UKUUG Logging Tutorial: report

Aaron Wilson

UKUUG Logging Tutorial

The Marlborough Hotel, London

14th November 2004

An intimate group of us gathered on a rainy day in London for a one-day tutorial on Building a Logging Infrastructure, given by Tina Bird. Although the size of the group was small this did allow for a more informal gathering with highly relevant discussion throughout the day - as we did not necessarily follow Tina's slides exactly to plan.

On the whole I found this a good presentation of ideas on a subject which I find interesting. The ideas were clearly introduced and explained well and we got through the tutorial in a timely manner, mostly without feeling rushed.

Tina gave a good overview of logging on all kinds of systems - including routers (which she persisted in pronouncing incorrectly, being from over the pond), firewalls, various different flavours of UNIX, and of course MS Windows. For me, we dwelt too long (i.e any time at all) on the topic of Windows logging, but there where people at the tutorial to whom this was useful. It also appears that Microsoft have stolen some good ideas to use within the Event Log (one example being the use of a unique identifier for each event - something introduced into IBM Mainframes long ago).

The day, though useful did not introduce much that was ground breaking, and didn't go into too much detail on the multitude of different software products (both open source and commercial) that are out there. I would have liked to have spent more time going over the log analysis side of the subject than we did. A lot of what was said was common sense, but often these are the things that tend to get overlooked and so this was helpful.

The hotel used for the venue was easy to find and well set out; the lunch provided (in a hotel across the road) was enjoyable - although they did seem to run out of the good (and probably extremely unhealthy) looking chocolate desert before I got there.

Announcement: Open Source Skills Framework

The Open Skills Initiative is a joint project between Open Forum Europe and The Institute for IT Training, together with other interested parties. The project has produced an outline skills framework for commonly used Open Source technologies.

The framework should allow individuals to self-assess their overall skills profile, and allow potential users of those skills to know what is important. It should also give a perspective on the curriculum offered by various training providers. The first draft of the framework can be seen on the Open Forum Europe website
`http://www.openforumeurope.org`

Members of UKUUG have been invited to comment on the suitability and content of the framework: it's currently in draft stage and will almost certainly evolve further. For those interested, now is your chance to affect that evolution.

The official announcement is given below:

Open Skills Initiative

The OSCoP Competency Framework

"When you don't know where you're going any road will get you there"!

The OSCoP Competency Framework is a definition of the skills needed by IT Professionals to excel within the open source environment. It has been agreed, and is supported, by many of the major open source product providers (including HP, IBM, Novell, Sun, RedHat) and endorsed by independent organisations such as the LPI.

There are a number of key benefits to IT Professionals when they create their skills record against the Competency Framework, including:

- A simple process to endorse an IT Professional's current level of skills;
- A comprehensive process to validate those skills;
- The ability to target skills development plans to a precise and highly granular level;
- The production of a continuous professional development (CPD) record based on ability.

In addition the records are of substantial benefit to the IT Professionals' managers; these benefits include:

- A skills assessment tool that allows a manager to benchmark the skills of their team against accepted standards of excellence;

- A valuable collateral in determining personal development plans, and career planning;
- A powerful tool for validating staff competences prior to staff assignment to a new project;
- A methodology by which system integrators and external service providers can differentiate themselves in the provision of open source solutions to clients;
- A metrics-driven approach by which organisations can specify SLAs for skills within outsourced services contracts.

For a Competency Framework to be effective it should be:

- Consensual – the OSCoP Competency Framework has been agreed and is supported by most of the major organisations within the Open Source environment;
- Configurable – the framework comprises 13 specialisations, each of which can be included or excluded within the individual's personal framework;
- Role-oriented – the framework recognises 5 job roles from management to support.
- Metrics-driven – the OSCoP SkillsTracker is points driven and recognises three types of points (competence, ability, and experience) within each specialisation and role.
- Easy to use and maintain – to build a skills record within (for example) three specialisations will take approximately ten minutes.
- Easy to change – the competency framework is maintained by the Open Skills Council and will be updated regularly as technology (and the skills to use that technology) evolve.

We welcome your comments and suggestions on both the concepts behind and the detail within the OSCoP Competency Framework.

Linux in a High School - Case Study

Mike Banahan

Orwell High School in Felixstowe is a school with some 1,000 students ranging in age from 11 to 18. The school has recently received "Specialist School for Technology" status through a Government initiative.

Funding is never easy for schools in the UK public sector and John Osborne, the Deputy Head of the School responsible for for the Specialist School initiative, found himself faced with a difficult situation in early 2004. Funding for hardware was very limited and he couldn't contemplate upgrading to Windows XP since he would have to replace some fifty or so PCs with higher-end models just to run the software. A capital cost in the region of £ 25,000 was well outside the budget and when he took into account a software licensing spend in the region of £ 13,000 per year, John became convinced that he had to find a better way of using the school's resources.

