AUUG Membership and General Correspondence

The AUUG Secretary
PO Box 366
Kensington NSW 2033
Telephone: 02 8824 9511
or 1800 625 655 (Toll−Free)
Facsimile: 02 8824 9522
Email: auug@auug.org.au

AUUG Management Committee
Email: auugexec@auug.org.au

President
David Purdue
iPlanet e−commerce solutions
The Tea House
Level 1
28 Clarendon Street
South Melbourne, Victoria, 3205
<David.Purdue@auug.org.au>

Vice−President
Michael Paddon
<Michael.Paddon@auug.org.au>

Secretary
Greg Lehey
IBM Australia
PO Box 460
Echunga, SA, 5153
<Greg.Lehey@auug.org.au>

Treasurer
Luigi Cantoni
Objective Management Pty Ltd
PO Box 51
North Perth WA 6906
<Luigi.Cantoni@auug.org.au>

Committee Members

Warren Toomey <warren.toomey@auug.org.au>
Sarah Bolderoff <sarah.bolderoff@auug.org.au>
Peter Gray <peter.gray@auug.org.au>
Conrad Parker <conrad.parker@auug.org.au>
Malcolm Caldwell <malcolm.caldwell@auug.org.au>

AUUG Business Manager
Elizabeth Carroll
PO Box 366
Kensington NSW 2033
<busmgr@auug.org.au>

Editorial

Con Zymaris
auugn@auug.org.au

A long long time ago, in an industry far far removed, we were awash in a myriad of local technical publications catering to our needs; we, the industry's technologists at−heart; those with a thirst for knowledge and a yen for the more complex computer crafts. Those days seem to be over. In short, both the status and state of the technical computer publication market in Australia is the worst that it's seemed in the 23 years that I've been reading the local IT trade rags. Witness the recent folding of Systems Developer Magazine; the industry downturn hit the mag and ***boom***, now it's gone. I've had many enjoyable conversations with Systems Developer editor Richard Chirgwin over the years and wish him all the best with his other publications; his work in this area will be sorely missed.

Another example? I've been a long−time reader of ACPTech's Australian Personal Computer magazine. Due to falling circulation figures, however, the magazine, and new editor David Flynn, have decided to veer the magazine increasingly away from its original market of technically−savvy IT enthusiasts and professionals, and more towards the domain of technology−lifestyle magazine. The substantive technical content is less, seemingly replaced by reviews of bland DVD movie releases and the latest web−cams.

The point of my musings? I believe that there is a substantial market out there for technically oriented, minimally−marketing−tinged publications; something that can both edify and amuse; something you can share with your technical friends and colleagues; something to improve our state and status as propeller−head IT professionals. I want AUUGN to be that something.

Sure, you can graze information off the Net, but most prefer to read and digest complex technical material in printed, preferrably bound form. This is what being a member of AUUG affords you; the costs of printing and distributing AUUGN are covered by your membership fees. We have plenty more content to put into AUUGN, but need more production costs covered. Thus my proposal to you, dear member. When you have finished reading this copy of AUUGN, don't store it away with the others on the shelf next to the Boba Fett & Han Solo figurines; stamp your name on it clearly, and proffer it to others within your place−of−work or ***maison d'***education. Our aim with AUUGN is to make it simultaneously a crowd−pleaser with regards to breadth of content, but also as a way to introduce (or re−introduce) AUUG (the organisation,) to those same colleagues, and seek their membership. A user group's vibrancy and strength flows from its members. Go forth young Padawan learner, with this copy in hand, and spread the ways of AUUG.

Cheers, Con

Contribution Deadlines for AUUGN in 2002

Volume 23 • Number 2 – June 2002: **May 15th, 2002**

Volume 23 • Number 3 – September 2002: August **15th, 2002**

Volume 23 • Number 1 – December 2002: **November 15th, 2002**

AUUG Incorporated gratefully acknowledges the support of its corporate sponsor:

AUUGN Editorial Committee

The AUUGN Editorial Committee can be reached by sending email to: auugn@auug.org.au

auugn@auug.org.au

Or to the following address:
AUUGN Editor
PO Box 366
Kensington NSW 2033

Editor:
Con Zymaris

Sub−Editors:
Frank Crawford

Public Relations and Marketing:
Elizabeth Carroll

AUUGN Submission Guidelines

Submission guidelines for AUUGN contributions can be obtained from the AUUG World Wide Web site at:

www.auug.org.au

Alternately, send email to the above correspondence address, requesting a copy.

AUUGN Back Issues

A variety of back issues of AUUGN are still available. For price and availability please contact the AUUG Secretariat, or write to:
AUUG Inc.
Back Issues Department
PO Box 366
Kensington NSW 2033

Conference Proceedings
A limited number of copies of the Conference Proceedings from previous AUUG Conferences are still available. Contact the AUUG Secretariat for details.

Mailing Lists

Enquiries regarding the purchase of the AUUGN mailing list should be directed to the AUUG Secretariat.

Disclaimer

Opinions expressed by the authors and reviewers are not necessarily those of AUUG Inc., its Journal, or its editorial committee.

Copyright Information

AUUGN is the journal of AUUG Inc., an organisation with the aim of promoting knowledge and understanding of Open Systems, including, but not restricted to, the UNIX® operating system, user interfaces, graphics, networking, programming and development environments and related standards.

Copyright without fee is permitted, provided that copies are made without modification, and are not made or distributed for commercial advantage.

President's Column

David Purdue, <David.Purdue@auug.org.au>

> **weakness** *n.* **1** being weak. **2** weak point. **3** (foll. by *for*) self indulgent liking (*weakness for chocolate*). − *The Pocket Oxford Dictionary*

Take a look at CERT Advisory CA−2002−03 <http://www.cert.org/advisories/CA−2002−03.html>. It reports on work done by OUSPG (an academic research group located at Oulu University in Finland that specialises in implementation level security issues and software security testing), that has identified security vulnerabilities in many implementations of the SNMPv1 protocol.

In all, 115 vendors are identified − although admittedly several of those do not ship SNMP implementations by default.

It should be noted that the vulnerabilities identified were in the implementations rather than the SNMP protocol. So over 100 vendors of products that use SNMP had the same security vulnerabilities.

What's going on here? Well, here is a guess.

One of the real strengths of the Internet has always been that for a protocol to be accepted as an Internet standard there must be a practical implementation of that protocol. This ensures that the protocol will work in real life, and that the paper and pencil thinking has not missed something obvious.

But what happens if everyone uses the reference implementation?

It is very tempting to just stick the reference implementation of a protocol in to your product − after all, it is usually open source, and so you save yourself the cost of development and, often, the cost of licensing. It is also the easiest way to ensure interoperability − you know for certain that you can interoperate with the reference implementation, so you should have a correct implementation of the protocol.

The problem here is that the developer of the reference implementation had different goals than you should as a product developer. The reference implementation is constructed the show the protocol works, but the goals of a product developer should not only be to work with the protocol, but also ensure that security and performance are maximised.

So is a weakness in the open source process that it encourages laziness? Once a piece of software is there, everyone uses it without checking it. While there is benefit in not reinventing the wheel, we should also be mindful of the principle outlined by Frederick Brooks in *The Mythical Man Month*: be prepared to throw one away. The best way to develop good software is to learn from other software.

It should be noted that the requirement for a protocol to become an Internet standard is now that there be two independently developed implementations.

State of the AUUG

There has been a lot of discussion in the AUUG Management Committee regarding where AUUG is and where we are going.

AUUG is running an increasing number of increasingly successful and profitable events.

However, we are faced with declining membership numbers.

The decline in membership means that AUUG does not have a secure financial base to operate from. We are dependant on the income from events − and this restricts the kind of activity we can embark upon. We can not run events that will not make a profit, we can not embark on any "loss−leader" activities, and we can not expand member benefits.

The AUUG Management Committee is looking at several options for the way forward, and one option under discussion is the dissolution of AUUG Inc.

We are also looking at ways to better target our constituency; namely, technical computing professionals.

If you have any opinions or ideas, now is the time to make them known, before it is too late. Please write to the Management Committee at auugexec@auug.org.au. Or, better yet, run for a position on the Management Committee − information on the upcoming AUUG election can be found in this issue of AUUGN.

/var/spool/mail/auugn

Editor: <auugn@auug.org.au>

What follows are ***none*** of the regular AUUG-related email exchanges, due to the fact that nonesuch have crossed your editor's desk in recent times! Instead, I've trolled many a site looking for the kind of mail I would hope to find hitting my intray, and also populating the auug-talk mailing list. Speaking of which, If you want to contribute to the list, mail majordomo@tip.net.au with:

```
subscribe talk Your Name <your@email.com.au>
```

From: tadejm@opencores.org
To: auugn@auug.org.au
Subject: Open Source PCI Bridge Soft Core
Date: Sun, 24 Feb 2002 08:38:48 +0100

Hello,

I thought this could be interesting to you. If you have a colleague interested in the subject, I'd like to ask you if you can pass this email to him/her.

The OpenCores organization announces the immediate availability of the open-source, free, complete 33/66MHz 32-bit PCI Bridge Soft Core solution.

PCI Bridge Complete & Tested

The PCI Bridge Soft Core is a complete, synthesizable RTL (Verilog) code that provides bridging between the PCI and a WISHBONE (System-on-Chip) bus. The complete package includes comprehensive specification and design documentation, a comprehensive verification suite, and a test application.

Test application is a 'VGA card' implemented using a Xilinx Spartan II device on a PCI development board from Insight Electronics. PCI bridge core is connected to a simple VGA controller core forming a system-on-chip and comes with a Linux frame buffer device driver.

The PCI Bridge Soft Core supports common ASIC and FPGA libraries and is highly configurable including options for Master/Target or Target-only as well as for +Host or Guest operation. You can download the PCI Bridge Soft Core from the OpenCores PCI Project Website at <http://www.opencores.org/projects/pci>.

OpenCores invites companies and universities to use our PCI Bridge Soft Core in your projects. Our main motivation and the sole reason for launching the PCI project has been and is to see the core used in many different projects. What we have in mind is a development much like that of open-source software (e.g. Linux).

About OpenCores

OpenCores is an organization whose main objective is to design, reuse, and integrate IP cores under the General Public License (GPL) helping the concept +of freely available, open-source hardware to emerge and become visible and apparent. For more information about our projects and us, please visit us at <http://www.opencores.org>.

We are also looking for organizations interested to support/sponsor OpenCores projects.

Best regards,
Tadej Markovic
OpenCores PCI Team

From: Dion Johnson <dionj@caldera.com>
To: wht@minnie.tuhs.org
Subject: Liberal license for ancient UNIX sources
Cc: dmr@bell-labs.com, ken@plan9.bell-labs.com, grog@lemis.com, John Terpstra <jht@caldera.com>, drew@caldera.com, maddog@li.org, evan@starnix.com, phatch@caldera.com, ransom@caldera.com

Dear Warren, and friends,

I'm happy to let you know that Caldera International has placed the ancient UNIX releases (V1-7 and 32V) under a "BSD-style" license. I've attached a PDF of the license letter hereto. Feel free to propogate it as you see fit.

I apologize that this has taken so long. We do not have a well regulated archive of these ancient releases, so we must depend upon you UNIX enthusiasts, historians, and original authors to help the community of interested parties figure out exactly what is available, where, and how.

Many thanks to Warren Toomey, of PUPS, and to Caldera's Bill Broderick, director of licensing services here. Both of these gentlemen were instrumental in making this happen. And thanks to our CEO, Ransom Love, whose vision for Caldera International prescribes cooperation and mutual respect for the open source communities.

Of course, there are thousands of other people who should be acknowledged. I regret I do not have time or wisdom to make a list of them all, but maybe someone does, or has.

Anyway, here it is. Feel free to write to us if you want to understand more about how/why Caldera International has released this code, or you have any other comments that we should hear.

Sincerely,
Dion L. Johnson II - dionj@caldera.com
Product Manager and one of many open source enthusiasts in Caldera Intl.

http://www.tuhs.org/archive_sites.html

Call for Papers: AUUG 2002 Theme: "Measure, Monitor, Control"

Message from the Programme Chair

Greetings one and all. My name is Adrian Close and I'll be your Programme Chair for this year's Winter Conference, to be held in Melbourne from the 1st – 6th of September. It's only March as I write, but already significant effort has been invested in bringing you a conference of the traditional calibre. Of course, I have a hard–working committee to back me up and that's just to organise speakers.

We're working hard to bring you a solid programme this year and we've made great progress thus far. Still, we definitely have room for more speakers and tutorial presenters, so if you've got something you'd like to talk about, I encourage you to have a look at the Call for Papers (http://www.auug.org.au/winter/auug2002/cfp.html) and get in touch with the Programme Committee (auug2002prog@auug.org.au).

We are seriously looking at the possibility of running a Student Day, the idea being to provide information of interest and relevance to students, in an inexpensive fashion. We envisage an expose on the inner workings of the IT industry and something of a student survival guide for those interested in joining the insanity, together with a sample of the strange clue attractor that is AUUG. The committee welcomes suggestions for this (and indeed any other) part of the programme.

As always, a successful conference needs sponsors. At this stage we'd like to acknowledge the kind support of IBM and Checkpoint. Of course, we welcome the possibility of other sponsors, so if your company would like to help out, please see http://www.auug.org.au/winter/auug2002/sponsor.html.

Finally, if you're interested in attending the conference this year, the logistics people would be especially grateful if you could register your interest via the link on the conference web site.

When: 1st – 6th September 2002 (3 days tutorials + 3 days conference)
Location: Duxton Hotel, 328 Flinders St, Melbourne, Australia Web:
http://www.auug.org.au/winter/auug2002/
Next important date: 10th May – Tutorial/Paper abstracts due.

I'm looking forward to a great conference!

Adrian Close <adrian@auug.org.au>
Programme Chair, AUUG 2002

The AUUG Annual Conference will be held in Melbourne, Australia, on 4, 5 and 6 September 2002 *(subject to change)*.

The Conference will be preceded by three days of tutorials, to be held on 1, 2 and 3 September 2002.

AUUG 2002 SPONSORS

CHECK POINT™
Software Technologies Ltd.
We Secure the Internet.

IBM®

Call for Papers: AUUG 2002 Theme: "Measure, Monitor, Control"

The Programme Committee invites proposals for papers and tutorials relating to:

- Cluster Computing
- Managing Distributed Networks
- Performance Management and Measurement
- Open Source Systems Administration Tools
- System and Application Monitoring
- Security in the Enterprise
- Technical aspects of Computing
- Networking in the Enterprise
- Business Experience and Case Studies
- Open Source projects
- Business cases for Open Source
- Technical aspects of Unix, Linux, and BSD variants
- Open Systems or other operating systems
- Computer Security
- Networking, Internet (including the World Wide Web)

Presentations may be given as tutorials, technical papers, or management studies. Technical papers are designed for those who need in−depth knowledge, whereas management studies present case studies of real−life experiences in the conference's fields of interest.

A written paper, for inclusion in the conference proceedings must accompany all presentations.

Speakers may select one of two presentation formats:

Technical presentation:

- A 30−minute talk, with 10 minutes for questions.

Management presentation:

- A 25−30 minute talk, with 10−15 minutes for questions (i.e. a total 40 minutes).

Panel sessions will also be timetabled in the conference and speakers should indicate their willingness to participate, and may like to suggest panel topics.

Tutorials, which may be of either a technical or management orientation, provide a more thorough presentation, of either a half−day or full−day duration.

Representing the largest Technical Computing event held in Australia, this conference offers an unparalleled opportunity to present your ideas and experiences to an audience with a major influence on the direction of Computing in Australia.

Call for Papers: AUUG 2002 Theme: "Measure, Monitor, Control"

Submission Guidelines:

Those proposing to submit papers should submit an extended abstract (1−3 pages) and a brief biography, and clearly indicate their preferred presentation format.

Those submitting tutorial proposals should submit an outline of the tutorial and a brief biography, and clearly indicate whether the tutorial is of half−day or full−day duration.

Speaker Incentives

Presenters of papers are afforded complimentary conference registration.

Tutorial presenters may select 25% of the profit of their session OR complimentary conference registration. Past experience suggests that a successful tutorial session of either duration can generate a reasonable return to the presenter.

Please note that with the GST changes to tax legislation we will be requiring the presentation of a tax invoice (which we will assist in producing) containing an ABN for your payment. If that is not provided then tax will have to be withheld from your payment.

Important Dates

Abstracts/Proposals Due	− 10 May 2002
Authors notified	− 7 June 2002
Final copy due	− 6 July 2002

Tutorials	− 1−3 September 2002
Conference	− 4−6 September 2002

Proposals should be sent to:

AUUG Inc.
PO Box 366
Kensington NSW 2033
AUSTRALIA

Email: **auug2002prog@auug.org.au**

Phone: 1800 625 655 or +61 2 8824 9511
Fax: +61 2 8824 9522

Please refer to the AUUG website for further information and up−to−date details:

http://www.auug.org.au

Call for Papers: AUUG 2002: Sponsorship Opportunities

Diamond Sponsorship

Cost
A$10,000 (plus 10% GST applicable)

Includes

- 2 complimentary registrations for the conference
- 2 complimentary invitations for the cocktail reception
- 2 complimentary invitations for the conference dinner
- logo displayed in conference plenary hall
- acknowledged on all appropriate occasions in both print and verbally
- small display area
- listed and identified as a sponsor in the conference brochure and final programme, with company description
- logo displayed and identified as a sponsor on the AUUG website with a link back to organisation's site

Choice of
Conference Brochure

- wide distribution to key decision makers
- areas of exclusive advertising
- immediate impact prior to the conference

Welcome Reception

- prestigious event allowing sponsor to make first impression on the delegates
- reception identified as being sponsored by the XYZ company on all printed material
- signage on the evening
- opportunity to address delegates

Conference Dinner

- dinner identified as being sponsored by the XYZ company
- name printed on dinner menu
- opportunity to distribute mementos and address to the audience
- banner identifying the sponsoring company

Platinum Sponsorship

Cost
A$7,500 (plus 10% GST applicable)

Includes

- 1 complimentary registration for the conference
- 2 complimentary invitations for the cocktail reception
- 2 complimentary invitations for the conference dinner
- logo displayed in conference plenary hall
- acknowledged on all appropriate occasions in both print and verbally
- display space
- listed and identified as a sponsor in the conference brochure and the final programme
- logo displayed and identified as a sponsor on the AUUG website with a link back to organisation's site

Choice of
Conference Proceedings

- 2 A4 pages of exclusive advertising
- long term usage and shelf life as it is a reference material

Tee–Shirts

- offering long term usage and company message to recipient

Conference Satchel

- Satchel offering long term usage and company message to recipient

Gold Sponsorship

Cost
A$5,000 (plus 10% GST applicable)

Includes

- 1 complimentary invitation for the cocktail reception
- 1 complimentary invitation for the conference dinner
- logo displayed in conference plenary hall
- acknowledged on all appropriate occasions in both print and verbally
- display space available for one day at the conference
- listed and identified as a sponsor in the conference brochure and conference final programme
- logo displayed and identified as a sponsor on the AUUG website with a link back to organisation's site

Choice of
Speakers Reception

- event allowing sponsor to make first impression with speakers
- reception identified as being sponsored by the XYZ company on all printed material
- signage on the evening
- opportunity to address speakers

Pen'n'Paper

- Company logo on pens and writing pads distributed to delegates, offering long term usage and company message to recipient

Lapel Badges

- Company logo on delegate lapel badges, offering company visibility for duration of conference

Keynote Sessions

- opportunity to introduce the keynote session

Silver Sponsorship

Cost
A$2,500 (plus 10% GST applicable)

Includes

- 1 complimentary invitation to the cocktail reception
- logo displayed in conference plenary hall
- acknowledged on all appropriate occasions in both print and verbally
- rack space for promotional material
- listed and identified as a sponsor in the conference brochure and the conference final programme
- logo displayed and identified as a sponsor on the AUUG website with a link back to organisation's site

Choice of
Conference folder insert

- individual inserts in conference satchels

Advertisement

- A4 sized advertisement in conference proceedings

Registration desk handouts

- Promotional material to be available to delegates from the Conference Registration Desk

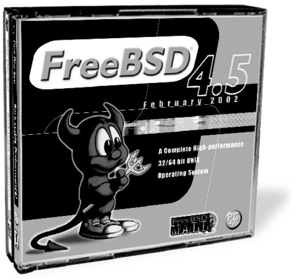
Additional Opportunities

Audio Visual *
Conference Network *
* Contact the AUUG Business Manager, for further details

¥3623 Sanford Street ¥Concord CA 94520 ¥Phone +1.925.674.0783 ¥Fax +1.925.674.0821 ¥info@freebsdmall.com

FreeBSD 4.5

FreeBSD 4.5 represents the latest release in the FreeBSD 4.x-STABLE branch. Hundreds of new features, bug fixes, and security issues have been addressed since FreeBSD 4.4.