When John contacted Andy Trevor of Ipswich-based IT providers Total Solution Computing Limited (www.totalsolution.co.uk) to discuss his cabling and server requirements, an idea arose. Andy had recently been working closely with Mike Banahan of GBdirect (and now UKUUG), discussing Open Source implementations in education. Orwell High School looked like a first-class candidate for a predominantly Linux and Open Source based deployment.

The school required four principal ICT classrooms with approximately 30 workstations in each one, distributed printing services and support for a number of smaller clusters of one to five workstations. All staff at the school have laptops, and the school wanted to link these to the network wirelessly.

The school had specific software requirements for the teaching environment, nearly all of which are met and exceeded by standard Open Source software packages such as OpenOffice.org, MySQL and The Gimp. These have a huge advantage over their proprietary counterparts because the students can also run them at home on their PCs without needing to worry about software licensing. Running free software on top of Windows is a useful Trojan horse in the battle to spread knowledge about the software's capabilities.

Some proprietary teaching packages have no direct equivalent in the Open Source world at the moment and some of the teaching packs in use were based on Microsoft software so support for this legacy software was also important.

Total Solution, assisted by GBdirect, proposed a low-cost solution that fully met the objectives of Orwell High School at a fraction of the cost of the Windows-based proprietary equivalent. The solution has Linux at its core (currently SuSE Linux 9.1) with a desktop based on KDE kiosk-ised to reduce administrative complexity and cost.

A crucial component of the Linux-based solution was a switch to thin-client workstations accessing software running on two central application servers. This allowed all of the existing PC hardware to be re-used without any upgrades. When the PCs boot they no longer use local hard drives, but download copies of the Linux Terminal Server (www.ltsp.org) software. Running that software, they become clients for the application servers. Instead of spending significant amounts of money on upgrading the hardware, this has prolonged the life of the workstations by several years at least (and as a consequence also reduces the load on the local landfill site). Since the workstations no longer need hard drives, their power consumption and their noise output is noticeably reduced. As discussed later, the thin-client model also slashes administration effort.

Students can log-in to any application server from any workstation when they sign-on, usually picking the most lightly loaded one. All their files are available from any server; if one server needs to be taken down for maintenance the full load of all 120 simultaneous users can still be supported on the remaining server. Each workstation provides local support for sound (an important operational requirement).

The Linux-based desktop uses a range of standard applications, amongst them Star Office which provides word processing, a presentation package and a spreadsheet; all of them are able to to save and import files in their native XML format whilst retaining compatibility with Microsoft formats. NVU is used as the HTML editor, the KDE education package provides an assortment of educational software components, Scribus is the desktop publishing package and The Gimp is an excellent image manipulation tool with a wide range of capabilities.

Whilst between them those meet by far the greatest part of the needs of the students, there is inevitably also a need for access to software that will only run in a Microsoft environment. To provide access to those legacy applications, a server running Microsoft Terminal Server 2003 is used, with the students using a RDP client from the Linux desktop. The students' files are accessible in both environments because they are stored on a single network storage server rather than locally to the Windows or Linux application servers.

Every student has a personal quota for file space and printer usage. Their personal FTP space is accessible both inside and outside the school and is used to share their files between home and school. There is additional shared FTP space administered by staff, used for setting assignments and sharing background documents. Email is provided to students and staff through Squirrelmail which gives a web interface very similar to Hotmail or Yahoo mail, this too is visible from home as well as from school. The shared-calendar features of Squirrelmail are also proving popular.

Web and email content filtering and caching is provided by an Equiinet CachePilot proxy server and firewall which front-ends the application and terminal server devices. A Quantum Snap storage server with tape backup is shared between all the application servers, rendering the choice of server transparent to the students.

The main servers are IBM blades in a modest configuration consisting of three twin-Xeon application servers with 4GB RAM each, one server used for FTP, Web and email, a further server for DHCP and LTSP boot services and a legacy Windows 2003 Terminal Server system which will be retired when the move to Open Source packages has been completed. Despite the school's plans to develop a Managed Learning environment over the next year and increase the number of PCs on the network, they will not be considering upgrading their server capacity for some time to come. In addition the space saving layout has provided a significant improvement to the server room.

Overall, the project has been a resounding success. Deputy Head John Osborne said: "I can't believe how easy it has been to move to Linux. The systems were installed and working within a week and it has been a revelation how simple and painless the process has been. I have saved thousands of pounds per year and got a brand-new ICT infrastructure at the same time". He added: "Without switching to Linux, I would have been forced to cut back on our ICT hardware and software provision. There simply wasn't the budget to upgrade to the latest versions of the software nor to keep replacing suites of PCs on a three or four year cycle. Now I have no licensing costs to worry about for the Open Source parts of the solution. We shall be moving to a complete Open Source basis as quickly as is practical and hope to start working with other schools interested in this type of development to share ideas and best practise".

The students have taken to the new system without any difficulty whatsoever. They much prefer it to the Windows systems they had been using before, commenting particularly on the reliability of the system and one observing that he was astonished to discover, having accidentally switched off his workstation before logging out, that KDE's session-restore facility returned him back to where he had been previously when he logged in again.

The administration overhead of the previous Windows-based classrooms had kept the school's ICT technician working twelve hours a day. The new system has greatly reduced this workload. John Osborne said: "The significant amount of additional work that will arise as a result of our new status would have made his job impossible had we remained with our Windows based network, and we would have been looking to increase our technician staffing to cope. This would have been another significant ongoing cost which we now feel we can avoid. This funding can now be better spent on developing materials for the staff and students to use rather than on keeping the network running." He believes a single technician could now administer something between three and five separate schools if they all used systems like those in place at the school.

From a technical perspective a number of issues were discovered. Printing proved to be problematic, especially from OpenOffice.org. Switching to Star Office, for which the school had a site licence, noticeably reduced the size of the print files and alleviated the problem. Using alternative printer definition files to reduce the resolution of the printers had a further and decisive effect, eliminating that source of concern. Printer management and quotas remain something of a problem but are being worked on. OpenOffice.org also proved to be a severe memory hog and for a while caused the application servers to swap heavily. Upgrading the memory beyond 4GB was difficult due to the cost of large memory boards for the IBM blade servers and their limited number of memory slots. Fortunately the switch to Star Office also solved that problem, since it appears to be much more frugal.

Schools appear to be excellent sites for desktop rollouts of this nature. The kids adapt quickly because they don't have years of investment in learning something different. The staff had two major gripes. The first was that their lesson plans had to change to suit the different software in use. Ideally there would be a set of free lesson plans available for schools who choose this route but at present they have not been published. The second gripe was to do with the clip-art that some students had come to rely on. Once again Star Office helped with that (better clip art) and then finding openclipart.org fixed it completely.

It's hoped that the school will become a template for many similar installations all over the country.

For further information, please contact Andy Trevor of Total Solution on 01473 384864, or Mike Banahan of GBdirect on 0870 200 7273 (+44 113 314 5740). John Osborne can initially be contacted by email at john.osborne@orwellhs.suffolk.sch.uk. The school website will also be used to keep interested parties up to date with developments.

Programming from the Ground Up

Jonathan Bartlett

Bartlett Publishing

ISBN 0-975-28384-7
332 pp.
£ 19.95

reviewed by Owen Le Blanc

Jonathan Bartlett sets a quite specific, explicit, and carefully limited goal in this book: he wants to write an introduction to programming, a book that students will use as their first programming book. The programming language he wishes to use is assembler language, indeed $i^3$86 assembler language under Linux. He writes:

the point of the book is to help the student understand how assembly language and computer programming works [sic], not to be a reference to the subject.

A reader who has finished working through the book should understand how programs work, learn new programming languages quickly, and have in general a good basis for going further in computer science.

It would be dishonest to review this book without admitting openly that I disagree with the author's views: I don't think assembler should be a student's first programming language, and despite the easy availability of machines having the $i^3$86 architecture, I think the inherited backwards-compatibility of those processors make their assembler languages more difficult to learn than, most other processors. Considerations of this kind may colour what I have to say.

The book is released under the GNU Free Documentation license, and this published version is its seventh version, according to the included documentation history. The back of the title page includes full instructions for downloading a copy from
`http://savannah.nongnu.org/projects/pgubook/`

I gather that the first 5 versions were available only online, and that printed copies became available only in January 2004.

The text contains an introduction, 12 further chapters, and an index. The introduction contains, among other advice, the recommendation to use Knoppix if you don't have a Linux of your own and can't get an account under Linux from some ISP. The author assumes that the student has already learned to use some text editor.

The chapters cover computer architecture, writing your first program, (including compiling and running it), functions, files, input and output, and optimisation. Other chapters discuss robust programming, writing and using shared libraries, memory, number representations (decimal, binary, octal, and hexadecimal), and high level languages. The last chapter gives a considered list of books which a student may wish to study after completing this one.

The book's appendices cover GUI programming (using Gnome libraries), x86 instructions, important system calls, debugging with GDB, and an index.

Perhaps because the book aims to reach an audience of beginners, it is written in an unusually clear and careful style. Each chapter begins with a little summary, which states the goals towards which the chapter aims. If necessary, it explains why they student should read and study that chapter. Each chapter ends with a review, with sections titled 'Know the Concepts' (a list of questions the student should be able to answer), 'Use the Concepts' (a list of exercises), and 'Going Further' (topics for research). The chapters vary slightly in length, but average about 20 pages each.