The FreeBSD 4.5 Jewel Case CD Set contains:

¥ *Disc 1:* Installation & Packages
¥ *Disc 2:* Live File System (for system recovery), CVS Repository, and Commercial Software Demos
¥ *Disc 3:* Additional Packages
¥ *Disc 4:* More Packages

FreeBSD is a powerful, professional quality UNIX-compatible operating system. FreeBSD is based on 4.4BSD, developed by the University of California, Berkeley, and its contributors. 25 years of development has made it the world s most mature and robust network operating system.

FreeBSD is the choice of thousands of Internet Service Providers and other organizations which depend on FreeBSD s high performance and reliability. Many of the world s largest Internet sites run on FreeBSD, including Yahoo! and Microsoft s Hotmail subsidiary.

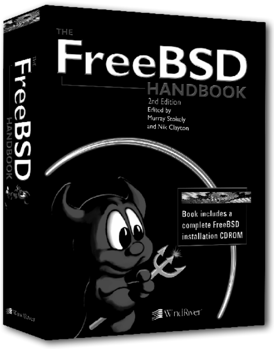
. . . . $39.95

The FreeBSD Handbook

The FreeBSD Handbook is a comprehensive FreeBSD Tutorial and reference. It covers installation, day-to-day use of FreeBSD, and much more!

This book begins with an installation walk-through followed by an introduction to the basic system components covering topics such as UNIX basics, installing applications, and the X Window System.

In addition, you will find in-depth coverage of various FreeBSD system administration topics such as system configuration and tuning, user account management, building a custom kernel, updating your system, sound and multimedia, system security, advanced networking, and Linux binary compatibility, to name a few.

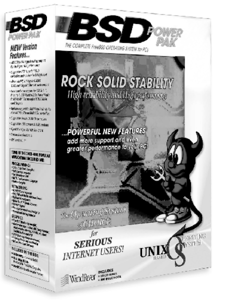

THE FreeBSD HANDBOOK
2nd Edition
Edited by Murray Stokely and Nik Clayton
Book includes a complete FreeBSD installation CDROM
WindRiver

This handbook is authored by the FreeBSD Documentation Project, a group of volunteers coordinating efforts through the Internet.

The FreeBSD Handbook comes with the full FreeBSD Operating System on a CDROM. It provides everything you need to run a complete UNIX desktop or a powerful network server!

. . . . $49.95

The FreeBSD PowerPak



The FreeBSD PowerPak contains everything needed to run a complete FreeBSD system, and more! Perfect for the FreeBSD enthusiast, or power user.

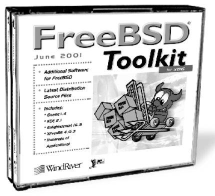
Inside the PowerPak, you'll find:

¥ FreeBSD 4-disc Installation installation & packages
¥ FreeBSD 6-disc Toolkit distfiles & more packages
¥ The Complete FreeBSD, Third Edition by Greg Lehey
¥ Installation CDs, gigabytes of installable open-source software
¥ Complete reference book.

The PowerPak will get you going in no time! See why FreeBSD carries its reputation as a mature and robust network operating system!

. . . . $99.95

The FreeBSD Toolkit



The FreeBSD Toolkit is a collection of additional material that does not fit on the official FreeBSD 4-disc set. Within the Toolkit, you'll find six CDROMs packed with open-source software:

¥ *Discs 1 & 2:* Additional Packages
¥ *Discs 3-6:* Ports distfiles

If you dislike compiling software from ports or source, the packages are for you. If you have slow or no Internet access, the distfiles offer you gigabytes of source code distributions you would otherwise have to download.

The Toolkit makes a perfect companion for any FreeBSD system!

. . . . $39.95

Please note: all prices are in US$]

Promo Items

FreeBSD Classic and Power To Serve T-Shirt
These Hanes Beefy Tees are 100% cotton and are the perfect low-cost way to proclaim your love for FreeBSD in public! Please specify size and Classic or Power To Serve when ordering. **M, L, XL, and XXL** $16.00

FreeBSD MicroFiber Jacket
Tan Munsingwear MicroFiber Jackets with the BSD Daemon logo embroidered on chest. Warm, yet lightweight. **M, L, XL, and XXL** $80.00

FreeBSD Denim Shirt
Long-sleeve, 100% cotton, button-down collar and the BSD Daemon embroidered above the left pocket. **M, L, XL, and XXL** $45.00

FreeBSD Polo Shirt
Land s End, 100% cotton polos with the BSD Daemon embroidered on the left. Available in white, grey or black. **M, L, XL, and XXL** $45.00

FreeBSD Sweatshirts M, L, XL, and XXL	$30.00
FreeBSD Tan or Black Baseball Cap	$20.00
FreeBSD Boxer Shorts M, L, XL, and XXL	$16.00
FreeBSD Daemon Horns	$1.00
FreeBSD Neoprene CDROM Case (holds 6 CDs)	$12.00
FreeBSD Mouse Pad	$10.00
FreeBSD Stainless Steel Travel Mug	$20.00
FreeBSD Plush Daemon	$12.00
FreeBSD Beanie Daemon	$6.00
FreeBSD Bumper Sticker	$2.00
FreeBSD Lapel Pin	$5.00
FreeBSD Button	$1.00
FreeBSD Case Plate	$1.00
FreeBSD Stickers	$0.50

FreeBSD Power To Serve T-Shirt

FreeBSD Classic T-Shirt

FreeBSD Case Plate

FreeBSD Jacket

FreeBSD Button

FreeBSD Plush Daemon

FreeBSD Caps

FreeBSD Beanie Daemon

Stainless Steel FreeBSD Mug

FreeBSD Mousepad

FreeBSD CDROM Case

Name ______________________

Address ______________________

City, St, Zip ______________________

Country ______________________

Email ______________________

Phone ______________________

Use this form and mail to:

FreeBSD Mall
3623 Sanford St.
Concord CA 94520

or go to our website at:
www.freebsdmall.com

QUANTITY	PRODUCT NAME	PRICE	TOTAL

Payment Method:

○ VISA ○ American Express

○ Discover ○ Mastercard

○ Check/Money Order

Card Number ______________ Exp ________
Name on Card ______________________
Signature ______________________

z *Shipping Charge*
$5.00 US/Canada/Mexico
$9.00 Overseas

SUB TOTAL	
TAX: CA only, 8.25%	
SHIPPING	
TOTAL	

Annual Election of Officers and General Committee Members: Call For Nominations

GET INVOLVED!

AUUG has a proud 27 year history of sharing knowledge, providing member services and, most importantly, creating a community of like minded professionals. Every year brings fresh challenges and new opportunities. As a result, AUUG is in a constant process of evolution; a process of which every member in our association is a part. This year will mark a particularly interesting chapter in AUUG's evolution: for the first time in nearly ten years, we will reevaluate our position in the industry. We expect significant changes as a result.

The role of AUUG's Officers and General Committee Members is to manage, plan and execute, according to the will of the general membership. This stewardship is not passive, nor is it always easy. However, serving the AUUG community is also immensely rewarding because, simply, our goals matter and we can make a difference.

What should AUUG be doing next year? How can we serve our members and our community better? What great ideas are out there, just waiting for their chance to be tried out? How do we better promote our knowledge and philosophies? Do you know the answers to some of these questions? Are you the sort of person who knows how to get things done? Or do you know someone like this? AUUG needs people with fire and clue. Help make AUUG the kind of association you want it to be--nominate the best people for election to our Management Committee.

If you would like to know more about serving on the Management Committee, email the current committee at auugexec@auug.org.au. In order to stand for office, you must be an Individual Member of the AUUG, and you need to be nominated by three voting members of AUUG (that is, either Individual Members or Institutional Members). If you can't find three people to nominate you, send in your nomination form anyway. We should be able to find someone to sign it.

In order to nominate a member for the Committee, please copy and fill out the following official nomination form, and send it to the AUUG Secretary. All nominations must be received by 14 April 2002. You can send in nominations by fax or (snail) mail: Fax: (02) 8824 9522 Mail:

AUUG Inc.
PO Box 366
Kensington NSW 2033
Australia

We encourage nominees to include a policy statement of up to two hundred words. This statement will be circulated to members with election materials, and is intended to assist them in making voting decisions. The Secretary reserves the right to truncate lengthy statements in order to minimise election expenses.

AUUG Inc. 2002 Annual Election: Nomination Form

We,

(1) Name: AUUG Member #: and

____

(2) Name: AUUG Member #: and

____

(3) Name: AUUG Member #:

____

being current financial members of AUUG Inc do hereby nominate:

____

for the following position(s):
Mark the boxes against the positions for which nomination is desired. Each person may be elected to at most one position, and election shall be determined in the order shown on this nomination form.

1. President
2. Vice President
3. Secretary
4. Treasurer
5. Ordinary Management Committee Member (5 positions)
6. Returning Officer
7. Assistant Returning Officer

Signed (1) ________________________ Date: _____________

Signed (2) ________________________ Date: _____________

Signed (3) ________________________ Date: _____________

I (name):_______________ AUUG Member #:___________
do hereby consent to my nomination to the above position(s), and declare that I am currently a financial Individual Member of AUUG Inc.

Signed:___________ Date:___________

Public Notices

Upcoming Conferences & Events

SANE 2002
The System Administration and Networking Conference
May 27–31
The Netherlands

USENIX '02
USENIX Annual Technical Conference
June 10–15
Monterey, CA

JVM '02
2nd Java™ Virtual Machine Research and Technology Symposium
August 1–2
San Francisco, CA

Security '02
11th USENIX Security Symposium
August 5–9
San Francisco, CA

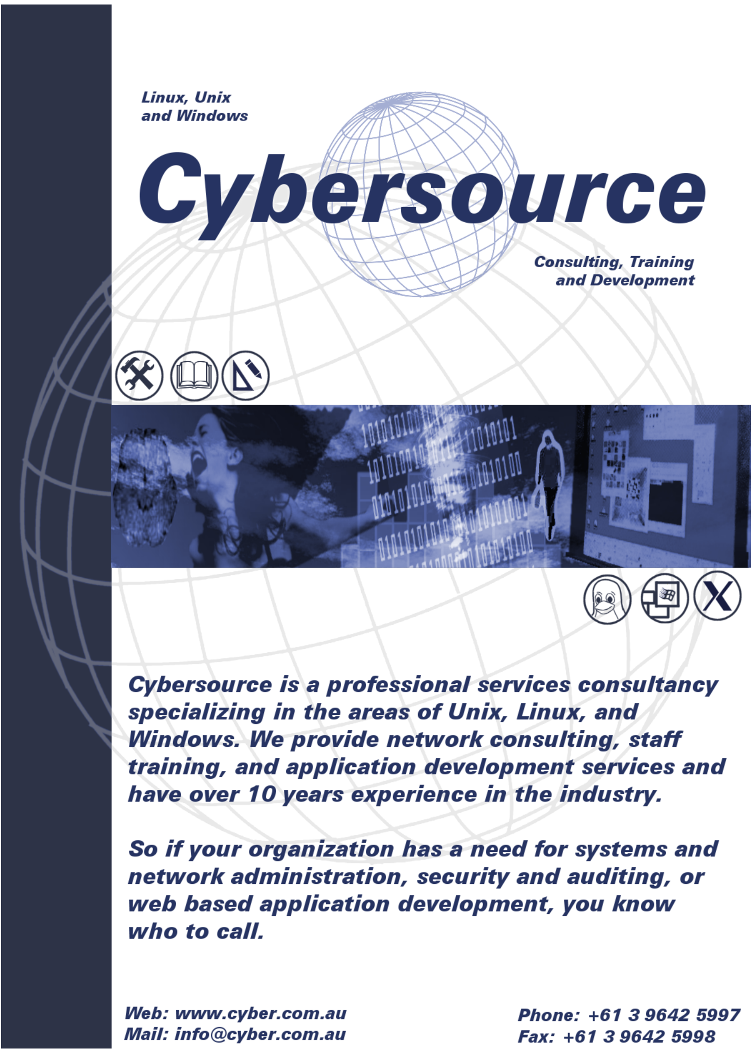
SAGE-AU 2002 Conference
Melbourne, August 5 – 9, 2002

AUUG'2002 Annual Conference
Melbourne, September 4th – 6th

LISA '02
16th Systems Administration Conference
November 3–8
Philadelphia, PA

IMW 2002
Internet Measurement Workshop 2002
November 6–8
France

OSDI '02
5th Symposium on Operating Systems Design and Implementation
December 9–11
Boston, MA



My Home Network (March 2002)

By: Frank Crawford <frank@crawford.emu.id.au>

Well it is a new year, but unfortunately the same old problems exist, viruses, hackers and other nasties. In the last issue I talked about the basic building blocks for virus scanning and protection, uvscan and automatic updates, along with regular scans of the disk. However, while catching viruses after they hit your disk is useful, it is a bit late. It is much better to catch it before it gets in.

While there are a couple of entry vectors, the most common, these days, are via the network and in particular mail and web pages. Hence, these are the areas I have concentrated on.

The first entry to be closed is mail, and luckily enough this is also an area which has had considerable work done for Unix. In this area there is one standard product called AMaViS – 'A Mail Virus Scanner', which can be found at http://www.amavis.org. This is probably the most widely used antivirus product on open systems, and has a number of different configurations and options, depending on the system requirements. Unfortunately, this is also one of the biggest drawbacks, as the documentation is less than spectacular and confusing.

AMaViS has been through a number of generations, and the most recent version is primarily a daemon written in Perl with a small C program to interface your mail transport agent (MTA). Supported MTAs include sendmail, qmail, postfix and exim, while the scanner can support a wide variety of MIME types. The scanner works by saving each part of the mail to disk and then running a commercial virus scanner across it. One design issue with AMaVis that increases its complexity is that it makes no assumptions about what types of files the virus scanner can handle, and extracts all files before scanning.

This added complexity is handled by an extensive use of Perl modules, which causes some initial problems during installation. The list of modules required caused me some initial dismay, especially as a number of them require other executables (e.g. 'zoo') which then also have to be installed. I found it easiest to download most of the perl modules from CPAN (or the Comprehensive Perl Archive Network – http://www.cpan.org), the various Red Hat distributions (e.g. Powertools) and from links of the AMaViS page.

Once installation of all the prerequisites was completed, the actual installation of amavisd was pretty simple. My setup consists of a fairly standard Red Hat 7.1 (now 7.2) installation with sendmail 8.11 as the MTA.
Now AMaViS has two possible configurations for sendmail, one replacing the standard local delivery agent with 'amavis', a stub which passes the mail to the 'amavisd' daemon, and the second which uses the new milter interface to forward mail directly from sendmail. The use of amavis as a local delivery agent means that only mail being delivered locally can be scanned, while the use of the milter interface allows all mail passing though sendmail, either being delivered or forwarded, to be scanned. Unfortunately, as shipped by Red Hat, the milter interface is not enabled, and would require recompilation to use it. As my use is to handle locally delivered mail, it is not a big issue and I decided to go and use 'amavis'.

Once this was decided and compilation, etc, was completed, as described in the README file, the final installation was simple and involved starting the daemon (amavisd) and modifying '/etc/sendmail.cf' to pass locally delivered mail to amavis. Red Hat's standard configuration invokes procmail to perform the final delivery and consists of a statement of the form:

```
Mlocal,         P=/usr/bin/procmail,
F=lsDFMAw5:/|@qSPfhn9,
S=EnvFromL/HdrFromL,+R=EnvToL/HdrToL,
                T=DNS/RFC822/X-Unix,
                A=procmail -Y -a $h -d $u
```

which I changed to:

```
Mlocal,         P=/usr/sbin/amavis,
F=lsDFMAw5:/|@qSPhn9, S=EnvFromL/HdrFromL,
+R=EnvToL/HdrToL,
                T=DNS/RFC822/X-Unix,
                A=amavis $f $u --
/usr/bin/procmail -Y -d $u
```

Aside from the replacement of procmail with amavis, there are two other subtle changes, the removal of the 'f' option from the flags (i.e. "F=") which would cause an additional '–f' added to the call to amavis; and the removal of the "–a $h" from the procmail arguments, which causes a problem on local delivery, as '$h' is then null (this probably indicates a problem in argument handling somewhere in the chain of programs).

The creation of an alias "virusalert" to receive messages about problem mails and a restart of sendmail completed the process. This should immediately be followed by sending some test mail containing the EICAR test virus (similar to the test performed with the original 'uvscan' installation). To do this, you can either mail the EICAR.COM file used by 'uvscan' to yourself, or make use of the basic testing facilities supplied with AMaViS. Either way, you should see a warning, and the mail won't be delivered.

To make this really a set and forget system, you will also need to set up an init script so 'amavisd' is started when the system is rebooted, otherwise all local mail delivery will stay queued until communications with the amavisd daemon can be established.

Closing the next hole, i.e. via the Web, is slightly more complicated, as it involves more system issues than AMaVis. The package I've been using is called Viralator and can be found at http://viralator.loddington.com. It consists of a CGI script to which all downloads are redirected.

The redirection is enabled through the "redirect_program" option in squid, which passes all squid requests through a separate program. The package required for this is squirm, an excellent security package in itself. The Squirm package can be found at http://www.senet.com.au/squirm/, and while the online documentation is a bit out of date, it is simple to install. The biggest difference from the online documentation is that the current version (squirm-1.23) can use the system regex library, so you can just edit the Makefile, followed by running make and "make install".

Once squirm is installed, add the following two lines:

```
redirect_program        /usr/local/squirm/bin/squirm
redirect_children       5
```

in '/etc/squid/squid.conf', and an appropriate configuration file installed in /usr/local/squid/etc. In my installation, I've made a couple of modifications to 'viralator.cgi' to simplify the setup.

I won't fully explain how squirm is configured, but firstly, I've set up the following patterns in '/usr/local/squirm/etc/viralator.patterns':

```
# Viralator setup
abortregexi ^(http|ftp|https)://192\.168\.2\.5/.*
abortregexi
^(http|ftp|https)://(bits|www|wpad|www-proxy|www-
cache)\.crawford\.emu\.id\.au/+.*
# Allow automatic download of Symantec Liveupdate (zip file)
abortregexi ^http://liveupdate\.symantecliveupdate\.com/.*
abortregexi ^http://www\.ht\.com\.au/scripts/xworks\.exe.*
regexi ^(.*\.zip)$ http://www.crawford.emu.id.au/cgi-
bin/viralator.cgi?url=|\1
regexi ^(.*\.doc)$ http://www.crawford.emu.id.au/cgi-
bin/viralator.cgi?url=|\1
regexi ^(.*\.exe)$ http://www.crawford.emu.id.au/cgi-
bin/viralator.cgi?url=|\1
regexi ^(.*\.xls)$ http://www.crawford.emu.id.au/cgi-
bin/viralator.cgi?url=|\1
regexi ^(.*\.tgz)$ http://www.crawford.emu.id.au/cgi-
bin/viralator.cgi?url=|\1  regexi ^(.*\.tar)$
http://www.crawford.emu.id.au/cgi-bin/viralator.cgi?url=|\1
regexi ^(.*\.tar\.gz)$
http://www.crawford.emu.id.au/cgi-bin/viralator.cgi?url=|\1
regexi ^(.*\.arj)$ http://www.crawford.emu.id.au/cgi-
bin/viralator.cgi?url=|\1
regexi ^(.*\.vbs)$ http://www.crawford.emu.id.au/cgi-
bin/viralator.cgi?url=|\1
regexi ^(.*\.shs)$ http://www.crawford.emu.id.au/cgi-
bin/viralator.cgi?url=|\1
```

and then configure squirm (in '/usr/local/squirm/etc/squirm.conf') with:

```
begin
network 192.168.2.0/24
pattern squirm.patterns all
pattern viralator.patterns all
end

begin
network 127.0.0.0/24
pattern squirm.patterns all
end
```

('squirm.patterns' is part of the Squirm distribution).