I have tried quite a number of the example programs and code fragments, and I have not found anything that doesn't work. I think the author develops his theme with great care, carrying the reader through one topic to the next. I am impressed, despite my scepticism. Nevertheless I have some concerns. The editing (attributed to Dominick Bruno, Jr.) leaves much to be desired; for example:

Modern computer architecture is based off of an architecture called the Von Neumann architecture, named after its creator.

(P.7; awkward phrasing, the word 'architecture' three times in a short sentence, and the unidiomatic 'based off of'.)

Local variables are data storage that a function uses while processing that is thrown away when it returns... Static variables are data storage that a function uses while processing that is not thrown away afterwards, but is reused for every time the function's code is activated... Global variables are data storage that a function uses for processing which are managed outside the function.

(P.51; unclear word order and ambiguity about whether 'storage' is singular or plural.)

Rewrite the programs in this chapter to use command-line arguments to specify the filesnames.

(P.115; the text hasn't run through a spelling checker.)

Overlooking these minor irritants – and there are, to be fair, not very many of them – I could happily recommend this book for use as an introduction to assembler language as a second programming language. Despite my disagreements with the author, I found this an impressive book. Perhaps its open GPLD license will lead to a swift correction of its remaining weaknesses.

An introduction to Programming in Emacs Lisp

Robert J Chassell

GNU Press

ISBN 1-882114-56-6
289 pp.
£ 19.55

reviewed by John Collins

Users of the Emacs editor are probably aware that its functionality, with all the operating "modes" of the editor are constructed from an internal programming language called Emacs Lisp or "Elisp". Emacs is basically an interpreter for this language, which is based upon ordinary Lisp, with built-in functions for manipulating files and windows and for responding to key presses and mouse clicks.

If you want to write your own "modes" for handling documents you commonly deal with or for writing functions to process data in particular ways, you will need to get to grips with Emacs Lisp. At a risk of being heretical here, and as much as I can't live without Emacs, I think this is a mistake. Lisp is a “functional” language, with things like recursion and functions that generate other functions second nature, as opposed to “procedural” with a list of jobs to do like the contents of a C or Perl function, yet what you do with editors 99% of the time is inherently procedural - something that Lisp handles only as an afterthought with “progn” blocks and the like.

Nowhere is this more obvious than with keyboard macros, which are stored as a sequence of keystrokes (which might differ in effect in different modes) rather than the functions they invoke. Even MS Word “Record Macro” does better than this if you can live with Basic. I can’t see that many people will want to get deep into a language whose only function is to extend an editor. If I want to do systematic changes to sets of files, I’ll reach for Perl or shell scripts.

Lisp is not an easy-to-read language. Everything is expressed as deeply nested lists, with parentheses used to set out the lists. Constructs with 10 or more consecutive closing parentheses are far from uncommon. You will definitely need an editor which tracks parentheses to write it (of course Emacs does this well).

All that said, I think this book handles the subject carefully and well, with lots of examples and exercises for you to try. To work with this book you will need to be able to try out examples on Emacs itself. Be careful though, some distributions make the default mode something other than “lisp interaction”, which is the default for “vanilla” emacs.

The book takes you from the beginning, through the basics of Lisp and at an early stage you can try out examples whilst functions, interactive functions and commands, conditionals and so forth are explained. The book covers in turn all the aspects of the editing system, regular expressions, loops, customisation and debugging, all with plenty of examples.

I am sure this will enable the reader to feel confident in extending Emacs with this book. Alas that is likely to be all he can do with his new-found knowledge. But if you want to do it, this is a good book to learn from.

Using GCC

Richard M Stallman and the GCC Developer Community

GNU Press

ISBN 1-882114-39-6
427 pp.
£ 25.14

reviewed by John Collins

GCC stands for “GNU Compiler Collection” although it used to stand for “GNU C Compiler”. The collection currently consists of compilers for C, C++, Objective-C, Java, Fortran and Ada. The compilers share common back-ends and it is possible to mix supported languages in a program compiled with compilers from the collection.

The compilers compile for and run on a wide variety of platforms and operating systems. It is quite easy to develop cross-compilers (although a working set of cross-assemblers and cross-linkers is required as well) and I personally use cross-compilers built from it almost daily.

Current language standards are adhered to, together with various extensions; however as always it is unwise to use these if your program is to be portable to systems not so blessed with GNU tools.

This book is very little more than a compilation of all the manual pages and other documentation found with the GCC source kit, so it starts off with all the options to the compiler to warn about this, optimise that and enable the other, with variations for the different languages and variations for different architectures and operating systems.

Then follow implementation dependent material such as the order of bitfields, extensions to the languages, useful utilities and problem reporting, some comments by Richard Stallman (such as the need to refer to Linux as “GNU/Linux”) and a not very comprehensive index.

This book will have little new to offer to readers who have met the manual pages relevant to them either on Linux or, especially, all the manual pages, comments and README files in the GCC sources. It is well-written, interesting and clear as far as it goes but I would wish, for instance, that there were sections on porting GCC to new operating systems and developing cross-compilers, not to mention some discussion of the internals of the system.

GNU Make

Richard M Stallman, Roland McGrath and Paul D Smith

GNU Press

ISBN 1-882114-83-3
183 pp.
£ 13.97

reviewed by John Collins

GNU Make is the GNU version of the “make” program originally offered by Version 7 (or so) Unix with many variants since. Make is a program to build a target, usually a program executable, from a series of dependencies, rebuilding as needed only the parts that have changed and things dependent upon them. A “Makefile” defines which parts depend upon which other parts.

GNU Make has all sorts of potentially useful extensions, such as conditional constructs, pattern matching and substitution etc. I say “potentially useful” because of course, like extensions to standard languages, those who want their software to be portable to other systems without GNU Make cannot safely use them. (Although in my experience GNU Make is more reliable than standard “makes” on many systems, so I usually port it anyhow).

The book is modelled closely on the “Info” documentation for GNU make. In most places the text, and the order in which it appears, are identical. Headings are altered slightly in a few places.

This is a good manual, but if you have access to a recent “Info” online description of it, and can find your way around it, you probably won’t need to buy this book as well. One nit I might pick with both this and the online documentation is that it doesn’t remember to warn you all the time where extensions to the “standard” make are being described.

Linux Unwired

Roger Weeks, Edd Dunbill and Brian Jepson

O’Reilly and Associates

ISBN 0-596-00583-0
297 pp.
£ 17.50

reviewed by John Collins

This book describes itself as “A complete guide to Wireless Configuration”.

It kicks off with a little of the physics of radio networks and the advantages and drawbacks of the various technologies from that point of view.

There then follow five chapters about Wi-Fi and one each on Bluetooth, Infrared, Cellular networks and GPS. The majority of the information about configuring the kernel, loading modules and installing utilities are to be found in the Wi-Fi chapters with knowledge assumed later on.

Kernel configuration is for up to 2.4 series kernels, configuration of 2.6 being slightly different but probably came too late for publication. Recent versions of Linux with 2.4 kernels from all the major suppliers are described and their compatibility discussed. The book also covers which hardware suppliers are more or less co-operative about supplying Linux drivers or information to enable others to do so. There is a chapter on security, on configuring access points and on building your own access points.

The book concentrates more on manufacturers and specific bits of hardware the authors have tried rather than producing a comprehensive list. There are photographs of various bits of hardware throughout. Various utilities for configuring the hardware are described.

The Bluetooth description is a little more skimpy, relying largely on the reader having read and digested the Wi-Fi sections. The Infrared section is shorter still, with the emphasis on talking to Palm Pilots and Pocket PCs.

The cell phone section is exclusively addressed to the US market and most of the phones, frequencies used and other hardware described and pictured, although similar to UK and European ones, are somewhat different. The networks are completely different. Talking to the devices is much more basic involving typing the appropriate “runes” using Kermit. You have to edit PPP chat scripts by hand.

The GPS section is brief. it explains how to interpret the strings generated by the various devices.

I think this book is an outline guide rather than a reference manual. Some readers may get irritated by statements like “we tried X hardware on machine Y running Z and it worked” if they’re trying to get help running different combinations. The later chapters assume that you’ve read and inwardly digested the earlier ones and the cross-referencing is poor, as is the index.

It would have been nice to have put up a Perl script or similar to manipulate the obscure data files in the later chapters, such as the GPS device output.

Parts, particularly the section on cell phones, need to be revised to be useful outside the USA.

I would have liked an appendix of a comprehensive list of hardware with comments about ease or otherwise of configuration.

I thought that the book made a great start, but tapered off a bit in the later chapters.

I would suggest this book as an introduction to the concepts, but not as a reference. As I have said, there needs to be a UK/European edition, particularly in regard to cell phones.

IRC Hacks

Paul Mutton

O’Reilly and Associates

ISBN 0-596-00687-X
432 pp.
£ 17.50

reviewed by Mike Smith

So the problem with the Internet is that you never know how to pronounce something (apart from Linus’s famous audio file, perhaps). Is it eye-arr-see, or like the work Irk. (Similar debates go on about vee-eye and oh-ess-ten.) I say the former (eye-arr-see) and couldn’t see any hints in this book to the contrary, but I understand regular IRC users (I’m not) refer to it as Irk.

IRC is one of the great Internet applications: A protocol that enables a global, scalable, realtime chat system - but you knew that already.

Chapter one tells you which clients to use for various operating systems. The obvious ones are covered - mIRC in MS world, XChat and others.

There are then a couple of chapters on the basics of IRC. Interesting stuff, but not really hacks as such. Also a few hints on configuring various clients - for instance setting colours and adding sounds to mIRC. Also a little on scripting solutions in some of the clients and pointers to some libraries of useful scripts.