From these configurations you can see that URLs with the extension: zip, doc, exe, xls, tgz, tar, tar.gz, arj, vbs and shs, are passed to viralator.cgi for processing. What viralator.cgi does is download the given URL to a secure area with 'wget', runs the virus scanner over it and then forwards it to the original caller.

One of the big issues with viralator is to ensure that it doesn't get into a loop, with "wget" calling through squid, which then calls "viralator.cgi", which then calls "wget", which ...

Anyway, there are a number of ways around this impasse, one of which is described in the online documentation, another of which is to force wget to bind to the loopback interface, and then use the squirm configuration I gave earlier. To enable this, you need to apply the following patch to the latest version of viralator (viralator-09pre2):

```
@@ -568,7 +592,7 @@
 # print "<\!\-\- untainted url $fileurl
\-\-\>\n";
 # print "<\!\-\- untainted filename  $filename
\-\-\>\n";

-open(TR,"wget -c  --http-user=".$username.
open(TR,"wget -c --tries=1 --bind-
address=localhost --http-user=".$username.
      "  --http-pass=".$password.
      " \"$fileurl\" -O $downloads/\"".$filename.
      "\" 2>&1|") || die "Error";
```

There is one final item with Viralator that causes some concern, after downloading the file it returns to the original download page, however, due to the way some pages are constructed, it causes another download, and around and around. I've submitted a patch to the author (BTW a fellow Aussie) which uses the JavaScript 'history.back()' function and seems to work better. If you need it, drop me a line.

With AMaViS and Viralator, that closes two major holes, mail and web, although what it doesn't do is protect against various web tricks, as it is only as good as the virus scanner and is intended to handle viruses within files. These have proved to be effective, with AMaViS having caught a few bad mail items before they hit any local mailboxes. On the other hand, viralator hasn't yet picked up any, although that may be more due to a choice of sites than the software.

Unfortunately, that still leaves one major hole, instant messaging file downloads to be blocked, and one that doesn't seem to be even available for commercial antiviruses packages. It is an area that I can see how to address, but not something I am willing to undertake.

So, that is the current state of my protection against viruses. It offers good protection, although it can be better, and something I will continue to improve over time.

What do others suggest?

AUUG Corporate Members

as at 1st March 2002

- ac3
- Andersen Consulting
- ANSTO
- ANU
- Aust Centre for Remote Sensing
- Australian Bureau of Statistics
- Australian Industry Group
- Australian Water Technologies P/L
- British Aerospace Australia
- Bureau of Meteorology
- C.I.S.R.A.
- Cape Grim B.A.P.S.
- Central Sydney Area Health Service
- Centrelink
- CITEC
- Commonwealth Steel Company
- Computer Science, Australian Defence Force Academy
- Computing Services, Dept Premier & Cabinet
- Corinthian Industries (Holdings) Pty Ltd
- Crane Distribution Limited
- CSC Australia Pty. Ltd.
- SIRO Manufacturing Science and Technology
- Curtin University of Technology
- Cybersource Pty. Ltd.
- Deakin University
- Department of Land & Water Conservation
- Energex
- Everything Linux & Linux Help
- Fulcrum Consulting Group
- G.James Australia Pty. Ltd.
- ING
- IP Australia
- IT Services Centre, ADFA
- Land and Property Information, NSW
- LPINSW
- Macquarie University
- Multibase WebAustralis Pty Limited
- Namadgi Systems Pty Ltd
- NSW National Parks & Wildlife Service
- NSW Public Works & Services, Information Services
- Peter Harding & Associates Pty. Ltd.
- Rinbina Pty. Ltd.
- Security Mailing Services Pty Ltd
- St. John of God Health Care Inc.
- St. Vincent's Private Hospital
- Stallion Technologies Pty. Ltd.
- TAB Queensland Limited
- The University of Western Australia
- Thiess Pty Ltd · Tower Technology Pty. Ltd.
- Uniq Advances Pty Ltd
- University of Melbourne
- University of New England
- University of New South Wales
- University of Sydney
- University of Technology, Sydney
- Victoria University of Technology
- Westrail · Workcover Queensland

Implementing a Bridging Firewall

Author: David Whitmarsh
<david.whitmarsh@sparkle-consultancy.co.uk>

What is the difference between a bridging firewall and a conventional firewall? Usually a firewall also acts as a router: systems on the inside are configured to see the firewall as a gateway to the network outside, and routers outside are configured to see the firewall as the gateway to the protected network. A bridge is piece of equipment that connects two (or more) network segments together and passes packets back and forth without the rest of the network being aware of its existence. In other words, a router connects two networks together and translates between them; a bridge is like a patch cable, connecting two portions of one network together. A bridging firewall acts as a bridge but also filters the packets it passes, while remaining unseen by either side.

Why might you want to so such a thing? A couple of reasons spring to mind:

- You can plug in a firewall without changing any of your existing network software configuration.
- You may want to protect part of a network where you do not have control of the external routing into your network.

My Problem

In my office I had a shiny new ADSL connection from Demon Internet with an assigned 16 address subnet (less base, broadcast and router IP = 13 IP addresses). Because of the vagaries of the UK commercial and regulatory environment, the line and router were installed and owned by British Telecom plc. and there was no facility to configure the router to use an internal gateway. This left me two choices:

- Connect every host directly to the ADSL router and set up iptables separately for each one.
- Use a firewall with ip masquerading to present a single ip address to the outside world.

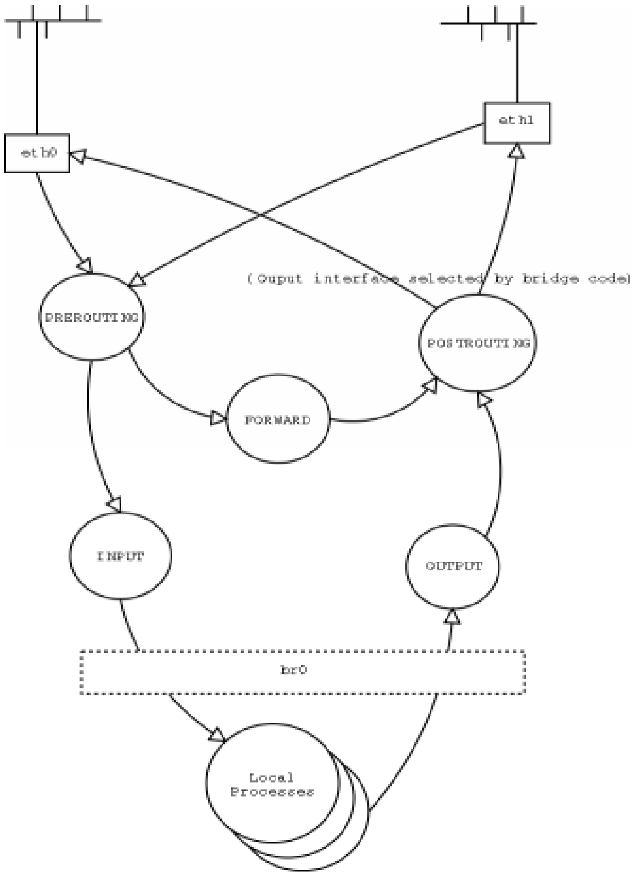
The first was untenable. Multiplying the number of iptable configurations multiplied the chances of error and the administration overhead. The second had its own drawbacks. While most things can be set up to work quite happily with IP masquerading, there are exceptions, including some technologies that I wished to explore, such as VPNs. A bridging firewall would solve this problem. The firewall could stand between the ADSL router and the rest of the router and protect the network without reconfiguring the router. The one remaining obstacle was that the bridging code in the standard Linux kernel completely bypasses iptables, so you can have a box which is either a bridge, or a firewall, but not both.

The Solution

Fortunately, there is a project to implement bridging in conjunction with iptables, so that any packets transmitted across the bridge can be subject to iptables rules. The result is a firewall that can be totally transparent to the network, requiring no special routing. As far as the Internet is concerned, the firewall does not exist, except that certain connections are blocked. The bridge software is a kernel patch to allow the existing bridge code to work inside iptables. Conveniently, the developers have made available a Redhat 7.2 kernel rpm with the patch installed. Less conveniently, documentation on how to use it is minimal, so I thought to document this implementation as an aid to anyone else following the same path.

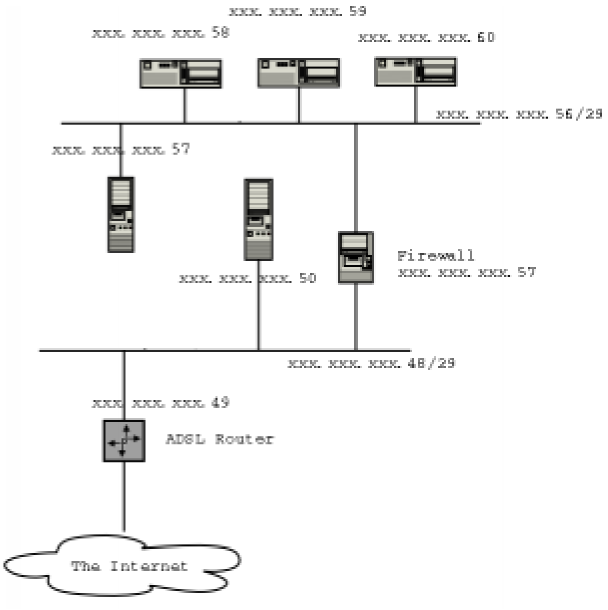
Bridging and Routing – how it works

Briefly. the linux bridge implementation works by tying together two or more network interfaces. By monitoring activity on all the attached network segments the bridge code learns which MAC addresses are accessible from each interface and uses this information to decide which packets to send out on each interface. The interfaces attached to the bridge to not normally have an IP address associated with them, but the entire bridge is configured as a single interface to the firewall.



NETWORK TOPOLOGY

My allocated static IP addresses are in the range xxx.xxx.xxx.48-63, i.e. a subnet mask of 255.255.255.240. I decided to split this range into two network segments,xx.xxx.xxx.48-56 would be used outside the firewall, and this includes the IP address of the ADSL router itself (xxx.xxx.xxx.49). xxx.xxx.xxx.57-62 would be the secure section behind the firewall. Note that these are not truly subnets as they are linked by a bridge rather than a router.



FIREWALL RULES

The sample firewall script is broadly similar to a conventional firewall setup (cribbed from Oskar Andreasson's iptables tutorial. The basic firewall policy is:

- Block packets from unlikely IP addresses
- Allow any outgoing connections from behind the firewall
- Allow packets in that belong to established connections
- Allow connections to specified ports and hosts from outside

Variable definitions

For clarity and maintainability it is a good idea to keep interface names
and IP addresses as variables. The values used for these examples are:

```
BR_IP="xxx.xxx.xxx.57"
BR_IFACE=br0
LAN_BCAST_ADDRESS="xxx.xxx.xxx.63"
INTERNAL_ADDRESS_RANGE="xxx.xxx.xxx.56/2
9" INET_IFACE="eth1"
LAN_IFACE="eth0" LO_IFACE="lo"
LO_IP="127.0.0.1"
```

"xxx.xxx.xxx" represents the first three bytes of the network IP addresses. $INTERNAL_ADDRESS_RANGE is the secure network segment.

Setting up the bridge

We have to do a some less conventional things to set up the bridge. First we shut down our two interfaces and remove any IP address from them.

```
ifdown $INET_IFACE
ifdown $LAN_IFACE
ifconfig $INET_IFACE 0.0.0.0
ifconfig $LAN_IFACE 0.0.0.0
```

If you just executed these commands from a telnet connection (or ssh as you are so security conscious), get up and cross the room to your firewall's console.
Next we create a bridge and assign the Ethernet interfaces to it.

```
brctl addbr $BR_IFACE brctl addif
$BR_IFACE $INET_IFACE
brctl addif $BR_IFACE $LAN_IFACE
```

You can now bring up the bridge as an internal interface if you wish:

ifconfig $BR_IFACE $BR_IP

Blocking spoofs

We can block spoofed packets in the mangle PREROUTING chain. By blocking here we can catch both INPUT and FORWARDED packets at the same time. We use mangle PREROUTING rather than nat PREROUTING because only the first packet of each stream is checked in the nat table.

This line ensures that only packets with valid internal addresses are accepted on the internal interface.

```
$IPTABLES -t mangle -A PREROUTING -i $LAN_IFACE -s
$INTERNAL_ADDRESS_RANGE -j ACCEPT
```

And this prevents packets with internal addresses being accepted on the external interface:

```
$IPTABLES -t mangle -A PREROUTING -i $INET_IFACE !
-s $INTERNAL_ADDRESS_RANGE -j ACCEPT
```

Accessing the firewall from the internal network

You may choose to leave your firewall completely invisible to the network, or you may wish for convenience to allow connections from within. These commands will allow all connections to the firewall from the internal network only. You may wish to be more selective depending on your level of trust of your network systems and users.

```
$IPTABLES -A INPUT -p ALL -i $BR_IFACE -s
$INTERNAL_ADDRESS_RANGE -d $LAN_BCAST_ADDRESS -j
ACCEPT
$IPTABLES -A INPUT -p ALL -i $BR_IFACE -s
$INTERNAL_ADDRESS_RANGE -d $BR_IP -j ACCEPT
```

Remember that we have already eliminated packets that claim to be from $INTERNAL_ADDRESS_RANGE that appear on the wrong interface.

More information

The kernel patch (http://bridge.soureforge.net/) without which all your iptables rules are in vain. Oskar Andreasson's iptables tutorial is recommended reading. (http://www.boingworld.com/workshops/linux/iptables-tutorial/iptables-tutorial/iptables-tutorial.html)

Try Rusty's Remarkably Unreliable Guides (http://netfilter.samba.org/unreliable-guides/) for background on packet filtering and networking.

Sparkle Home Page (http://www.sparkle-cc.co.uk/index.html) (the author's company)

Acknowledgments

Thanks to Lennert Buytenhek for a really useful patch, and also for reviewing this article.

David is an independent consultant working mostly for financial institutions in the City of London through his company, Sparkle Computer Co Ltd. With four young children, spare time is a precious commodity, so the daily commute from Sussex is his main opportunity for tinkering with Linux and open source software on his laptop.

This article is re-printed with permission. The originals can be found at:

http://www.linuxgazette.com/issue76/whitmarsh.html

Paranoid II – The Revenge of TinFoil Hat

Author: Berislav Kucan <ninja@evilmutant.com>

I need my encryption

In these times where privacy and security are one of the main topics surrounding Internet users and little bit advanced computer users, tools like Pretty Good Privacy and GNU Privacy Guard can make your private files really private. By creating a set of keys - private and public ones, you can be assured that your files or e-mails will be read by just the person(s) that those files were ment to. These key rings, should not be lost, because without them and the appropriate passphrase you won't have access to any of the files encrypted for you by a colleague, friend or mistress. If you are a Windows user, I presume you are used to lot of partition formatting and re-installing your favorite Miscrosoft Windows Something OS. By doing a backup of your files, Murphy's laws will make you forget to backup your GPG keyrings, and you are left in the dark then. It is important to have your keys and passphrase backuped somewhere (keys on a floppy, cdrom, graffiti in the notebook and passphrase in the lonely parts of your brain). Another important thing is that you must be able to encrypt or decrypt your files wherever you are. By influence of already mentioned Murphy's laws this scenario will happen:

> It is the third day of your business trip, and you got a few minutes to check your e-mail on a computer in your partner's network. You receive an important business related e-mail and it is ofcourse PGP'ed as it contains details about the latest part in the developement of your company's flagship product, and this is highly important to stay top secret. You are not on your desktop computer, so you don't have necessary PGP or GNU Privacy Guard installed, so what to do now?

TinFoil Hat saves the day

TinFoil Hat Linux is a small Linux distribution that can be easily booted from a floppy disk. As from my perspective its two best sides are that you have your ring pair backuped on one place and that you can securely encrypt and decrypt files wherever you are located. TinFoil Hat Linux is created to be a little paranoid place, so its other features surely go door-to-door with paranoia:

Anti KeyLogger feature: KeyLoggers are little software or hardware pests that are installed by people with malicious intent (for instance if your computer has been compromised so a KeyLogger is installed to snoop all your keystrokes and send them to attacker's e-mail address) or by people within the company infrastructure (that usually install hardware KeyLoggers to spy on what their employees spend their business time on). If you are on a non secure computer and you really need to use encryption, TinFoil Hat Linux has a nifty feature that gives you the abbility of entering your password in a secure manner, so the hardware KeyLogger (TinFoil Hat Linux boots from a floppy and as it is a seperate Operating System software keyloggers from other OS on this computer, of course, cannot work) and the people that check its logs cannot get to your passphrase. TinFoil Hat Linux used a wrapper for GPG, called gpggrid, that lets you use a video game style character entry system instead of typing in your passphrase. Don't think that it is something heavily graphical, as it is just a simple grid where column and row characters are randomly being changed. This finishes up with the possible KeyLogger getting aZ zT jP cJ aM hY instead of your passphrase "sensei" (just an example).

Anti Tempest feature: As seen from the readme.txt file attributed to this small Linux distribution - " TinFoil Hat Linux uses ctheme to manipulates the VGA console palette. It's an amusing hack, and does make it harder to photograph the screen with a digital camera, but it won't complicate tempest observation. It's the best I could figure out without having greyscale fonts." This feature is very interesting if you are into illuminati and big brother theories. What the heck – if your encrypted file is worth 2 million dollars, you should be afraid of people looking your screen over your shoulder, agents using their machines to grab your monitor signals, corporate spies using tele–kynesis powers and... and... well, you watched Enemy of the State, didn't you? BTW what to say about a software package that is being described by its author with the following line – "An exercise in paranoia or a day to day tool". Also, the thing degrates in a positive anti–paranoia sense – in order to complicate listening to radiation from the keyboard, TinFoil Hat Linux blinks encrypted messages in morse code on the keyboard LEDs.

SomeOtherThings feature: This operating system doesn't support networking, all binaries are compiled staticly, and all non–root partitions are mounted with no–execute permissions. All the files you work with are stored on an encrypted ramdisk which gets destroyed when you isue the final S (shutdown).

INSTALING THE TIN–HEAD

There are lot of people that like manuals, rather then read from accompanying readme files or from the software related web sites, so this is a step-by-step guide. First you should grab a copy of TinFoil Hat Linux that can be found on http://tinfoilhat.cultists.net. When you download the image, you should make a bootable floppy out of it:

1) Linux users
Linux users can use a simple dd command.

```
[bhz@localhost misc]$ dd if=tinfoil.img  \
of=/dev/fd0
2880+0 records in
2880+0 records out
bhz@localhost misc]$
```

2) Windows users
Windows users can use RawWrite that can be downloaded from:
http://uranus.it.swin.edu.au/~jn/linux

Figure 1.1 : Using rawrite under Windows for creating a boot disk

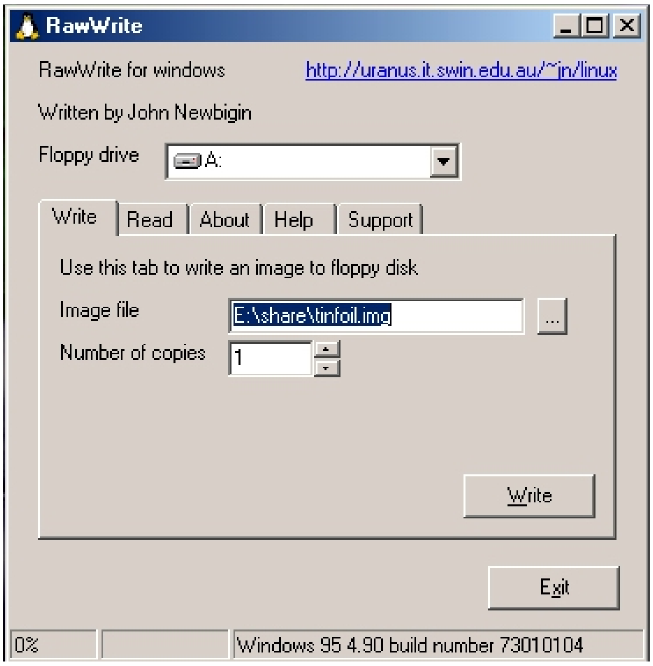
Now boot in your linux, mount the floppy and copy the contents of .gnupg directory in your home, to /gnupg directory on the floppy (/dev/fd0). If you wouldn't like your GPG information being un–encrypted, you can type:

```
[bhz@localhost misc]$ tar -cvf - $HOME/.gnupg |gpg
-co /mnt/floppy/ring.gpg
```

When you transfer mentioned files to your floppy (include public keys of people that you would like to correspond with), you should create an entropy.bin file which will make GPG encryption not predictable:

```
[bhz@localhost misc]$ gpg --gen-random 2 512 >
/mnt/floppy/entropy.bin
```

Now you are ready to go – floppy you created is a full working personal version of your GPG mini–box.