Then there is some material on internals and writing clients in various scripting languages (the protocol is covered in RFC1459), and at this point it begins to get a little more interesting.

There are many chapters devoted to Bots - several examples of bot code and the author's own software, PieSpy. This, if you haven't come across it, displays a diagram of who is talking to whom, realtime. These chapters include how to write bots (providing some code), logging bots, community bots (for message passing and general usefulness), Search and Query, Fun, Announcements, Network and Channel Management.

The final chapters cover the IRC protocol, some non-conventional ways to connect to IRC (pocket PCs, phones etc) and finally a chapter on Servers and services. This talks about setting up your own IRC network.

In summary, this book has a lot about the basics of what IRC is, and some coverage of particular clients and tools. It majors on various types of bots - which are what make IRC a really interesting environment not only for chatting, but for automating information services. The Author is from the UK (at the University of Kent completing his PhD at the time of writing the book, apparently) - so that's a plus point. Manchester United even gets a mention in one of the screen dumps. I'm still making my mind up whether to recommend the book. On balance I think I will. For instance if I were inclined to investigate bot writing further, it contains a useful set of material to get me started. I quite liked it.

PDF Hacks

Sid Steward

O'Reilly and Associates

ISBN 0-596-00655-1
312 pp.
£ 17.50

reviewed by Mike Smith

Well, I didn't know you could play games with PDF - literally !!

So I think that proves why books are still good. With our friend Google you can know anything - but you still need to know what to look for (like you have to know how to spell in order to use a dictionary, but I digress). Books, because they're written by other people, can cover things you wouldn't expect - and that's where the value is.

How useful playing PDF games is, I don't know, but its one of the great tips in this Hacks book!

I'll give you the usual rundown on what's in it: 100 Tips, broken down into 7 Chapters - ranging from "Consuming" PDFs (I take that to mean reading them), Publishing (which is a wider topic than just dealing with PDFs), creating them, performing dynamic operations (like Forms processing) and scripting. There are two other chapters - one looking at collections of PDFs, and another about "Manipulating" PDFs. All-in-all quite a varied set of sections - though its not that obvious from the titles what's included in each. So I'll give you some examples.

The first tip is to use Adobe Reader (used to be Adobe Acrobat Reader) to read PDFs. That's pretty obvious! ... but it does set the seen for the opening chapter. There are many other ways of viewing PDFs - OS/X includes a facility to read them, or you can use GSView, for instance.

I like Hack number 4 - how to speed up Adobe Reader by removing any unwanted plugins. I use a little utility to do this, ar-speedup, but you can do it manually.

There's a good tip (16) about creating bookmarks directly to a specific page in a PDF (when you're viewing it online). That's clever.

The chapter on Publishing has some good stuff too - for instance I didn't know you could actually buy ISBN numbers online. Not cheap, but not outrageous either ($244 for 10 of them). I thought that would be restricted to the big publishers, but obviously not. A few tools are also mentioned for creating graphics for embedding in PDFs - like Graphviz, which is great.

There are lots and lots of ways to create PDFs. Of course there are the Adobe tools, but there are plenty of other options. I won't go into them all here (perl, PHP, printer filters ...) Life's too short.

Manipulating PDFs covers a multitude of sins. Splitting and merging is possible. Most people just consider PDFs as a single document, but these days you can attach (or embed) other documents too. pdftk keeps getting a mention - worth checking out - free software that has a lot of functions in this area, and others, in the book. Further tips demonstrate how you can generate PDFs in a variety of ways, superimpose water marks, use templates and other nifty tricks.

There are lots of other good tips - with examples of code too. Really quite a good book with lots of varied techniques ... and eventually we get to Battleships!

SpamAssassin

Alan Schwartz

O'Reilly and Associates

ISBN 0-596-00707-8
207 pp.
£ 17.50

reviewed by Mike Smith

I hate spam, you hate spam, we all hate spam. But did you see the recent /. story about Jeremy Jaynes. He sent 10m spams a day, usually made over $400k/month, and in his best month made $750k. Incredible - there's no wonder there's so much of it.

SpamAssassin analyses mails to determine whether they are spam or ham. There are many rules, based on content, headers and consulting external blacklists. Version 3 was released recently, and the book also covers this version.

The book is quite short, at 190 pages of content, so I wasn't expecting much. However, as we'll see, I think most of the important points are covered. In my usual style, I give you a rundown.

The initial chapters provide an overview of what it does and how it does it, installation instructions, testing and how to quickly integrate it into your mail flow using procmail. If you're setting up a busy mail gateway you'll want to use spamd (the daemon) rather than invoking a new instance of SpamAssassin for each message. This is covered too.

There's a chapter on the Rules, which is the core of how SpamAssassin identifies what is spam, and change the default rules, add new ones etc, saving them in a database if you want to (MySQL, PostgreSQL and others) or even LDAP. There are some good examples of writing your own rules, together with an explanation of the various types of tests.

However, although the Rules are a great heuristic tool there are two other advanced mechanisms - Autowhitelisting and Bayesian Filtering. These are covered in chapter 4 - again with examples and recommendations on how to setup the system to learn.

Then we have a few chapters on setting up SpamAssassin with various MTAs - sendmail, using Milter (and MIMEDefang) rather than a basic procmail, Postfix, qmail, Exim. In each case the chapter shows how to setup a spam-checking gateway. One advantage of these integration methods is that you can refuse spam during the SMTP session rather than having to accept it then decide later.

The final, brief, chapter covers using SpamAssassin as a proxy in a POP3 environment. I used to use saproxy, but that went commercial (SAproxy Pro) a year or two ago and the old free version doesn't seem to keep up with today's anti-spam requirements. The only other free alternative I know of is Pop3proxy (its probably one of the original ones too), which is okay but again doesn't stop 100% (out of the box, anyway). So in recent times I'm afraid I resorted to using my ISP's facilities to avoid the burden of updating and managing my anti-spam environment.

At work we've developed an integrated anti-spam and anti-virus solution we call MailGuardian. It uses SpamAssassin for the core anti-spam engine, and has a pluggable framework to support various commercial and non-commercial anti-virus solutions. Perhaps not really useful for you (we use it in big outsource contracts), but I thought I'd mention it as it is relevant. We use some tweaked rules to improve the hit rates but other than that its fairly vanilla.

So despite this book being quite short, I think it is useful for the uninitiated to gain a quick understanding of what SpamAssassin is all about and how to set it up. I found it very readable, and often got sidetracked exploring the content rather than writing this review. A good sign!

PHP Pocket Reference Second Edition

Rasmus Lerdorf

O'Reilly and Associates

ISBN 0-596-00402-8
138 pp.
£ 6.95

reviewed by Lindsay Marshall

See the combined review below.

Learning PHP 5

David Sklar

O'Reilly and Associates

ISBN 0-596-00560-1
432 pp.
£ 20.95

reviewed by Lindsay Marshall

I'm a great fan of the O'Reilly Pocket Reference series: good content at a good price. Perfect for the penurious programmer who needs a memory jog. And the PHP version is no exception. Everything you need to know, and as it says on the back "thoroughly updated to include the specifics of PHP4, the latest version of the language". Ooops. It arrives in the same packet as Learning PHP 5. Someone at O'Reilly needs to look at their scheduling policies (oh while they are at it can they bring out a decent, up-to-date CSS reference please). That being said, if you write PHP and don't have total recall of the PHP manual and need to find a parameter spec for one of the gazillion functions then this is good to have on your desk. Nothing else to say about it really: does what it says on the cover at a price you can't argue with.

I can't say the same, though, for Learning PHP 5. To me the whole point of PHP 5 is the extended and improved object model and the introduction of exception handling. Everything else is PHP 4. The first proper mention of objects and classes occurs on page 242, where you are referred to the PHP manual pages for information of classes and objects, and another O'Reilly book for coverage of the new PHP 5 features. There follows two pages of rudimentary information on objects and that's it. There is no mention at all of exception handling in the book that I can find. Personally, I find this entirely unacceptable. When I get a book called Learning PHP 5, I want some stuff about PHP 5. There are dozens of books telling me about PHP 4, I don't need another one, particularly one that advocates rather insanitary PHP programming practices. Frankly, this is a con. I really don't think it is worth saying anything more about it. Capsule review: don't buy it.

GNU Emacs Manual

Richard M Stallman

GNU Press

ISBN 1-88211-485-X
602 pp.
£ 25.14

reviewed by Roger Whittaker

This is the printed version of the GNU Emacs Manual which is available in various formats from gnu.org. If you are an Emacs user, then the chances are that you are one of the 90% who use only 10% of the features (or whatever the famous estimated figure is in the case of Emacs). You almost certainly don't regularly type `M-x psychoanalyze-pinhead` or frequently use the Mayan calendar. Some people find it annoying that features like these are even there: I find them occasional life-enhancing additions to a truly intelligent and powerful text editor.

But there are many genuinely useful features which you may not have come across: some of these can genuinely improve your work efficiency. The best way to find out about them is to leave the paper copy of this book lying around so that you can casually browse it. That being said, you need to be very selective in terms of which features and commands you decide you want to try out and learn, because the book documents so many of them. One feature I came across by casually browsing through the book is the set of commands for splitting and comparing windows which I have fond to be very useful.

The reference card at the end of the book is very useful. The idiosyncratic format of Appendix B (Emacs 20 Antinews) describing (negatively) the differeneces between Emacs versions 20 and 21 "for those users who live backwards in time" is typical of the author.

If you use Emacs regularly, it's worth having a copy of this book.