RawWrite

RawWrite for windows http://uranus.it.swin.edu.au/~jn/linux

Written by John Newbigin

Floppy drive: A:

Use this tab to write an image to floppy disk

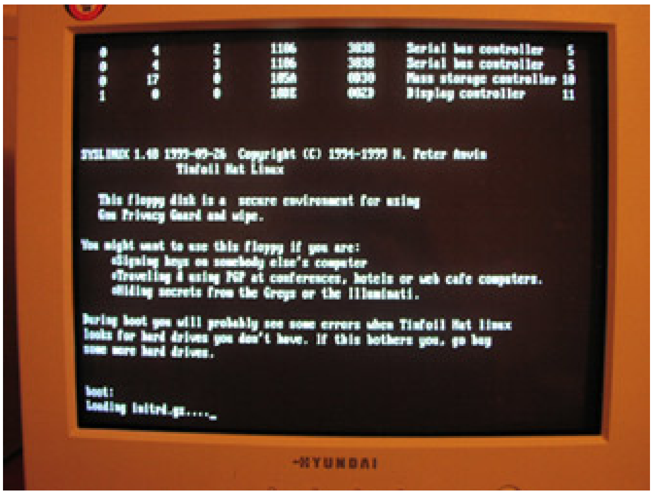
Image file: E:\share\tinfoil.img

Number of copies: 1

0%

Windows 95 4.90 build number 73010104

USING THE TIN–HEAD



```
0    4    2    1106    3038    Serial bus controller    5
0    4    3    1106    3038    Serial bus controller    5
0    17   0    105A    0D38    Mass storage controller 10
1    0    0    102B    0525    Display controller      11


SYSLINUX 1.48 1999-09-26  Copyright (C) 1994-1999 H. Peter Anvin
              Tinfoil Hat Linux

  This floppy disk is a  secure environment for using
  Gnu Privacy Guard and wipe.

You might want to use this floppy if you are:
    *Signing keys on somebody else's computer
    *Traveling & using PGP at conferences, hotels or web cafe computers.
    *Hiding secrets from the Greys or the Illuminati.

During boot you will probably see some errors when Tinfoil Hat linux
looks for hard drives you don't have. If this bothers you, go buy
some more hard drives.


boot:
Loading initrd.gz....._
```

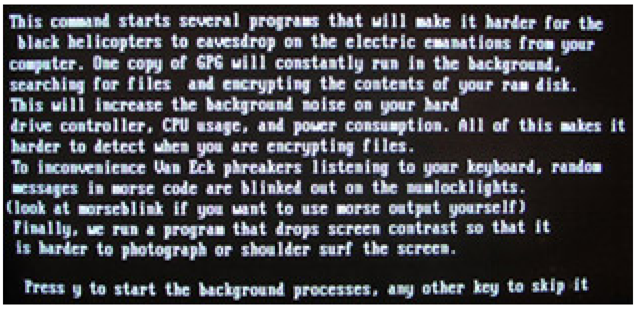
Figure 1.2 : TinFoil Hat Linux booting up

When you boot your TinFoil Hat Linux, you should enter "menu" command when it asks you to do it. After that, menu will who on your monitor and it looks something like this:

```
0) turn 0n paran0id settings
m) read the Manual for GPG, wipe, THL
f) switch to a new Floppy disk
e) start a text Editor
p) use GPG with gpggrid secure Passphrase entry
toolr
g) start using Gnu Privacyr
w) Wipe (erase) a floppy, hard disk or filer
s) Shutdown the computer (first backing up keys
and entropy to floppy)
x) eXit to shell<
```

Let's just quickly inspect the menu options:
- turn 0n paran0id settings

This will give you a possibility to use all the paranoid features of TinFoil Hat Linux I mentioned above. By entering 0, you will be presented with the text captured on the image below (click on the image for full size that is easily readable).



```
This command starts several programs that will make it harder for the
 black helicopters to eavesdrop on the electric emanations from your
computer. One copy of GPG will constantly run in the background,
searching for files  and encrypting the contents of your ram disk.
This will increase the background noise on your hard
drive controller, CPU usage, and power consumption. All of this makes it
harder to detect when you are encrypting files.
To inconvenience Van Eck phreakers listening to your keyboard, random
messages in morse code are blinked out on the numlocklights.
(look at morseblink if you want to use morse output yourself)
 Finally, we run a program that drops screen contrast so that it
 is harder to photograph or shoulder surf the screen.

  Press y to start the background processes, any other key to skip it
```

Figure 1.3 : entering the paranoid mode

After clicking yes, your Operating System becomes a paranoid box, witha grey screen that cannot be pictured with a digital camera (see example below).

Figure 1.4 : picture on the monitor goes grey – go away spies

- read the Manual for GPG, wipe, THL

This is self explanatory – you have three options to view whatever manual you are interested in reading on.

- switch to a new Floppy disk

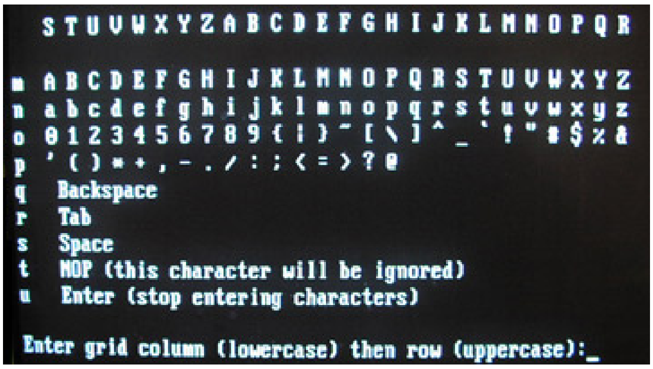
If you have additional software or files you would like to use. Just to note that you need to start this program every time you want to switch the floppy, or it won't work.

- start a text Editor

You can chose between vi and nano editors, which you can use for writing or reading files that are going to be encrypted, or that were just decrypted.

- use GPG with gpggrid secure Passphrase entry tool

This option lets you use mentioned gpggrid program, which is a secure way for entering your password. Images below show you how it is done.

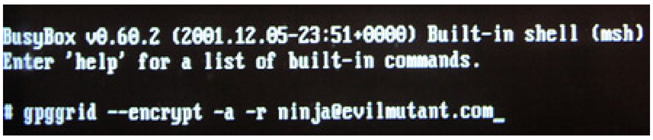


```
  S T U V W X Y Z A B C D E F G H I J K L M N O P Q R

m A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
n a b c d e f g h i j k l m n o p q r s t u v w x y z
o 0 1 2 3 4 5 6 7 8 9 { | } ~ [ \ ] ^ _ ` ! " # $ % &
p ' ( ) * + , - . / : ; < = > ? @
q  Backspace
r  Tab
s  Space
t  NOP (this character will be ignored)
u  Enter (stop entering characters)

 Enter grid column (lowercase) then row (uppercase):_
```

Figure 1.5 : using gpggrid to securely enter your password (note columns and rows)



```
BusyBox v0.60.2 (2001.12.05-23:51+0000) Built-in shell (msh)
Enter 'help' for a list of built-in commands.

# gpggrid --encrypt -a -r ninja@evilmutant.com_
```

Figure 1.6 : using gpggrid to start encrypting file for ninja@evilmutant.com

Quick note: When you are encrypting a text or file using –o flag (output file), be sure to move the newly created encrypted file from /tmp/ to /mnt/floppy or when you shut down the system it will be permanently deleted.

- start using Gnu Privacy Guard

This option drops you in the UNIX shell, so you can use your favorite GPG encryption tool in the manner you use it on your own computer.

- Wipe (erase) a floppy, hard disk or file

Wipe program securely erases files or hard drives byoverwriting them with many passes of random junk.

Wipe a file – "wipe filename"
Wipe a floppy – "wipe –D /dev/fd0"

- Shutdown the computer (first backing up keys and entropy to floppy)

When you are shutdowning the sytem, you can chose on of the following options:

1) Don't back it up (obviously doesn't backs up your GPG information)
2) Save it as an encrypted backup file (paran0id option)
3) Save it in the gnupg directory (defaulf option)

- eXit to shell

No comment ;)

BRIEF CONCULSION

TinFoil Hat Linux is a nifty tool that has not many features, but the ones it has are very interesting and useful. The whole idea behind this Linux distribution is pretty innovative, and I hope that further versions of TinFoil Hat Linux will incorporate additional security and privacy tools.

This article is re-printed with permission. The originals can be found at:

http://www.evilmutant.com/stuff/tinfoil/

Linux vs Windows 2000 Security Alert Comparison

Author: Con Zymaris <conz@cybersource.com.au>

In a recent piece on his WinInformant web site and mailing list, news editor Paul Thurrott (thurrott@winnetmag.com) questioned the generally accepted notion amongst IT professionals that Linux is more inherently secure than Microsoft's professional operating system platforms. Thurrott states:

> Let's examine a more recent example. In Friday's WinInfo Daily UPDATE newsletter, I mentioned a set of statistics from BugTraq, a reputable security- information provider, that shows how various OSs compare securitywise. The statistics show a surprising trend: When you aggregate all the Linux distributions, Linux, not Windows, has had the most security vulnerabilities, year after year.

There has been much discussion about the security vulnerability rates between Windows and Linux. Firstly, let me state that this focus on pure numbers and graph plots of vulnerabilities is pointless. There is no such thing as a truly secure operating system, there is only the ongoing process of keeping a host or network secure. One can never achieve a state of 'security Nirvana'. Think of it as a treadmill, constantly moving you (as a system administrator) backwards. You have to 'walk' forward just to keep still. If you don't move forward with security patches, security tools, revamped system security processes, you'll be flung off the end of the treadmill from sheer inactivity, and by the way, the crackers have access to the treadmill's speed control knob, and keep pushing up the speed.

As an ancillary, all operating systems can be made 'secure', by whatever reckoning you attribute to this term. It all boils down to time, effort, money and will. What is security worth to you and your network? Some operating systems seem to need more of these, some less. They all need some. The Open Source community has made much of the 'with enough eyeballs, all bugs are shallow' concept; that by using enough technical users, some or many security concerns can be overcome. I am a believer of this epithet, however, think about it for a second: 'with enough eyeballs, all bugs are shallow'. What this is saying, in effect, that when a bug becomes an issue, many people have the source code, and it can be quickly resolved. To paraphrase, when we get hit by a bug, we can swat it quickly and without waiting for a vendor. I believe that for widely used free software projects, this too is true. There is one important proviso to this train-of-thought to keep in mind though, which makes exploitable security bugs a slightly different beastie to general-purpose bugs. A general bug which hits an individual user or site, gets reported to the maintainers and gets resolved, generally doesn't have the same possible impact as a security bug, particularly a remotely exploitable one. A general bug (if catastrophic enough) can cause loss of data or system un-availability, but a security bug can cause your system to become 'owned' by a cracker, for you to lose data through deletion, have data sent to your competitors or leaked to the trade press, have invalid data inserted into your records, have customer credit cards stolen etc etc. Further, vulnerabilities become known and spread on back-room IRC channels like wildfire. While a general bug may be encountered by you and a few others over the course of a segment of time; a remotely exploitable vulnerability has the attribute of attracting penetrative tests against tens of thousands of hosts in a matter of hours of discovery, causing far more damage and strife than a general bug. Finally, catastrophic general bugs which affect many are few and far between (unless you include various Microsoft Service Packs), as most people do not tread the bleeding edge of operating system releases, and widely used systems and sub-system software generally doesn't harbour catastrophic general bugs for long. Security vulnerabilities, however, can arise in code or a subsystem which is widespread and very well entrenched, further accentuating the possible spread of damage.

In summary, the dues-ex-machina of 'with enough eyeballs, all bugs are shallow' holds, but possibly only after substantial damage has been done to many hosts on many networks. At least we know that if it's important for users of the said sub-system, the security problem will be resolved at the source-level, a surety we don't have with commercial closed-source or orphaned software.

While there are various industry correspondents who have eloquently outlined the steps that are necessary in the design and development of software which has a tendency to be more secure, a good approach to software security can be quickly given. Design the software with multiple layers of trust. Design it so that no part immediately trusts the other part. Make it small. Make it modular. Use languages which can either avoid buffer-overflow problems, or perhaps can be put through automated testing and parsing of the source for signatures of these problems. Allocate enough resources to security audits and reviews of the code from a security perspective. Design simple checklists for your coders (junior and senior) which point out the 10 most likely security failings for the platform/language/development paradigm you are developing your project under. It's easy stuff. Avoid complex security jargon, or excessive overtones of ideas or terminology which overshadows the many simple automaton-like things that can be done to improve information system security; it just scares developers away.

Now, onto a rebuttal of some of the points raised by Paul Thurrott, and a hint to others who have tried to run the vulnerability numbers through the analysis wringer. There is one crucial concept which seems to have gone missing from all the mainstream discussion to date, which I will present here. Thurrott claims that through sheer raw number of vulnerabilities calculated by BugTraq, Linux is less secure than Windows. Now, keeping in mind all we have said above about how the security of a system or network is linked to the process the system administrator uses, rather than the OS in question, let us proceed. Thurrott states:

> If you break down those numbers by Linux distribution (despite the fact that Windows 2000 and Windows NT are lumped together), Win2K/NT had 42 vulnerabilities in 2001 (data is through August only), and the leading Linux distribution, Red Hat, had 54. In 2000, Win2K/NT had 97 and Red Hat Linux had 95.

These numbers may in toto, be accurate. I don't dispute them. They appear to be slightly in Windows' favour. However, as mentioned above, what has not been discussed widely, reviewed and broadly digested (to my amazement), is that none of these industry observers has taken into account the substantial disparity in system functionality which is shipped on each platform, and which forms the software basis from which vulnerabilities arise . Let me elaborate. I reviewed the broadly categorised functionality packages which ship with Windows 2000 Server, presuming it be a reasonable superset of a generally available Microsoft platform, bundling most of the sub-systems which are needed by a user or business. The list of features is quite reasonable,and is shown by Microsoft here: (http://www.microsoft.com/catalog/display.asp?site=656&subid=22&pg=2) I count approximately 120 sub-systems in Windows 2000 Server. These include such this as Internet Information Services web server, Active Server Pages (ASP) Programming Environment, XML Parser etc. Now, to compare, I quickly researched a list of sub-systems which are shipped with a modern Linux distro. SuSe seemed to have such a list readily available for their 7.3 Professional release, so I used theirs. You too can view this list here: (http://www.suse.de/en/products/suse_linux/i386/packages_professional/index.html)

I'm sure the Red Hat, Debian et al. lists are similar. The weight-in? Just under 2600 packages. This means that based on just this simple analysis, a modern Linux distribution ships with approximately 20 times more functionality in the box than Microsoft ships with Windows 2000 Server. Note, this is just a count of approximate functionality. With the hundreds of millions of lines of source code shipping for these platforms, a much deeper analysis would be un-tenable. When one does a quick and dirty calculation therefore, Linux on a per-atomic-functionality basis, can be viewed as being 20 times more secure than Windows, i.e it ships with 20 times as much materiel, but releases approximately the same number of security alerts as Windows.

If this analysis proves anything, it's that this simple-minded churning of numbers is pointless. This is merely rhetoric flying back and forth, with the big minus being that Paul Thurrott and I are far cry from Socrates and Plato. But hey, he started it ;-)

References:
SecurityFocus Article:
http://securityfocus.com/vulns/stats.shtml

This article is re-printed with permission. The originals can be found at: http://www.cyber.com.au/users/conz/

1001 things to do with Liquid Nitrogen

Ulrich Schneider <ubws@gmx.net>

Once in a while a piece crosses your editor's desk which serves to remind that we here within AUUG are more emphatically linked to kindred spirits in the academemic and scientific world, than perhaps to other professional services communities such as the, ahem, accountants and management consultants. Here is such a piece. Enjoy. I did. Warning: don't try the experiment with the car.

In the course of studying physics one is officially taught that liquid nitrogen is simply (and mainly) used to cool things down to 77K. But everybody who once has observed students in practical courses "working" with this stuff knows that this is not true.

My intention is now to tell the truth about what is really done with liquid N2 before its remains are taken and used for cooling.

As we all know liquid nitrogen is mainly used for...

- making icecream by stirring for example yoghurt under it. (mind the carpet!; Darmstadt Group)

Roger Carlson comments on this topic "I have pix of making ice cream (with a good recipe), feel free to link to them if you want:" http://www.rogerandjudycarlson.com/roger/icecream/pix_ice_cream.html

- putting pieces of chalk in it for making little hovercrafts (best on linoleum floors!)

- twirling in large basins so that because of its low viscosity you get a (nearly) infinitly turning maelstrom. It's good fun to watch little paper-boats floating on it for minutes.

- inhaling its fumes because everybody will make eyes on you exhaling.

- freezing your partner's chair while he is shortly absent.

- for squirting water in it. If you use a spray-bottle you can squirt funny ice patterns into a basin with nitrogen. My alltime favorite: Helmar's ice-earrings

- one word: marshmellows

- its nice for cooling a good beer in a basin of water on which the nitrogen is poured (not much fun to look at, but great fun to drink; Darmstadt Group)

- Put on a rubber surgical glove with a hot dog (saussage) stuck in one of the fingers. Put the hot dog in the liquid nitrogen and then, to the amazement of your friends, smash your "finger" with a hammer. (Wes Denisson) Comment: Keep in mind which finger...

- Get a pot of boiling water and pour some nitrogen in it. You will watch the mists of hell shrouding the floor. It's good fun to test how long you can stand sticking a finger into it - a cool feeling ...

- Get about a liter of soap bubble solution hot and pour about a cupful of liquid nitrogen in it. Bubbles go everywhere! (Wes Denisson)

- Break a light bulb, put the filament into liquid nitrogen and turn it on. Looks cool! (Wes Denisson)

- Put a little bit of nitrogen in a can with a plastic snap on lid. We use a Pringles Chip can. After you pour in the nitrogen seal the lid. The lid will pop off with a boom and fly off. (David Hutchison)

- Blow up a balloon. Put the inflated balloon in the nitrogen. It will deflate, then take it out and it will inflate as it warms up. (David Hutchison)

- A siberian frog frozen in liquid nitrogen shall come to life again if you throwit back into the water. (Prof. Alois Loidl, who never tried it in public, but used a wind-up frog of his children instead, for demonstration)

- Freeze a can of shaving cream and then peel the can away from the cream. Put the canless cream into someone's car. Let the oven-like heat from the car's sitting in the sun defrost the shaving cream. 2 cans will fill an entire car. (Coulter C. Henry,Jr.)