Free Software Free Society: selected essays of Richard Stallman

edited by Joshua Gay

GNU Press

ISBN 1-882-11498-1
219 pp.
£ 13.97

reviewed by Roger Whittaker

This book is a collection of essays, speeches, transcripts of meetings and other writings by Richard Stallman. It is no surprise that much of the material here is familiar, in that it mostly is taken from sources on the Internet which predate the printed version. (And of course they are all copyrighted under terms that allow redistribution.) So the book starts with his well-known description of the GNU project (chapter 1) and the GNU Manifesto (chapter 2), and ends with an appendix containing the GNU licences.

The book's title is 'Free Software, Free Society', but apart from wide and deep discussions of the effects of what are often called 'Intellectual Property Issues' (an example of a piece of terminology which he advises us to avoid), Stallman does not really engage with the question of what a Free Society would be like, or what kind of society he would like to see (though he has a lot to say about what he wants to prevent). This is probably a blessing in that I suspect a full exposition of Stallman's political outlook might well be embarrassing. That being said, it would almost certainly be more congenial than the loudly expressed political outlook of Eric Raymond (supposedly the more pragmatic and 'user-friendly' of the two).

Stallman is famously interested in the naming of things. His article 'What's in a Name' about the question of "Linux versus GNU/Linux" is a prime example of this, and the one which most people remember (and some mock). Personally I find it hard to summon up much interest in that issue on either side, though his own (and not entirely disinterested) motivations are clear. It may be that his attitude to this question has harmed his credibility in other matters.

His article 'Why "Free Software" is Better than "Open Source"' is a much more interesting example of the same thing. Many readers will have had plenty of time to arrive at a position on this controversy. His reasoning here is very well put, but unfortunately the reason he has probably lost this battle was simply the dual meaning of the word 'free' in English, which he himself admits is a problem. If an English word with the meaning of 'libre' had existed and had been chosen from the start, things might have been different.

Much more interesting in other places is his analysis of how the choice of language used by those with vested interests stifles debate and clear thinking about the real issues. I particularly like his list of "Words to Avoid" (chapter 21), and his explanation of how the phrase 'Intellectual Property' is used to create a false analogy with physical property to load the debate before it starts. He also notes the use of the term 'Creator' in the context of publishing which as he says "is used to elevate the author's moral status above that of ordinary people, to justify increased copyright power that the publishers can exercise in the name of the authors".

In the same section he discusses the word 'Piracy', saying "If you don't believe that illegal copying is just like kidnapping or murder, you might prefer not to use the word 'piracy' to describe it. [...] Some of us might even prefer to use a positive term such as 'sharing information with your neighbour'. " This is perhaps another example of Stallman's lack of political calculation: while this is nicely put, and a valid criticism of the use of language by the corporations, a more careful propagandist would have taken more care to avoid laying himself open to out-of-context accusations which can easily be transferred to the community as a whole.

There are chapters on the subjects of Software patents (chapter 16) and "trusted computing" (chapter 17), each of which describes the issue and puts the case very clearly. His dystopian story ‘The Right to Read’ (chapter 11) is a vision of a future in which his warnings are not heeded. His description of the way rights have often been taken away before people know that they actually have them (as in the case of DVDs and e-books) is important in this regard.

Elsewhere there are places where it is easy to make the accusation that Stallman’s idealism is unrealistic. “Free Software Needs Free Documentation” (chapter 9) is one instance, with its implied claim that O’Reilly and other such publishers and the authors who write for them are harmful and parasitic. It’s here that one is most tempted to use the dreaded phase “the real world”. His stated position on this seems to me implicitly to weaken his position on more important matters, and this is another example of how rather than putting forward a platform of ideas to try to gain the widest possible acceptance and further the cause as much as possible, he simply expressing his own views on every matter that concerns him. He would view that as the only possible or right thing to do, but this simply points up the oddness of the fact that in the world of Free Software (and ‘Open Source’) the leaders are people with a personality type that would be most unlikely to rise to a leadership role in any other walk of life.

Contacts

Ray Miller
Council Chairman; Events; Newsletter
Oxford
Tel: 01865 273 200
`ray.miller@ukuug.org`

Mike Banahan
Ely
`mike.banahan@ukuug.org`

James Youngman
UKUUG Treasurer
Manchester
`james.youngman@ukuug.org`

Sam Smith
Website
Manchester
`sam.smith@ukuug.org`

Alasdair Kergon
Events
Reading
`alasdair.kergon@ukuug.org`

Alain Williams
Watford
`alain.williams@ukuug.org`

Roger Whittaker
Schools; Newsletter
London
`roger.whittaker@ukuug.org`

Newsletter
`newsletter@ukuug.org`

Jane Morrison
UKUUG Secretariat
PO Box 37
Buntingford
Herts
SG9 9UQ
Tel: 01763 273 475
Fax: 01763 273 255
`office@ukuug.org`