- Freeze a banana in liquid nitrogen and use it to hammer a nail. (Wes Dennison)

[Und fur-our Deutsche-sprachen freunds :--ed]

Here is a small anecdote I will just quote: "Wir haben hier nebenbei auch 'ne Anwendung entdeckt. Eigentlich wollten wir eine wassergefüllte PET-Flasche (Cola) unter Druck setzen und dann als Rakete hochschießen. Mit Aufpumpen haben wir leider nur 5 bar erreicht. Deswegen haben wir in die Colaflasche ca. halb mit Wasser gefüllt. und dann ca 100 - 200 ml LN2 zugegeben und den Deckel geschlossen. Im Deckel war ein Loch in das wir ein Fahrradventil (nur die äussere Röhre ohne den eigentlichen Ventileinsatz) gesteckt hatten. Da drin war ein Gummistöpsel. Eigentlich sollte es bei Erreichen des Enddrucks (was auch immer der hätte sein sollen) den Stopfen rausdrücken und die Rakete vom Wasserstrahl hochgehoben werden. Es hat aber den gesamten Schraubdeckel abgerissen. Das Wasser ging ziemlich schnell raus und die Rakete ist immerhin bis zum 7. Stock (ca. 30m) geflogen." (Thnx to Markus Selve in Stuttgart)

Here is another quote: As an employee of the Franklin Institute Science Museum in Philadelphia, Pennsylvania, I had many occasions to use liquid nitrogen in our Hot & Cold show:

One thing we used to do for smaller groups was to freeze a graham cracker and then eat it. The vapors released through your mouth and nose are quite dramatic and it really does tintilate your tastebuds! Of course, we usually waved the cracker around just a little before eating it to be sure no drops of the really cold stuff linger. (Thnx to Jeeplass in Philadelphia)

This story was mailed to me too: For several years our Society of Physics Students chapter has entertained visiting students with a spectacular liquid nitrogen depth charge.

The term "depth charge" is used because we have a large extremely durable plastic trash can filled with about 40 cm of water. - After a short safty talk, focusing on the rule of NEVER tightly sealing a vessel containing liquid nitrogen, we use a long-necked metal funnel to pour perhaps half a liter of liquid nitrogen into an ordinary 2 liter soda bottle. Then we tightly screw on the cap, and drop it into the water! - For several seconds, one can hear the bottle expanding! The preferential orientation of the polymers makes the bottle get longer and longer, rather than a more spherical expansion. However, eventually the polymers just can't take it anymore, and BOOM! A quite satisfying detonation, sending water, nitrogen vapor, and bits of plastic high into the air. - The heavy duty plastic can serves to direct the "shrapnel" upwards, it is lots safer this way versus just setting the bottle on the grass and running away! (picture 1, picture 2; Thnx to Earl Blodgett in Wisconsin)

Larry Weinstein sent me the following: We have two more demos we use LN2 for here at ODU:

1) Take a 'ringshooter' (used to demonstrate Lenz's Law by placing an aluminum ring around an AC electromagnet [made by wrapping wire around a long thin iron core - typically 15-20 cm high and 3 cm in diameter] - the Al ring will jump into the air, a split Al ring and a nonconducting ring will not move) and demonstrate that the Al ring will jump from the magnetic repulsion. Now chill the Al ring in LN2. Repeat the demonstration and the ring will jump MUCH higher (since its resistance decreases substantially at -200 C)

2) Take a thinwalled metal cone, point downward (a sealed metal funnel will work). Fill it with LN2. Wait. Oxygen will condense out of the air and drip from the tip of the cone. Hold the tip of the funnel between the poles of a strong magnet. The drops of liquid oxygen will levitate there (if the field is strong enough) giving a rare good demonstration of paramagnetism. (This demo is courtesy of Sebastian Kuhn, also at ODU.)

Four suggestions by TOM MILLER(Air Force Research Lab):

(1) Start a show by sticking one end of very flexible tubing (e.g., latex or tygon) down into a dewar; the heat of the tubing will cause LN2 to spray out the other end of the tubing, and you can direct the spray at the audience. After the submerged end of the tubing is completely frozen (and the spraying stops), remove from the dewar and whack the frozen end on a table and watch it break into pieces.
(2) Wrap a long piece of latex tubing around itself and stick the whole thing into a dewar of LN2 until completely frozen. Remove and place on a table, and continue with the rest of your show. After a few minutes, the tubing will slowly start to move, sometimes crawling across the table.
(3) Stick flowers in LN2 and then crumble them in my hand; large ones like carnations are best. Sounds simple, but the kids love it.
(4) I freeze balloons, as you mention, but in a better way. Blow up a balloon and slip the end of the balloon over the open end of a test tube, and place the closed end in a dewar full of LN2. Your breath in the balloon will slowly liquify (10-15 minutes). When the balloon is completely deflated, lift the test tube out of the dewar and the audience can see your liquified breath in the test tube. The tube will frost up, but you can wipe the frost off with your fingers. Rest the test tube in a beaker, and as time passes, the balloon will inflate again.

This article is re-printed with permission. The originals can be found at:

http://www.physik.uni-augsburg.de/~ubws/nitrogen.html

The Great Giveaway

The New Scientist's Experiment in Open Content Licencing

Introduction

Good ideas are worth money. So why are hard headed operators giving them away for free? Join our experiment to find out says Graham Lawton

IF YOU'VE BEEN to a computer show in recent months you might have seen it: a shiny silver drinks can with a ring-pull logo and the words "opencola" on the side. Inside is a fizzy drink that tastes very much like Coca-Cola. Or is it Pepsi?

There's something else written on the can, though, which sets the drink apart. It says "check out the source at opencola.com". Go to that Web address and you'll see something that's not available on Coca-Cola's website, or Pepsi's--the recipe for cola. For the first time ever, you can make the real thing in your own home.

OpenCola is the world's first "open source" consumer product. By calling it open source, its manufacturer is saying that instructions for making it are freely available. Anybody can make the drink, and anyone can modify and improve on the recipe as long as they, too, release their recipe into the public domain. As a way of doing business it's rather unusual--the Coca-Cola Company doesn't make a habit of giving away precious commercial secrets. But that's the point.

OpenCola is the most prominent sign yet that a long-running battle between rival philosophies in software development has spilt over into the rest of the world. What started as a technical debate over the best way to debug computer programs is developing into a political battle over the ownership of knowledge and how it is used, between those who put their faith in the free circulation of ideas and those who prefer to designate them "intellectual property". No one knows what the outcome will be. But in a world of growing opposition to corporate power, restrictive intellectual property rights and globalisation, open source is emerging as a possible alternative, a potentially potent means of fighting back. And you're helping to test its value right now.

The open source movement originated in 1984 when computer scientist Richard Stallman quit his job at MIT and set up the Free Software Foundation. His aim was to create high-quality software that was freely available to everybody. Stallman's beef was with commercial companies that smother their software with patents and copyrights and keep the source code--the original program, written in a computer language such as C++--a closely guarded secret. Stallman saw this as damaging. It generated poor-quality, bug-ridden software. And worse, it choked off the free flow of ideas. Stallman fretted that if computer scientists could no longer learn from one another's code, the art of programming would stagnate (New Scientist, 12 December 1998, p 42).

Stallman's move resonated round the computer science community and now there are thousands of similar projects. The star of the movement is Linux, an operating system created by Finnish student Linus Torvalds in the early 1990s and installed on around 18 million computers worldwide.

What sets open source software apart from commercial software is the fact that it's free, in both the political and the economic sense. If you want to use a commercial product such as Windows XP or Mac OS X you have to pay a fee and agree to abide by a licence that stops you from modifying or sharing the software. But if you want to run Linux or another open source package, you can do so without paying a penny--although several companies will sell you the software bundled with support services. You can also modify the software in any way you choose, copy it and share it without restrictions. This freedom acts as an open invitation--some say challenge--to its users to make improvements. As a result, thousands of volunteers are constantly working on Linux, adding new features and winkling out bugs. Their contributions are reviewed by a panel and the best ones are added to Linux. For programmers, the kudos of a successful contribution is its own reward. The result is a stable, powerful system that adapts rapidly to technological change. Linux is so successful that even IBM installs it on the computers it sells.

To maintain this benign state of affairs, open source software is covered by a special legal instrument called the General Public License. Instead of restricting how the software can be used, as a standard software license does, the GPL--often known as a "copyleft"--grants as much freedom as possible (see http://www.fsf.org/licenses/gpl.html). Software released under the GPL (or a similar copyleft licence) can be copied, modified and distributed by anyone, as long as they, too, release it under a copyleft. That restriction is crucial, because it prevents the material from being co-opted into later proprietary products. It also makes open source software different from programs that are merely distributed free of charge. In FSF's words, the GPL "makes it free and guarantees it remains free".

Open source has proved a very successful way of writing software. But it has also come to embody a political stand--one that values freedom of expression, mistrusts corporate power, and is uncomfortable with private ownership of knowledge. It's "a broadly libertarian view of the proper relationship between individuals and institutions", according to open source guru Eric Raymond.

But it's not just software companies that lock knowledge away and release it only to those prepared to pay. Every time you buy a CD, a book, a copy of New Scientist, even a can of Coca-Cola, you're forking out for access to someone else's intellectual property. Your money buys you the right to listen to, read or consume the contents, but not to rework them, or make copies and redistribute them. No surprise, then, that people within the open source movement have asked whether their methods would work on other products. As yet no one's sure--but plenty of people are trying it.

Take OpenCola. Although originally intended as a promotional tool to explain open source software, the drink has taken on a life of its own. The Toronto-based OpenCola company has become better known for the drink than the software it was supposed to promote. Laird Brown, the company's senior strategist, attributes its success to a widespread mistrust of big corporations and the "proprietary nature of almost everything". A website selling the stuff has shifted 150,000 cans. Politically minded students in the US have started mixing up the recipe for parties.

OpenCola is a happy accident and poses no real threat to Coke or Pepsi, but elsewhere people are deliberately using the open source model to challenge entrenched interests. One popular target is the music industry. At the forefront of the attack is the Electronic Frontier Foundation, a San Francisco group set up to defend civil liberties in the digital society. In April of last year, the EFF published a model copyleft called the Open Audio License (OAL). The idea is to let musicians take advantage of digital music's properties--ease of copying and distribution--rather than fighting against them. Musicians who release music under an OAL consent to their work being freely copied, performed, reworked and reissued, as long as these new products are released under the same licence. They can then rely on "viral distribution" to get heard. "If the people like the music, they will support the artist to ensure the artist can continue to make music," says Robin Gross of the EFF.

It's a little early to judge whether the OAL will capture imaginations in the same way as OpenCola. But it's already clear that some of the strengths of open source software simply don't apply to music. In computing, the open source method lets users improve software by eliminating errors and inefficient bits of code, but it's not obvious how that might happen with music. In fact, the music is not really "open source" at all. The files posted on the OAL music website http://www.openmusicregistry.org so far are all MP3s and Ogg Vorbises--formats which allow you to listen but not to modify. It's also not clear why any mainstream artists would ever choose to release music under an OAL. Many bands objected to the way Napster members circulated their music behind their backs, so why would they now allow unrestricted distribution, or consent to strangers fiddling round with their music? Sure enough, you're unlikely to have heard of any of the 20 bands that have posted music on the registry. It's hard to avoid the conclusion that Open Audio amounts to little more than an opportunity for obscure artists to put themselves in the shop window.

The problems with open music, however, haven't put people off trying open source methods elsewhere. Encyclopedias, for example, look like fertile ground. Like software, they're collaborative and modular, need regular upgrading, and improve with peer review. But the first attempt, a free online reference called Nupedia, hasn't exactly taken off. Two years on, only 25 of its target 60,000 articles have been completed. "At the current rate it will never be a large encyclopedia," says editor-in-chief Larry Sanger. The main problem is that the experts Sanger wants to recruit to write articles have little incentive to participate. They don't score academic brownie points in the same way software engineers do for upgrading Linux, and Nupedia can't pay them.

It's a problem that's inherent to most open source products: how do you get people to chip in? Sanger says he's exploring ways to make money out of Nupedia while preserving the freedom of its content. Banner adverts are a possibility. But his best hope is that academics start citing Nupedia articles so authors can earn academic credit.

There's another possibility: trust the collective goodwill of the open source community. A year ago, frustrated by the treacle-like progress of Nupedia, Sanger started another encyclopedia named Wikipedia (the name is taken from open source Web software called WikiWiki that allows pages to be edited by anyone on the Web). It's a lot less formal than Nupedia: anyone can write or edit an article on any topic, which probably explains the entries on beer and Star Trek. But it also explains its success. Wikipedia already contains 19,000 articles and is acquiring several thousand more each month. "People like the idea that knowledge can and should be freely distributed and developed," says Sanger. Over time, he reckons, thousands of dabblers should gradually fix any errors and fill in any gaps in the articles until Wikipedia evolves into an authoritative encyclopedia with hundreds of thousands of entries.

Another experiment that's proved its worth is the OpenLaw project at the Berkman Center for Internet and Society at Harvard Law School. Berkman lawyers specialise in cyberlaw--hacking, copyright, encryption and so on--and the centre has strong ties with the EFF and the open source software community. In 1998 faculty member Lawrence Lessig, now at Stanford Law School, was asked by online publisher Eldritch Press to mount a legal challenge to US copyright law. Eldritch takes books whose copyright has expired and publishes them on the Web, but new legislation to extend copyright from 50 to 70 years after the author's death was cutting off its supply of new material. Lessig invited law students at Harvard and elsewhere to help craft legal arguments challenging the new law on an online forum, which evolved into OpenLaw. Normal law firms write arguments the way commercial software companies write code. Lawyers discuss a case behind closed doors, and although their final product is released in court, the discussions or "source code" that produced it remain secret. In contrast, OpenLaw crafts its arguments in public and releases them under a copyleft. "We deliberately used free software as a model," says Wendy Selzer, who took over OpenLaw when Lessig moved to Stanford. Around 50 legal scholars now work on Eldritch's case, and OpenLaw has taken other cases, too.

"The gains are much the same as for software," Selzer says. "Hundreds of people scrutinise the 'code' for bugs, and make suggestions how to fix it. And people will take underdeveloped parts of the argument, work on them, then patch them in." Armed with arguments crafted in this way, OpenLaw has taken Eldritch's case--deemed unwinnable at the outset--right through the system and is now seeking a hearing in the Supreme Court.

There are drawbacks, though. The arguments are in the public domain right from the start, so OpenLaw can't spring a surprise in court. For the same reason, it can't take on cases where confidentiality is important. But where there's a strong public interest element, open sourcing has big advantages. Citizens' rights groups, for example, have taken parts of OpenLaw's legal arguments and used them elsewhere. "People use them on letters to Congress, or put them on flyers," Selzer says.

The open content movement is still at an early stage and it's hard to predict how far it will spread. "I'm not sure there are other areas where open source would work," says Sanger. "If there were, we might have started it ourselves." Eric Raymond has also expressed doubts. In his much-quoted 1997 essay, The Cathedral and the Bazaar, he warned against applying open source methods to other products. "Music and most books are not like software, because they don't generally need to be debugged or maintained," he wrote. Without that need, the products gain little from others' scrutiny and reworking, so there's little benefit in open sourcing. "I do not want to weaken the winning argument for open sourcing software by tying it to a potential loser," he wrote.

But Raymond's views have now shifted subtly. "I'm more willing to admit that I might talk about areas other than software someday," he told New Scientist. "But not now." The right time will be once open source software has won the battle of ideas, he says. He expects that to happen around 2005.

And so the experiment goes on. As a contribution to it, New Scientist has agreed to issue this article under a copyleft. That means you can copy it, redistribute it, reprint it in whole or in part, and generally play around with it as long as you, too, release your version under a copyleft and abide by the other terms and conditions in the licence. We also ask that you inform us of any use you make of the article, by e-mailing copyleft@newscientist.com.

One reason for doing so is that by releasing it under a copyleft, we can print the recipe for OpenCola without violating its copyleft. If nothing else, that demonstrates the power of the copyleft to spread itself. But there's another reason, too: to see what happens. To my knowledge this is the first magazine article published under a copyleft. Who knows what the outcome will be? Perhaps the article will disappear without a trace. Perhaps it will be photocopied, redistributed, re-edited, rewritten, cut and pasted onto websites, handbills and articles all over the world. I don't know--but that's the point. It's not up to me any more. The decision belongs to all of us.

FURTHER READING

The source code of this article plus details of the conditions can be found at:
http://www.newscientist.com/hottopics/copyleft

For a selection of copylefts, see:
http://www.eff.org/IP/Open_licenses/open_alternatives.html

The Cathedral and the Bazaar by Eric Raymond is available at:
http://tuxedo.org/~esr/writings/cathedral-bazaar/

This article is re-printed with permission. The originals can be found at:

http://www.newscientist.com/hottopics/copyleft/

Interview: Jordan K. Hubbard of the FreeBSD Project

Chat transcript of an interview with Jordan K. Hubbard (<_jkh>, core FreeBSD programmer and presently an employee of Apple) on 2002-01-27:

<_jkh> OK are we ready?

<starzz> I'm ready :)
<nev-bsd> im just a funnel

<Diesel> most of these questions came either from posts on the web site or users here on opn

<_jkh> OK fire away

<Diesel> many persons asked about smpng specifically what is the current status of SMPng? How do you think SMPng will fair against Linux 2.4.x's SMP support when it is stable?

<_jkh> Well, if I had to sum it up in a nutshell, I would say "on target, though late". the aims of SMPng are pretty grandiose; the complete multi-threading of the kernel, a rewrite of the scheduler, full preemptability and fine-grained resource locking... Naturally, everyone wanted to tackle all of the classic problems with making SMP scalable, not just one of them so far, interrupt threading is working very well and a lot of the finer-grained locking has been done or is in progress the scheduler rewrite is about to hit its first major milestone with KSE 3, which will be presented and discussed at BSDCon and what now remains is a lot more irritating lock-pushdown work and performance improvements to some of the locking code. I think that in around 6 months, it will be very fair to compare it to the Linux 2.4.x code and that it will compare very favorably, I'm hoping superior in a number of ways (both from a performance and a stability perspective)

<_jkh> next question?

<Diesel> ok from user "TheJerk": I was interested in what effects, if any, jordans ties to Mac and OSX have had on the freebsd project I have also been interested in how jordan feels towards the other BSD projects and why, when netbsd was just starting, did he feel it necessary to start the FreeBSD project?

<_jkh> is that the end of the question or does it pack any more in there? :)
<Diesel> haha true journalism here ;P

<_jkh> +i might not be a bad idea. OK, to take the first part of the question there haven't been a _lot_ of effects on FreeBSD from my Mac OS X work, but there have certainly been more lines of communication set up and there's a definite advantage to being able to communicate important bits of information back and forth like on security advisories and such and some code has flowed in both directions like the cool filesystem exerciser that Apple had which the FreeBSD folks were able to use to turn up about 4-5 really bad and long-term bugs in NFS and even one in the soft updates code that Kirk had been chasing for months that was a very simple thing to do, but it had major effects I hope to do more things like that as the opportunities come up so we'll see. as to NetBSD we both started literally around the same time and didn't even know about one another until we got big and well-organized enough to show up on eachothers radar by that time we'd already formed a group and decided on a mission and it was becoming increasingly clear that each group had a very different mission in mind you can't force volunteers with dissimilar interests to work together when they'd rather just communicate and work with some other set of engineers who share the same interests the net is big enough that you can reach critical mass without having to force fundamentally incompatible particles together. :)

<Diesel> very true. ..

<_jkh> so I think of what went on with all the *BSDs, to say nothing of the Linuxen, was simply engineers clumping up along social and technical lines. :) s/of // I'm done. :)

<Diesel> Several users were interested in how you felt about X. Would it be feasable to mimick some of the better attributes of the OS X gui with GNUStep or perhaps a BSD liscensed implementation

<_jkh> Mac OS X or the X Window System?

<Diesel> X windows

<_jkh> Well, I certainly have been using the X window system for a long time and have written a fair amount of software for it; for what it does, it rocks but for what it doesn't do, it really sucks too <_jkh> don't even talk to me about font handling or printing

<Diesel> lol

<_jkh> so I think that before you're going to see X really get some decent applications, you're going to have to finish the missing 5% of X the part that was scheduled to take 90% of the time <_jkh> and so nobody got around to it. Plus, the whole UI war thing needs to end Adobe is never going to port photoshop while nobody can answer "Which GUI environment is dominant and therefore recommended
for use?" so I think X will probably remain the DOS of window
systems

<Diesel> :(

<_jkh> used for a lot longer than anyone predicted, deeply loved by its adherants who know how to do absolutely anything with it, ignored by the mainstream who will have moved on. :) boy, I can tell that people don't like the harsh answers. :)

<Diesel> our next question again asked by many persons

<_jkh> what can I say, I don't have time to be politically correct. :)

<nev-bsd> some ppl only want to hear what they think they already know

<Diesel> regarding MTA in base system

<Diesel> "when will sendmail be replaced by more reliable MTA such as postfix?"

<_jkh> I don't think it will ever be _replaced_ <_jkh> but I fully expect the MTA to be an option in future FreeBSD releases we've talked about it for quite a long time. so you can consider sendmail, postfix or even possibly (gak) qmail to soon be selectable options. next?

<Diesel> Can you comment on the general attitudes of companies when they are approached for a freebsd port?

<_jkh> You mean of some application?

<Diesel> such as nvidia

<_jkh> Well, nvidia has its own rep where working with the open source folks is concerned, so I won't elaborate any more on that. :)

<nev-bsd> amen

<_jkh> but where we can make some kind of business case for it, they're generally pretty receptive

<Diesel> heh

<_jkh> it's just making a business case that's hard, and you can't really blame the company for wanting to hear one.

<wca> _jkh: how about on the application side?

<Diesel> would you care to comment on what you are currently working on ?

<_jkh> but I can say that companies have also been surprisingly willing to take a chance on us in the past. moving on to the next question

<wca> _jkh: I know Nik had some success with the theKompany.com
stuff

<_jkh> Do you mean for FreeBSD or Apple?

<wca> _jkh: and Loki/BSDi's relationship etc.

<wca> FreeBSD.

<_jkh> wca: Well, the Loki/BSDi relationship expired with Loki, I'm afraid. :(

<wca> Yeah, I know.

<wca> But it happened.

<_jkh> yes, Scott Draper was quite receptive to the idea

<wca> That's all that counts. Loki's expiration is not our fault, by any stretch of the imagination. even though I couldn't demonstrate any kind of FreeBSD desktop market at all. :) like I said, surprisingly willing to take a chance.

<wca> It's pretty hard to do that, given that we can't track the number of people who install FreeBSD and what they use it for without asking them directly.

<_jkh> Right now I'm just sort of taking some time to look at FreeBSD with a bit more objectivity than I've had in the past and think about what I'd still like to see it achieve

<wca> Your job at Apple gives you this opportunity?

<_jkh> hopefully I've got at least one more ports-collection type of idea in me where that's concerned. :) It gives me a different vantage point

<smn> _jkh The FreeBSD/sparc64 port webpage hasn't been updated in ages. What's its status? Is it going to support all sparc64 platforms, or just those with PCI?

<_jkh> wca: and I've managed to palm^H^H^Hdelegate away a lot of my more tedious FreeBSD responsibilities, like release engineering which gives me fresh energy to enjoy FreeBSD stuff again. next q?

<wca> _jkh: Yep, I know exactly how you feel :)

* _jkh brushes the cat's tail aside again, which is down the
middle of his monitor

<starzz> jkh: now you need to make an anti cat program I could use one of those ;) Jordan: any worries about elitism in the FreeBSD community being damaging to the projects goals... especially when dealing with linux users? that was a question from a user, Liemy I believe. Leimy that is!!

<_jkh> starzz: Nope

<nev-bsd> what are your thoughts on pf and ipf

<_jkh> starzz: Slashdot has been complaining about elitism in the *BSD community since day one, and even before slashdot it was something I heard was going to bring about the end of the world

<nev-bsd> oops

<_jkh> starzz: we're still here. :)

<starzz> :)

<_jkh> smn: I hear the Sparc64 port has actually reached single-user status and is doing very well lately. It was stuck more formerly than it is right now

<wca> _jkh: actually, jake and john reported early in january that we're multiuser on sparc64

* _jkh wonders what happened to Diesel

<wca> and close to being self-hosting

<teferi> What buses does it support, though?

<_jkh> wca: cool! that's even closer than I thought.

<Diesel> here..

<teferi> I have an ultra at home I'd *love* to run BSD instead of solaris on, but...

<wca> teferi: PCI only currently, sbus later

<Diesel> cant keep up with my terminal ;)

<teferi> (it's pre-pci) ahh.

<wca> teferi: Some Sun Microsystems guy really wanted to add support for sbus.
<teferi> rock. So sbus will be in 5.0?

<starzz> diesel: you can send some of the q's to me if you like

<wca> teferi: *shrug*

<nev-bsd> ok back to the questions that jordan might know more about what are your thoughts on pf and ipf

<_jkh> nev-bsd: I think there should have been more effort put into resolving the license issues, as we did, before going off
and writing a whole new IP filter. now we have 3: ipfw, ipf and
pf. Yay. :)

<wca> _jkh: pf was written before openbsd decided to jet ipf.

<nev-bsd> >:)

<starzz> new question? I have one ready

<_jkh> wca: perhaps we should try to have just one thread. :)

* wca agrees.

<_jkh> whether pf had its genesis before or after openbsd adopted it, we now have 3 mainstream filters and that seems a little silly. next question.

<starzz> From R0cky: We have all heard rumours about the upcoming 5.0, and the new features it will have. The most discussed and anticipated is probably SMPng and filesystem improvements. What do you feel is the "coolest" new feature, and which feature/improvement do you think will be the most noticable? Is SMPng and background fsck along with the filesystem tweaks really all its cranked up to be? Will fbsd be able to compete with other os'es that are typically used on multiple-cpu machines, such as Solaris and HP-UX, when it comes to SMP performance?

<_jkh> I think the coolest feature of 5.0 will be the ferret thread. This is a kernel thread which runs around throughout the system and randomly moves things around, playfully changes file modes, "bites" through open connections and just generally raises havok. We feel that this feature will make the computer more of a pet than a simple inanimate object

<Diesel> hahahaha

<nev-bsd> lol

<starzz> heh

<_jkh> sorry, the real answer

<_jkh> let me think about that for a moment to be honest, I really don't expect 5.0 to be something which really stands out to anyone but geeks in other words, if it's done correctly there will be very little visible impact though there will hopefully be some visible performance
gains the biggest advance will be internally, how things are put together and how scalable the internal mechanisms are

<Diesel> anything that stands out that is on radar for post 5.0?

<_jkh> that will allow 5.1 and other 5.x releases to perhaps be much more visibly impressive in some way well, I'm interested in what having a truly preemptable kernel will allow it means there will be a potential for real-time applications perhaps interesting A/V work for example an being able to leverage multiple CPUs in a truly useful way will make the n-way stuff much more interesting whether it's n-way internally (inside the CPU itself) or externally.

<SolarfluX> _jkh: from 'TheVince': I would like to know how many people work full-time on the FreeBSD project. Also, I read recently that you were considering improving the startup scripts by following a paper from the NetBSD project. What other things have NetBSD and OpenBSD done that influenced you to implement them as well?
<_jkh> it's hard to give a real number for "full time" but I would estimate perhaps 20 on average and at least a couple of hundred unique contributions each month and we take all kinds of stuff from NetBSD and OpenBSD that's what open source is about. :) I think the NetBSD startup stuff, for example, is pretty cool we're just trying to find enough bodies to finish that (the merge) I'm sure the NetBSD folks would like over 6000 ports in their ports collection too but they have the same problem. :) so we all share as much code as we have time to integrate. next q.

<starzz> one sec! from user blueroo: I'd be curious to see what exactly jkh knows of the navy's relationship with FreeBSD I recall an article a short time ago where the navy was pledging support for fbsd development. <EOQ>

<_jkh> We have a strictly don't ask, don't tell relationship with the Navy. sorry, sorry. in truth I don't know - I haven't heard much about
that project in quite some time. next q

<starzz> <eksffa> Recently, a Brazilian fbsd user group has posted a set of tools to make "live versions of freebsd" - freebsd ready to run on CDs - very easy tp customize and stuff... The tool set really works, and it seems as a good alternative to picobsd. Has anybody tried on the core or any important tests done using those scripts? If so, what did you think? Anyone considering it on the fbsd source tree
like picobsd is today? <EOQ>

<_jkh> well, first off, it's not really an alternative to picobsd picobsd can be embedded in very small (deliberately small) configurations, like flash cards or 1.44MB floppy disks this brazilian effort uses some of the same technology but it's more of a "demo CD" feature and other OSes, like Yggdrasil Linux, have had that going for years

<starzz> pocketlinux

<_jkh> so it's not exactly a new thing but I still think it's cool

<starzz> :) I agree

<_jkh> and I hope the brazilian group keeps cranking them out - they're useful to some people maybe someday they can add a rule to release/Makefile which does everything required and the release engineering team can consider making one available on an official basis. next q?

<starzz> <offset> jordan: what do you think about trustedbsd on freebsd 5.0?

<_jkh> I think trustedbsd will continue to merge code into 5.x; that's their charter other than that, it's up to them to set their schedule. next q?

<starzz> <Leimy> doesn't having a fully preemtable kernel add the possibility of reducing overall data throughput. I mean that's what the linux pre-emption stuff seemed to do.<EOQ>

<_jkh> No, since you're not going to change reality so fundamentally that just because you can be preempted, you frequently are. :)

<_jkh> it just makes it possible to deal with real-time and asyncronous events in a much cleaner and more natural fashion. I don't expect it to affect overall data throughput. next q?

<starzz> <kawfee> ok, Has Microsoft planned to implement a version of their OS with BSD? <eoq>

<nev-bsd> officially :)

<starzz> so thats a yes or no? :)

<_jkh> That would certainly explain why Bill's been looking at me strangely every time we have lunch lately. next q?

<starzz> heh

<starzz> <ppl_> ask if 4.5 will really be out tonight. only 12 minutes left until midnight <EOQ>

<_jkh> You'll know in 12 minutes next q?

<starzz> :) this one comes from an anonymous source!! Does apple have any plans to help the freebsd project in anyway? Funding, code, etc? They did use freebsd 3.2 as a starting point for Darwin after all, it seems only fair to give something back to the community that made MacOS X possible. PS: "Open Source" Darwin does not count :-) <eoq>

<_jkh> Hi Indigo2

<starzz> lol

<_jkh> The answer is "I don't know"

<Diesel> There seem to be a lot of new I/O implementations at all levels on the way, 3gio, hypertransport, serial ATA.... Are any of them definitive and close enough to start work on within FreeBSD? Is FreeBSD ever consulted bout these types of proposals for standards?

<_jkh> I know Apple would certainly like to find ways of helping where it can

<Diesel> sorry ..

<_jkh> though people don't fling money around much in this economy

*starzz pokes diesel

<Diesel> I hit enter like 2 minutes ago :(

<starzz> ;)

<_jkh> so we'll just have to see now I'll take Diesel's question; his lagged lagged question

<starzz> cool

<_jkh> I would say that of those, only serial ATA is really close and yeah, we have people already looking into it though we don't exactly get consulted about this kind of thing in advance either maybe in a more perfect world. next q?

<starzz> <teferi> Here's a question. Is the drm-kmod (hardware support for XFree86 4.x DRI) port going to be integrated into the core kernel/modules in 5.0? <eoq>

<_jkh> I don't specifically know if this is planned for 5.0, but I can say that critical mass for such a thing certainly seems to be building and it's something which could easily get done in time for 5.0 I use it myself with a Matrox G400 card and it rocks! next q?

<starzz> jkh: what do you think about FreeBSD using perforce internally and not everywhere as some would like? that is from <wca>
<eoq>

<_jkh> Internally meaning.. ? In some sense, it already is being used "internally" in that side-branch projects are being done in local/private Perforce repositories like the SCSI CAM code was done and then these repos are sync'd with the FreeBSD master CVS repo once the project advances to a certain milestone installing a p4d on freefall.freebsd.org probably won't ever be necessary, if that's the real question, since we have external special projects machines we'd tend to use anyway in such cases. next q?

<Diesel> When will we get some tasty blowfish, or AES or whatever for user password encryption? :P Are export restrictions the problem?
user chessie <eoq>

<_jkh> I believe that's already supported now. I see AES and blowfish support in libcrypto. though I'm not sure how one sets ones default password format to use them and perhaps that's the real question. I'd have to ask around. <sorry, I know that's a lame answer> next q?

<starzz> <movement_> how do you deal with interrupts being disabled / locks being held for long time wit preempt ? <EOQ>

<_jkh> 1. You don't disable interrupts for a long time. :) 2. In the worst case where you're contesting a lock, you generally just get put back to sleep and the preemption essentially refused though, of course, you'd be instrumenting the hell out of the kernel in order to find those sorts of lock contentions and fix them next q?

<starzz> chessie would like to know: Does jordan still do any "REAL" development work with FreeBSD? And what would he suggest to someone who was new to coding and that wanted to get involved and help out with the project / development? <eoq>

* _jkh waves to Grog

<_jkh> I no longer do any REAL development work with FreeBSD, no. Which is unfortunate since that's the only part I ever really enjoy, so I'm trying to get away from the grunt work again and more into the mode of doing small bursts of "real" work what I would suggest would be to lurk on the mailing lists for awhile and just listen to what people are talking/complaining about it's a bit like crossing a busy street - the best way to learn how is to become good at judging your moment and jumping in. :) and there's nothing like being hit a few times to teach you that lesson, so don't get discouraged if your first few efforts meet with scorn or indifference. :) next q?

<starzz> Anonymous: When will Perl be replaced by Python in the base system? <eoq>

<_jkh> starzz: Not before the editor learns to edit out questions like this, that's for sure. :)

* starzz hides

<starzz> dont kill the messenger!!!

<Diesel> that as my fault...

* starzz sighs

<Diesel> big list

<_jkh> On slashdot that would get moderated to −1 use that guideline maybe. :)

<nev−bsd> 2+ only

<_jkh> I also need to wander soon we've been doing this for 90 minutes

<Diesel> ay chance of some new artwork? Understood

<_jkh> New daemon artwork?

<Diesel> yes

<starzz> Thank you for your time jkh :)

<_jkh> That would be great, if any artists would consider doing something in the past, we've paid people to do stuff

<Diesel> hire openbsd guy?

<_jkh> and that never looked as good as the stuff people did for free diesel: We had him do one bit of art for us, but he's a bit unscrupulous personally so we wouldn't deal with him again

<Diesel> hmmm

<Diesel> birds of a feather?

<_jkh> no comment. :)

<Diesel> understood ;)

<_jkh> any last questions?

<nev−bsd> not fromme

<Diesel> do you mind if we post this to the web?

<starzz> here is a good one

<starzz> I have

<_jkh> diesel: nope

<starzz> its not really technical though :) jordan: i noticed there are not many if a few females that contribute to the project , why is that? that is from comp82

<_jkh> yes, why is that, I always ask I have no idea

<starzz> and as a female, I can relate to that question :)

<_jkh> we've tried very hard to recruit some but the only two female committers we ever had wandered off without doing anything

<nev−bsd> chix dig unix >:)

<_jkh> there seems to be some time conflict involved which I think has to do with the fact that most females try to have a life

<starzz> hmm. I should try that having a life some day ;)

<_jkh> whereas guys the same age are quite willing to forgo any semblance of one must be a hormonal thing

<starzz> :)

<Diesel> meaning we irc and play video games..

* _jkh is currently playing Giants: Citizen Kabuto

<starzz> actually, I have found more women in the telecommunications field, than the nix field... maybe we just feel intimidated?

<_jkh> and I get a lot of dark looks from the girlfriend. starzz: could be

<nev−bsd> i get "ugg compter crap"

<Diesel> is FreeBSD your os of choice? Still keep a windows box around for gaming?

<nev−bsd> thank you for your time and patience

<_jkh> starzz: but we still all wish you could just channel Grace Hopper a little and just soldier on into it. :)

<starzz> well, honestly, as a woman I was not given a lot of opportunity to learn a lot about *nix, I was not around it...

<_jkh> Diesel: FreeBSD for firewall, services, Unix desktop. Mac OS X for main desktop, office stuff and gaming. :)

<_jkh> and with that I must go. it's been fun folks

<starzz> Jordan we do thank you for your time, and patience in these questions.

<nev−bsd> and with the people

<Diesel> many thanks for your time!
<nev-bsd> >:)

<SolarfluX> Thanks for coming by Jordan
<_jkh> heh, sure thanks for organizing this. talk to y'all later!

* _jkh waves

*** _jkh has left #freebsd

This article is re-printed with permission. The originals can be found at:

http://bsdvault.net/hubbard.html

Symposium Report – 3rd AUUG Security Symposium

Author: Gary Gaskell <gary@gaskells.org>

The security symposium was held in Brisbane from the 19th to the 21st of November. There were 55 people in attendance and everyone reported the event of great interest to them.

This 3rd security symposium was the first to trial tutorials. While the AUUG should consider fine tuning the tutorials in the future, they were well received by attendees. In particular people believe it helps them justify interstate travel. Perhaps future years should run fewer tutorials on a single day as this spreads the attendees too thinly. A couple of people remarked to us how good the presenters were, compared to the numerous commercial options.

The symposium ran for two days with 14 presentations. There was also a panel on the last day. "Ask the experts" was a popular title, rather than aiming at discussing a particular topic. A proceedings was published with an ISBN. This is very important for academic contributors of which there was three.

The symposium is likely to have been better attended if we could have organised a big name international guest speaker. With the tight budget the executive decided that we should not pay for a speaker. We thought that we had one sponsored by eGlobal, however they renegged in the last couple of weeks, when they pulled out of the TruSecure service in Australia. We were very disappointed about this, as Russ Cooper would have been interesting.

The symposium has been far more financially successful that we imagined. The surplus will be in the order of $6000 – $7000 depending on a couple of final bills. This surplus should be dedicated to making the 4th AUUG Security Symposium even a bigger success in Sydney next year. The 2001 organisers strongly suggest that this surplus is used to under-write bringing a big name international speaker to the symposium.

Another striking success of the symposium was the number of new members that the AUUG received. Thirteen (13) new members are attributed to the symposium.

The organisers would like to thank the AUUG executive for their support. It was always a challenge to run a "big enough" symposium way up in Queensland – thanks for believing in us. In particular – thanks to Liz Carroll for all her assistance and advice.

Gary Gaskell and Warren Toomey
Co-chairs 2001

PS. Adrian Close and Pauline van Winsen offered to help organise the next security symposium. We suggest the exec sets the date ASAP and appoints the symposium committee, as long range planning is important to success.

Quantum Computing: Interview with Bruno Marchal

RB – Please present yourself.

Bruno Marchal – I am a mathematician from Brussels University. I got a PhD in Computer Science at the French University of Lille (France) bearing on a computationalist approach of the so-called mind-body problem.

RB – How did you get involved in quantum computing?

BM – Er... This is a rather long story! My initial question, which I made public in 1963 (at school) was "How long lives an amoeba?". You know amoeba divides themselves each days, so the question was "does an amoeba survive self-replication?". Molecular Biology gives the feeling that an amoeba survives, and that amoeba and us are just "mechanical device", but Biochemistry gives the feeling that it's not so obvious that we are 'mere' machine.

In 1971 I discover Godel's theorem, which helps me to choose between biology and chemistry. In some sense Biology wins, under an abstract form though, and I decide to do mathematics. I begun to realise that if we were "mere" machine then chemistry should be derivable from that abstract biology. A molecular orbital would be a map of machine accessible possible worlds. I maked it precise by deriving a sort of indeterminism and non-locality from Godel-like incompleteness results. This entails also that we should be able to put the "parallelism" of those "worlds/states" in the experimental realm. That's what happened with QM, especially through the

Einstein, Podolski, Rosen (1935) and then Bell 1964, and Aspect experimental works on non-locality in 1980. David Deutsch, in 1985, defined the concept of universal quantum computer, and did show that those hidden realities could be indirectly used to perform more powerful computations than a classical Universal Machine could do with equivalent amount of energy/time, so that those dreamy relative realities does no more belong to the speculation realm.

My work explains why matters should behave in some weird way, but I have not foreseen that the weirdness could be used to speed up computations and to provide new sort of communication like quantum teleportation.

RB – what are the advantages and the perspectives of quantum computing?

BM – It has not been immediately clear that a quantum computer offers really special advantage until the work of Simon and Shor shows that such a machine could factorise natural number n in polynomial time. Note the algorithm has been recently implemented (NATURE december 2001) and tested on the number 15 (!).

Nobody knows a classical algorithm capable of such time-polynomial factorisation, and this has been a real shock for the computing people. Grover showed also how to search an unstructured database more rapidly. Such quantum machine can generate truly random oracle. To sum up it seems that quantum computer could execute much more quickly some computation and provide new sort of resource.

Feynman also foresees that such computer would be able to simulate efficiently quantum phenomena. There are good reason (but still no proof) that quantum phenomena cannot be simulate efficiently on a classical computer. Since then we get surprising results every month in quantum communication, quantum information, quantum computing, but also quantum games and strategies. The number of publications grow in all countries.

Like what happened with classical computations, there is an explosion of idea for exploiting the weirdest quantum feature. The discovery of quantum information is really the discovery of a radically new sort of ressource for computation, communication, cryptography, game, strategy, etc.

RB – When can we hope to see an quantum computer used in a realworld situation?

BM – That is an hard question (especially for a theory minded). We must wait for more progress in nanoscience, mesoscopic physics, new materials, etc. Little quantum circuits have been implemented with ions trap, optical systems, nuclear magnetic resonance (NMR). Although the linear superposition needed in quantum computations are terribly sensible to the environment, such decoherence can be compensate by quantum error coding technics, and by building fault tolerant systems. New sort of fault tolerant technics based on quantum field theory are also emerging, like (optical) holonomic computer. New use of older technics keep appearing. For exemple teleportation appears as a real key for robust communication in the quantum circuit, even capable to be enhanced by the environment! A lot of surprising results show the beast will be done. Yet it is still difficult to predict exactly when, but it can be in 10, 30 or 50 years. Soon it will be like electricity, everywhere.

RB – What are you working on right now?

BM – I work on a new formulation of the "physics/machine psychology reversal" result – (see http://iridia.ulb.ac.be/~marchal). The new formulation single out new relations between classical computing and quantum computing. These two sorts of universal machines are much more intertwined than it's usually thought. I teach also classical and quantum computing and philosophy.

This article is re-printed with permission. The originals can be found at:

http://www.fosdem.org/interviews/1614.html

TeraServer: Build a large, cheap Linux file-server

Mark Kilmartin

Background:

We wanted a large >= 1TB file server mostly to store backups.

Well I looked at some commercial options and most of them came in at about $20000, and lacked the ability to be easily expanded. The only option to expand a lot of the commercial options was to add another unit and split your data across two of them.

It was the lack of easy expandability that actually won the day. A lot of the commercial options are very similar to what I build, a lot of them even use the 3ware cards which I was considering.

Well at about this time I noticed an article mentioned Slashdot titled "Build a Terabyte file server for under $5000". Excellent!!

Planning:

Controller:

I was well into planning the file server following the same plan as mentioned on Slashdot when I ran into the first problem 3ware (http://www.3ware.com/) who make the IDE RAID controllers had announced they were no longer going to produce them (thankfully they have reconsidered this move). Well this was no good – what would I do if any of them ever failed?

It was around then that I noticed a new comment on the original article which mention the IDEPlex (http://www.unicoretech.com/ideplex/index.html) made by Alcita (who are now called Unicore Technologies -- http://www.unicoretech.com/).

These looked perfect, they would even allow the box to be greatly expand at a later stage which was one of the down sides of the 3ware cards. These devices allowed you to plug up to 8 IDE devices into it it would then make each device show up as a different LUN on a SCSI ID. This method would allow for 56 drives to be attached to each SCSI channel. (8 devices per ID, and up to 7 devices on the SCSI channel)

The one and only downside to the IDEPlexs that I have so far encountered is the fact that they only allow transfer rates of up to 20MB/s but since these boxes were almost always going to be accessed across the network I didn't feel this was really a problem.

Drives:

This comes down to Performance or price.

For performance I would have gone for SCSI drives but with a 73GB SCSI drive coming in near £1000 and a 100GB IDE drive for about £200 (I was buying a fair number of them).

For this system the capacity was the main factor and since the IDE drives had a higher capacity this swung it to IDE.

Well the IDEPlexs won't support the newer ATA133 drives which would have allowed me to use 160GB disks.

The highest capacity ATA100 drives available at the time was 100GB (IBM and Maxtor now have a 120GB ATA100 drive available which you could probably get for about the same price.) I decided to go with Maxtor (http://www.linux.ie/articles/teraserver/www.maxtor.com) drives due to having good experiences with them in the past.

Case:

I knew the case would have to take at least 8 drives. And I would preferred to find a case to hold 16. It was after some searching that I came across the IPC-C4DE (http://www.pcicase.co.uk/pccases.ihtml?pid=205&step=4) case as sold by PCIcase (http://www.linux.ie/articles/teraserver/www.pcicase.co.uk).

This was a monster of a case.

It could hold 16 IDE or SCSI (an option when ordering) drives as well as space for a CD-ROM and Floppy drive.

Admittedly the CD-ROM and floppy had to be of the slim variety but this was not a problem as PCIcase could supply this with the case.

Also the case supported 13x12 inch Motherboards so I was not limited in my choice of motherboard. The case also had 3x225W power supplies configured in a N+1 hot swap configuration.

This picture only shows one power connector but I happily found that the actual case has two power connectors for fault tolerance.

This is not going to be a nice quite computer that you can leave beside you desk.

There is total 11 fans.

- 3 x 80mm fans in front cover.
- 3 x 120mm fans behind the drives.
- 3 x 40mm fans in the PSU
- 2 x 60mm fans on the CPUs

I can already hear some of you think "my god but what size is this case".

Well the case comes in at a respectable 4U. It is no deeper than any other Rack-mounting case I have seen. And in fact when it was fitted it was smaller than some of the DELL PowerEdge servers which it is sitting beside.

Motherboard:

This is one place where money could easily be saved. I went for a Supermicro Super P3TDE6-G this is a massive 12x13 inch motherboard.

It holds two Pentium III CPUS and takes ECC RAM. It also has built in network card and also two 160MB/s SCSI channels.

CPU:

For the CPUs we decided to fit this with two Pentium III 1GHz CPUs. This was for the simple fact that we wanted the box to be usable for other tasks whenever needed.

Memory:

Well we had no choice but to use ECC RAM since the motherboard required it. We fitted 1GB of RAM this would allow for a fair amount of caching of data and also allow it to have enough memory to run other tasks.

Other Hardware:

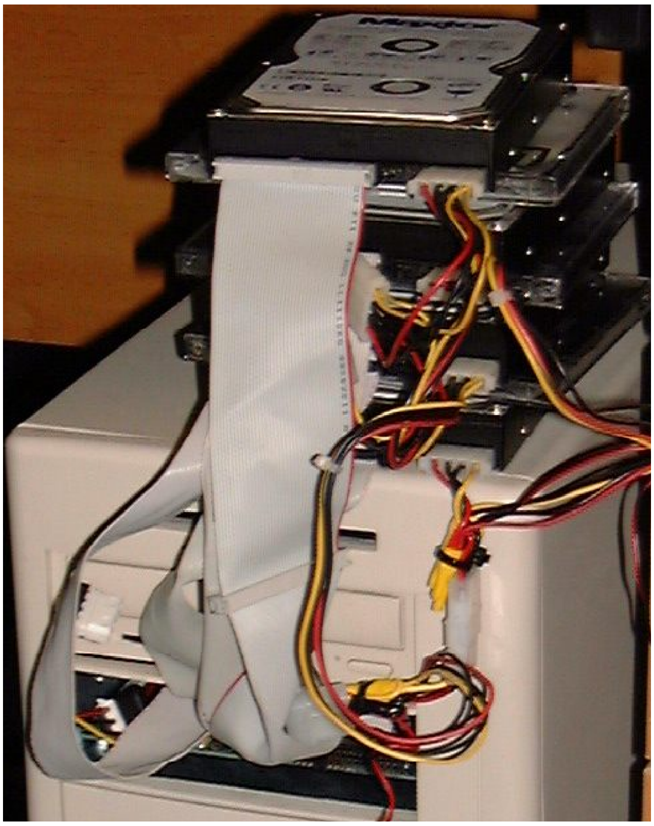
The motherboards came shipped with video cards designed for 2U slim cases so I had to get some basic video cards for the server, The SCSI and network card was already on board so I didn't have to worry about them. I needed a second network card since the box would be dual homed, for this I choose a fairly basic 3COM server NIC.

Building:

Well now that we have decided on the parts to use lets get down to building the boxes. I'll mostly be dealing with how I build them but will try to suggest ways of doing things for if you are building the even cheaper version I mentioned above.

Parts:

Well all the parts soon began to arrive. First was an IDEPlex and a couple of drives just to make sure everything worked. Here is a picture of four drives connected up to my workstation. The IDEPlex is hidden inside the open drive bay.



Well everything worked fine so time to order everything. I had a surprise when I went back to the supplier off which I purchased the four drives for testing. This supplier who will remain nameless (They have lovely special offers every day) informed me that they only had 30 in stock (I needed 50) and that he could only let me have 20 of them and there was no way he could get me more due to Maxtor not suppling them.

Well one distributor down I began calling around and looking for anybody who stocked the 100GB drives(there is not a massive demand for these apparently) well I struck pay dirt with Ahead computers when I asked about for 50 drives I was told that they didn't have them in stock but could get them in a couple of days no problem.

In the end I purchased the guts of three of these computers from these fine people.

Well after about a week some very large boxes started arriving.

Building:

Well the first task I had was to fit the hard-drives into the caddies that would hold them in the cases.

Second was to start fitting everything into the cases. This is where I first started to think "God will everything fit".

I had planned on fitting the IDEplexs on the side of the cases in some way but quickly realized that there was no way this was going to work as there just was not enough space.

The cases come with screw down fasteners for holding big PCI cards in place and also have guides at the back for holding full length cards in place. I then decided that it may be possible to hold the IDEplexs using the mounting designed for the cards. The IDEplexs come in mounting brackets to fit into a 5 1/4 inch bay. The first step was to cut notches in the card guides located at the back of the case.

There are about 7 guides but they are all located over to one side of the case in line with the PCI slots. Well this was no good for me I had to have them spread out. Well thankfully they are removable and there are mounting holes for them across the full width of the case. So using a wire snips a small groove was cut at the right heights.

The screw down holders for the cards were then removed there were two long holders and five smalled holders. I cut groves in the two larger holders and put them back in appropriate positions to hold the IDEplexs. The purpose of these was to actually hold the cards up and to put some forward pressure on them to hold them into the grooves on the card guides. I then used some of the short holders unmodified on the sides of the mounting brackets of the IDEplexs to keep them pushed toward the card guides. This resulted in a very steady and very strong mounting. It also meant that the cards could be removed by only loosing five thumb screws.

Cabling:

Now came the fun part actually fitting all the cables. First step was the strip the cases a bit.

1. Remove the three large 120mm fans mounted behind the drives. This involved undoing a thumb screw for each fan and simply unplugging it.
2. Once the fans are out you can remove the fan holder which involves removing two screws on either side of the case.

This now allowed for easy access to the back of the drives.

First I slightly tidied up the power cables. This simply involved using a few cable ties to make sure the cables ran neatly. Next was making the IDE cables. I chose to make my own IDE cables instead of buying premade cables. This had the following advantages.

Most of the IDE cables you buy have space for two IDE devices and since I was only going to be attaching one device per channel this would be a massive waste.

Also the unused connectors would be taking up space which I could not afford since space was going to be a little tight. Also the cables would most likely end up being too long so I would have to tuck the excess cable away some where again this would have taken up a lot of space.

When I got the cases I realized that I could not simply run the cables straight up from the drives as this would greatly affect the airflow around the drives and with 16 drives packed closely together I didn't want this. So I decided I would have to run the cable horizontally across the case and then vertically at a point where it would not interfere with the airflow.

The only problem with this there was not enough space between the drives and the fans to run horizontal cable so to get around this I decided to split each cable in to half (Giving 20 strands in each cable). This was then ran horizontally from the drives to the space between drives where it went vertically up to the top of the case.

I used cable ties to group the cable together for neatness and so they were not likely to move around over time and block the airflow. Normally I would frown at using cables ties on IDE cables since it makes it harder to remove drives and replace them but since all drives are in caddies this is not a problem.

Next the fan mounting and the fans are re-fitted.

Motherboard:

All that was left now was to fit the motherboard and actually connect everything up.

After the motherboard is fitted. All other cables are connected up. The power and reset buttons, the IDE cable for the CD-ROM the floppy drive cable and the SCSI cable for the IDEplexs.

After this the IDEplexs them selves are fitted in place. Next the IDE cables from the drives are connected to the IDEplexs.

Next the SCSI from the motherboard to the IDEplexs. And finally the CD-ROM and floppy are connected.

Software:

OK, the moment of truth. Would everything actually work?

Well after power was applied the first change to make was to enter the SCSI BIOS and enable the SCSI card to see all LUNS this is not the default on Adaptec cards.

OK next came the trusty Linuxcare BBC and Debian slink was started to install.

The first thing that I noticed was that the debian install could only see two SCSI drives hda and hdb.
I presumed that the Kernel on the CD was not compiled with ability to address different LUNs or was simply too old. To check this I rebooted with the ILUG BBC and all 16 drives were visible.

Well back to the debian install, I installed on sda. Next I did a dist-upgrade to woody and set about installing a new kernel.

When I was finished all this and rebooted I gratified to see that all drives were visible.

Next to decide on how to actually partition up the drives the scheme I came up with was to partition the first 8 drives with the first partition being 1G and the second one being the remaining drive.

Two of the 1G partitions were to be set up for software mirroring to hold the root partition.

The remaining 6 1G partitions were set up as Swap space specifying the same priority on these cause Linux to stripe across them.

The remainder of the 8 drives were configured in Software RAID5.

The other 8 drives were fully used for Software RAID5.

The two RAID5 arrays were them used for LVM volumes.

ReiserFS is the FS of choice. I really recommend using a journaled file-system on one of these boxes. I really don't want to sit there waiting for over 1TB of that to be fscked.

In the end I have just short of 1.3TB of space. I can hear some of you saying but there are 16 100GB drives that should be 1.6TB.

Unfortunately hard-drives manufactures use 1000000000 bytes as the definition of 1Gigabyte where as the true definition of 1GB is more like 1073741824 bytes (1024 x 1024 x 1024) so in reality each drive is about 93GB and in turn I loose the capacity of two of these drives to the parity in the RAID5 arrays.

EXPANDABILITY:

To expand these boxes simply involves putting a number of drives in another box along with an IDEPlex. the SCSI from the IDEPlex is then brought out to the back of the box where you simple connect it to the original server using a standard external SCSI cable.

Just make sure you terminate the SCSI cable. The best setup for this is to have tow connections on the back of the box running into the case and connect one of these to the original box and either daisy chain more devices onto the other connector or put a terminator on it.

PROBLEMS.

1. When you buy 54 hard-drives in total you will have a few drives which simply won't work, but since I had bought a few spare to have in case of failures I was able to work around the few failures.
2. Re-syncing the RAID5 arrays when they are first created takes almost 2 days, as far as I know this will also be about the time to rebuild a failed drive.
3. Backups. Well backing up over a TB of data would require a lot of tapes, and since one of the reasons we were going to be using these boxes was to cut down on the number of tapes we would be using. The only solution to this was to build another box and house it off-site. We would also be doing monthly backups to tape.

This article is re-printed with permission. The originals can be found at:

http://www.linux.ie/articles/teraserver/index.php

Why Gnutella Can't Scale. No, Really.

Jordan Ritter <jpr5@darkridge.com>

Please note that this paper was first released in February of 2001.

FORWARD

In the spring of 2000, when Gnutella was a hot topic on everyone's mind, a concerned few of us in the open-source community just sat back and shook our heads. Something just wasn't right. Any competent network engineer that observed a running gnutella application would tell you, through simple empirical observation alone, that the application was an incredible burden on modern networks and would probably never scale. I myself was just stupefied at the gross abuse of my limited bandwidth, and that was just DSL -- god help the dialup folks! We wondered to ourselves, Is no one paying attention, was no one bothered?

That summer we all saw a rush of press on Gnutella, and the rumour mill started churning. Most stories covering Gnutella were grossly and inappropriately evangelistic, praising the not-yet-analyzed Gnutella as a technology capable of delivering on wildly fantastic promises of fully distributed, undeterrable, unstoppable, larger-than-life file sharing on the grandest scale. Many folks were convinced that Gnutella was the next generation Napster. Gene Kan, the first to spearhead the Gnutella evangelistic movement, claimed in one early interview: "Gnutella is going to kick Napster in the pants." Later Kan admitted "Gnutella isn't perfect", but still went on to say that "there's no huge glaring thing missing". Well, something just wasn't right, and though we couldn't see it, it did seem pretty glaring.

We all understood the excitement. Herein was a technology that could potentially prove the true magnitude of Metcalfe's Law. That realization evoked nothing short of the phrase "holy shit!". But what I couldn't understand was why no one was questioning the legitimacy of these claims. For several months the only analyses anyone heard of practical implementations were generalizations and speculative comments, without much scientific or mathematical basis.

So I quickly got fed up, and resolved to write a research paper. Sometime in late March, I had begun analyzing the network structure of the Gnutella system, trying to find a way to gauge the capacity of a GnutellaNet in generalized terms, and to predict its realistic limits. What later resulted was a set of mathematical equations that could describe reachability, capacity, and bandwidth throughput. I then fed those equations into Mathematica to produce 3−D plots depicting, much to my own satisfaction, visual realizations of exactly what didn't make sense.

At about the same time, a fellow colleague in the security industry wrote a short paper detailing the various and flagrant insecurities inherent in this particular implementation of a distributed system. Seth McGann's security advisory titled Self−Replication Using Gnutella (http://www.securityfocus.com/templates/archive.pike?list=1&mid=59387) centered on the characteristics an Internet Worm inside a GnutellaNet could thrive from, and also touched on a few other flaws that would be useful to an attacker. His advisory posted in May of 2000, and unfortunately went mostly unnoticed (or misunderstood, because of its technical nature).

Later in August, Xerox PARC published a research paper (http://www.parc.xerox.com/istl/groups/iea/papers/gnutella) on the social characteristics of a GnutellaNet, proving through empirical observation that transience hurts this type of fully distributed network considerably, and that Gnutella was not such an invincible proposition after all.

These days the Internet doesn't lack for useful papers on Gnutella. Research papers (http://dss.clip2.com/articles.html) by the folks at Distributed Search Solutions are fairly high in quality and remain objective, if not optimistic about the future of Gnutella. Other informative articles persist on O'Reilly's P2P Website (http://www.openp2p.com/), and elsewhere.

So where's my paper, and why haven't you seen it? Well, in case you didn't know, I'm one of the founding developers of Napster, and for several good reasons, including the sobering fact that I was one of the leaders of the main competitor, I did not release my material to the public. Several times I resigned myself to re−writing my paper to accommodate the release of new information and analyses, but I never finished. Now I regret having sat on this for so long, for every paper on Gnutella that has come out in the last year has served as nothing but vindication of my conclusion from so early on: Gnutella will never scale.

Following is what remains of my paper, hacked up, sliced, diced and re−written. The information and analyses are still useful, but as I just said, the conclusions are the same. This paper simply proves those conclusions through mathematics.

Onward, Through the Fog

This paper assumes a working knowledge of Gnutella networks and internals, and therefore uses terminology and phraseage specific to Gnutella. If the wording seems somewhat strange or foreign to you, please stop reading this paper and seek other documentation before proceeding. Furthermore, explanation of the accompanying math is intentionally terse. Every effort has been made to verify the accuracy of the equations herein, but this discussion is intentionally limited to that which is solely relevant to Gnutella in order to keep at a minimum any distraction from an already complex topic.

To Scale, or Not to Scale

Scaling Gnutella will require more than just better resource management tools −− in its current incarnation Gnutella is mathematically and technologically unable to scale to a network of any reasonably large size. Following herein is a discussion focused on mathematically describing the metrics of a GnutellaNet topology, and using derived equations to interpret and visualize realistic limits of the technology. In order to keep the math as simple as possible, let's assume we're examining a relatively quiet GnutellaNet network, and dissect the flow of information one step at a time.

Variables and Equations	
P	The number of users connected to the GnutellaNet.
N	The number of connections held open to other servents in the network. In the default configuration of the original Gnutella client, this is 4.
T	Our TTL, or Time To Live, on packets. TTL's are used to age a packet and ensure that it is relayed a finite number of times before being discarded.
B	The amount of available bandwidth, or alternatively, the maximum capacity of the network transport.
f(n, x, y)	A function describing the maximum number of reachable users that are at least *x* hops away, but no more than *y* hops away. f(n, x, y) = Sum[((n−1)^(t−1))*n, t = x−>y]
g(n, t)	A function describing the maximum number of reachable users for any given *n* and *t*. g(n, t) = f(n, 1, t)
h(n, t, s)	A function describing the maximum amount of bandwidth *generated* by relaying a transmission of *s* bytes given any *n* and *t*. *Generation* is defined as the formulation and outbound delivery of data. h(n, t, s) = n*s + f(n, 1, t−1)*(n−1)*s
i(n, t, s)	A function describing the maximum amount of bandwidth *incurred* by relaying transmission of *s* bytes given any *n* and *t*. *Incurrence* is defined as the reception or transmission of data across a unique connection to a network. i(n, t, s) = (1 + f(n, 1, t−1))*n*s + f(n, t, t)*s

Early reports of Gnutella's usage claimed upwards of 2000 to 4000 users on the GnutellaNet. This is significant because these reports inaccurately implied that all 4,000 users on the GnutellaNet were reachable and searchable. The reality is that even in an ideally balanced GnutellaNet, P is never relevant to your potential reach; N and T are the only limiting factors.

Reachable Users

	T=1	*T=2*	*T=3*	*T=4*	*T=5*	*T=6*	*T=7*	*T=8*
N=2	2	4	6	8	10	12	14	16
N=3	3	9	21	45	93	189	381	765
N=4	4	16	52	160	484	1,456	4,372	**13,120**
N=5	5	25	105	425	1,705	6,825	**27,305**	**109,225**
N=6	6	36	186	936	4,686	**23,436**	**117,186**	**585,936**
N=7	7	49	301	1,813	**10,885**	**65,317**	**391,909**	**2,351,461**
N=8	8	64	456	3,200	**22,408**	**156,864**	**1,098,056**	**7,686,400**

Raising N (number of connections open) and T (number of hops) extend the number of reachable users geometrically.

Keep in mind, the above illustrates potential reach given two assumptions: the network is fully balanced, and everyone shares the same N and T.

So, the next obvious step for an intrepid and now better-informed Gnutella user is to increase N and T, so as to extend their potential reach into the GnutellaNet web. Not so fast! As your reach increases geometrically, so does the amount of bandwidth generated and incurred. Let's now move the discussion towards B.

Delving Deeper into B

Before proceeding, it is very important to understand that many assumptions must be made in order to carry out these computations. Observed characteristics of GnutellaNet topologies are simply too varying to accurately generalize. That said, I still believe that there exists a statistical mean of each characteristic in a GnutellaNet, which is to say that if I were to take a snapshot of the current topology of a public GnutellaNet I could derive an average N, T, and so forth. While potentially inaccurate as a realistic representation, these means can still produce a useful generalization.

In our discussion of B, there are really two different perspectives on how to measure the amount of bandwidth: the amount generated, and the amount incurred. This is a very important distinction to make, because knowing the amount of raw data generated is statistically useful, but understanding the bandwidth cost incurred by individual events in the network is much more important since it more realistically signifies the impact on an Internet connection. As previously stated, h(n, t, s) represents the amount of bandwidth generated by relaying a packet through the network, counting only data that is outbound to another destination. i(n ,t, s), on the other hand, counts all outbound and inbound transmissions, yielding a more accurate perspective on bandwidth usage. Let's introduce an example.

Joe Smith likes classic rock, and is desperately searching for any live recordings of The Grateful Dead. Joe loads up his Gnutella client, connects to the GnutellaNet, and executes his search, "grateful dead live". What actually happens?

Search Query Packet Makeup

IP header	**20 bytes**
TCP header	**20 bytes**
Gnutella header	**23 bytes**
Minimum Speed	**1 byte**
Search string	**18 bytes + 1 byte** *(trailing null)*
Total:	**83 bytes**

It isn't useful to account for Data Link Layer transmissions since they vary widely and don't significantly impact these calculations, so they have been intentionally ommitted.

IP and TCP header calculations assume simplest case scenario.

Joe's search request results in an 83 byte data packet. Initially, everyone would agree that it looks like a tiny, unnoticeable amount of data. Let's take a look at the bandwidth cost of simply relaying the search request. h(n, t, s) is comprised of the data Joe transmits across his connections to other Gnutella users (n*s), plus transmissions of all tiers between Joe and the last tier, which is only receiving.

Bandwidth Generated in Bytes (S=83)

	T=1	*T=2*	*T=3*	*T=4*	*T=5*	*T=6*	*T=7*	*T=8*
N=2	166	332	498	664	830	996	1,162	1,328
N=3	249	747	1,743	3,735	7,719	15,687	31,623	63,495
N=4	332	1,328	4,316	13,280	40,172	120,848	362,876	1,088,960
N=5	415	2,075	8,715	35,275	141,515	566,475	2,266,315	**9,065,675**
N=6	498	2,988	15,438	77,688	388,938	1,945,188	**9,726,438**	**48,632,688**
N=7	581	4,067	24,983	150,479	903,455	**5,421,311**	**32,528,447**	**195,171,263**
N=8	664	5,312	37,848	265,600	1,859,864	**13,019,712**	**91,138,648**	**637,971,200**

From above, given a concurrent demographic comparable to Napster (assuming equally balanced), searching for a simple 18 byte string "grateful dead live" unleashes 90 megabytes worth of data to be transmitted.

Even so, I don't consider h(n, t, s) to be the best measure. Let's now look at i(n, t, s), which is comprised of the originating transmission, 1 reception and N-1 transmission for tiers 1 through T-1, and 1 reception for the last tier.

Bandwidth Incurred in Bytes (S=83)

	T=1	*T=2*	*T=3*	*T=4*	*T=5*	*T=6*	*T=7*	*T=8*
N=2	332	664	996	1,328	1,660	1,992	2,324	2,656
N=3	498	1,494	3,486	7,470	15,438	31,374	63,246	126,990
N=4	664	2,656	8,632	26,560	80,344	241,696	725,752	2,177,920
N=5	830	4,150	17,430	70,550	283,030	1,132,950	4,532,630	**18,131,350**
N=6	996	5,976	30,876	155,376	777,876	3,890,376	**19,452,876**	**97,265,376**
N=7	1,162	8,134	49,966	300,958	1,806,910	**10,842,622**	**65,056,894**	**390,342,526**

N=8	1,328	10,624	75,696	531,200	3,719,728	**26,039,424**	**182,277,296**	**1,275,942,400**

i(n, t, s) has the unique property of representing double ***h(n, t, s)***.

From above, a whopping 1.2 gigabytes of aggregate data could potentially cross everyone's networks, just to relay an 18 byte search query. This is of course where Gnutella suffers greatly from being fully distributed.

Also, let's not forget that there is no consideration of time in this set of calculations. In the average case, 1.2 gigabytes worth of data takes a very long time to generate and propagate through the Internet. However, even in more realistic cases, propagating a few megabytes worth of data through several hundred thousand nodes across the Internet still takes a considerable amount of time.

At this point, though, our exercise is still incomplete. What percentage of Gnutella clients share content? Of them, what percentage are likely to respond to Joe's query? And of those, what would be the mean number of responses, and their mean length?

The Anatomy of a Firestorm

This is where we'll begin to see generalizations diverging from reality. Still though, let's take a quick gander at what evangelists thought Gnutella would be capable of. For this, we'll need to introduce a few more variables and equations.

	More Variables and Equations
a	Mean percentage of users who typically share content.
b	Mean percentage of users who typically have responses to search queries.
r	Mean number of search responses the typical respondent offers.
l	Mean length of search responses the typical respondent offers.
R	A function representing the *Response Factor*, a constant value that describes the product of the percentage of users responding and the amount of data generated by each user. ***R = (a*b) * (88 + r*(10 + l))***
j(n, T, R)	A function describing the amount of data generated in response to a search query by tier ***T***, given any ***n*** and *Response Factor* ***R***. ***j(n, T, R) = f(n, T, T) * R***
k(n, t, R)	A function decsribing the maximum amount of bandwidth *generated* in response to a search query, including relayed data, given any ***n*** and ***t*** and Response Factor ***R***. ***k(n, t, R) = Sum[j(n, T, R) * T, T = 1−>t]***

Assuming that a mean exists for the characteristics of our measurement makes these calculations much simpler. That said, recall that I don't believe this assumption to be false; that at any given moment there does exist some measurable a, b, r and l. Let's assume conservative estimates for now, and apply observed behaviour from other reports later.

The difficulty in gauging the sheer amount of data coming back to us stems from our inability to realistically discern where in the partial mesh of connections the data is coming from. By design, the only thing we will know about about the packets received is the (hopefully) unique message ID. If the message ID correlates to the message ID of one of our pending queries, the response is ours. Otherwise, the response is someone else's traffic, and if it correlates to an known ID in our routing table, it is simply passed along.

Search Response Packet Makeup	
IP header	**20 bytes**
TCP header	**20 bytes**
Gnutella header	**23 bytes**
Number of hits	**1 byte**
Port	**1 byte**
IP Address	**4 bytes**
Speed	**3 bytes**
Result Set	***r ** (8 + *l* + *2*) bytes**
Servent Identifier	**16 bytes**
Total:	**88 + *r**(10 + *l*) bytes**

Let's take a look now at what the variation of N and T yields in terms of bandwidth costs. For our first case, let's choose some reasonable values: a = 30%, b = 50%, r = 5 and l = 40, or R = 50.7.

Bandwidth Generated in Bytes (R=50.7)

	T=1	***T=2***	***T=3***	***T=4***	***T=5***	***T=6***	***T=7***	***T=8***
N=2								
N=3	152.1	760.5	2,585.7	7,452.9	19,620.9	48,824.1	116,965	272,715
N=4	202.8	1,419.6	6,895.2	28,797.6	110,932	496,614	1,441,500	4,989,690
N=5	253.5	2,281.5	14,449.5	79,345.5	403,826	1,961,330	**9,229,680**	**42,456,400**
N=6	304.2	3,346.2	26,161.2	178,261	1,128,890	**6,832,640**	**40,104,500**	**230,230,000**
N=7	354.9	4,613.7	42,942.9	349,577	2,649,330	**19,207,500**	**135,115,000**	**929,909,000**
N=8	405.6	6,084	65,707.2	622,190	**5,491,420**	**46,392,900**	**380,422,000**	**3,052,650,000**

Precision is limited to 6 or less digits; sorry, I don't know how to make mathematica behave differently in this case.

With 30% of Gnutella users sharing, and only half of them responding, the standard client settings yield over 14MB of return responses. I believe this particular R value to be near reality as far as percentages are concerned, but r and l are probably conservative, given recent reports by Clip2 DSS and others. Let's raise R a bit, here's R = 72.

Bandwidth Generated in Bytes (R=72)

	T=1	***T=2***	***T=3***	***T=4***	***T=5***	***T=6***	***T=7***	***T=8***
N=2								
N=3	216	1,080	3,672	10,584	27,864	69,336	166,104	387,288
N=4	288	2,016	9,792	40,896	157,536	577,440	2,047,104	**7,085,952**
N=5	360	3,240	20,520	112,680	573,480	2,785,320	**13,107,240**	**60,293,160**
N=6	432	4,752	37,152	253,152	1,603,152	**9,703,152**	**56,953,152**	**326,953,152**
N=7	504	6,552	60,984	496,440	3,762,360	**27,276,984**	**191,879,352**	**1,320,581,304**
N=8	576	8,640	93,312	883,584	**7,798,464**	**65,883,456**	**540,244,224**	**4,335,130,368**

These different values don't appear to have much of an impact on the overall bottom line; just over 13MB of traffic generated in response with standard client settings. Let's take one more look and adjust some of the values: a = 30%, b = 40%, r = 10 and l = 60, or R = 94.56. I believe this R to be the most realistic.

Bandwidth Generated in Bytes (R=94.56)

	T=1	*T=2*	*T=3*	*T=4*	*T=5*	*T=6*	*T=7*	*T=8*
N=2								
N=3	283.68	1,418.4	4,822.56	13,900.3	36,594.7	91,061.3	218,150	508.638
N=4	378.24	2,647.68	12,860.2	53,710.1	206,897	758,371	2,688,530	**9,306,220**
N=5	472.8	4,255.2	26,949.6	147,986	753,170	3,658,050	**17,214,200**	**79,185,000**
N=6	567.36	6,240.96	48,793	332,473	2,105,470	**12,743,500**	**74,798,500**	**429,398,000**
N=7	661.92	8,604.96	80,092.3	651,991	4,941,123	**35,823,800**	**252,002,000**	**1,734,360,000**
N=8	756.48	11,347.2	122,550	1,160,440	**10,242,000**	**86,526,900**	**709,521,000**	**5,693,470,000**

Standard client settings yield a whopping 17MB generated in response to Joe's search query.

Bringing it all together
So, now that we have all the pieces to the puzzle, let's fit them together. How much aggregate data, including request and response, is generated by Joe's search for "grateful dead live"? Let's intersect h(n, t, s) with k(n, t, R) to get The Big Picture.

Bandwidth Generated in Bytes (S=83, R=94.56)

	T=1	*T=2*	*T=3*	*T=4*	*T=5*	*T=6*	*T=7*	*T=8*
N=2								
N=3	532.68	2,165.4	6,565.56	17,635.3	44,313.7	106,748	249,773	572,133
N=4	710.24	3,975.68	17,176.2	66,990.1	247,069	879,219	3,051,410	**10,395,200**
N=5	887.8	6330.2	35,664.6	183,261	894,685	4,224,530	**19,480,500**	**88,250,700**
N=6	1,065.36	9,228.96	64,231	410,161	2,494,410	**14,688,700**	**84,524,900**	**478,031,000**
N=7	1,242.92	12,672	105,075	802,470	**5,844,690**	**31,245,100**	**284,530,000**	**1,929,530,000**
N=8	1,420.48	16,659.2	160,398	1,426,040	**12,101,800**	**99,546,700**	**800,659,000**	**6,331,440,000**

The Big Picture, h(n, t, s) and k(n, t, R) combined.

What's really stunning about the above table is the stark realization that in supporting numbers of users comparable to Napster, Gnutella would generate more than an unbelievably significant 800MB worth of data for just one of those users to search the entire network for "grateful dead live" and receive responses.

Our job is still not finished yet, though. What remains is to apply these statistics to observed query rates to gain an understanding of the real-time impact of a GnutellaNet on a network.

Behold, The Firestorm
When Napster, Inc. was served with an injunction designed to halt all file-sharing service through the Napster network, Gnutella and similar services experienced what is now commonly referred to as the "Napster Flood". While an inordinate number of users perceived the injunction as their personal charge to download from Napster as much as possible before the service was brought down, still a great many flocked to other file-sharing services such as Gnutella.

During this period of time, Clip2 DSS observed query rates peaking at 10 queries per second, double the normal 3-5 per second. The possibility of exceeding 10 qps during periods of heavy usage these days is not unlikely.

The final item of interest in this paper is the extrapolation of bandwidth rates (per second) from the bandwidth costs calculated above and observed rates. For thoroughness, query rates for a quiet (3qps), normal (5 qps), and burdened (10 qps) GnutellaNet are examined. For each test case, the main assumption is that Joe Smith's behaviour satisfies the typical user demographic.

Bandwidth rates for 3 qps (S=83, R=94.56)

	T=1	*T=2*	*T=3*	*T=4*	*T=5*	*T=6*	*T=7*	*T=8*
N=2								
N=3	1.6KBps	6.5KBps	19.7KBps	52.9KBps	132.9KBps	320.2KBps	749.3KBps	**1.7MBps**
N=4	2.1KBps	11.9KBps	51.5KBps	201KBps	741KBps	**2.6MBps**	**9.1MBps**	**31.2MBps**
N=5	2.7KBps	19KBps	107KBps	548.8KBps	**2.7MBps**	**12.7MBps**	**58.4MBps**	**264MBps**
N=6	3.2KBps	27.7KBps	192.7KBps	**1.2MBps**	**7.5MBps**	**44.1MBps**	**253.6MBps**	**1.4GBps**
N=7	3.7KBps	38.1KBps	315.2KBps	**2.4MBps**	**17.5MBps**	**123.7MBps**	**853.6MBps**	**5.8GBps**
N=8	4.2KBps	50KBps	481.2KBps	**4.3MBps**	**36.3MBps**	**298.6MBps**	**2.4GBps**	**19GBps**

Bandwidth rates for 5 qps (S=83, R=94.56)

	T=1	*T=2*	*T=3*	*T=4*	*T=5*	*T=6*	*T=7*	*T=8*
N=2								
N=3	2.7KBps	10.8KBps	32.8KBps	88.1KBps	221.6KBps	533.7KBps	**1.2MBps**	**2.9MBps**
N=4	3.6KBps	19.9KBps	85.9KBps	335KBps	**1.2MBps**	**4.4MBps**	**15.3MBps**	**52MBps**
N=5	4.4KBps	31.7KBps	178.3KBps	916.3KBps	**4.5MBps**	**21.1MBps**	**97.4MBps**	**441.3MBps**
N=6	5.3KBps	46.1KBps	321.2KBps	**2.1MBps**	**12.5MBps**	**73.4MBps**	**422.6MBps**	**2.4GBps**
N=7	6.2KBps	63.4KBps	525.4KBps	**4MBps**	**29.2MBps**	**206.2MBps**	**1.4GBps**	**9.6GBps**
N=8	7.1KBps	83.3KBps	802KBps	**7.1MBps**	**60.5MBps**	**497.7MBps**	**4GBps**	**31.7GBps**

Bandwidth rates for 10 qps (S=83, R=94.56)

	T=1	*T=2*	*T=3*	*T=4*	*T=5*	*T=6*	*T=7*	*T=8*
N=2								
N=3	5.4KBps	21.6KBps	65.6KBps	176.2KBps	443.2KBps	**1.1MBps**	**2.4MBps**	**5.8MBps**
N=4	7.2KBps	39.8KBps	171.8KBps	670KBps	**2.4MBps**	**8.8MBps**	**30.6MBps**	**104MBps**
N=5	8.8KBps	63.4KBps	356.6KBps	**1.8MBps**	**9MBps**	**42.2MBps**	**194.8MBps**	**882.6MBps**
N=6	10.6KBps	92.2KBps	642.4KBps	**4.2MBps**	**25MBps**	**146.8MBps**	**845.2MBps**	**4.8GBps**
N=7	12.4KBps	126.8KBps	**1.1MBps**	**8MBps**	**58.4MBps**	**412.4MBps**	**2.8GBps**	**19.2GBps**
N=8	14.2KBps	166.6KBps	**1.6MBps**	**14.2MBps**	**121MBps**	**995.4MBps**	**8GBps**	**63.4GBps**

Keeping things in Perspective
From the charts above, it becomes mind-numbingly clear that the Gnutella distributed architecture is fundamentally flawed and can have a horrific impact on any network. On a slow day, a GnutellaNet would have to move 2.4 gigabytes per second in order to support numbers of users comparable to Napster. On a heavy day, 8 gigabytes per second.

A lot of potentially obscure assumptions are made here, though, and they should be carefully examined and understood before making conclusions:

- the test GnutellaNet is ideal, which is to say that all participants form a topology which conforms to g(n, t);
- being ideal, its topology is static -- meaning all responses to a search query are received by the requestor, without being cut off by transient nodes;
- query rates are constant,
- query demographics correlate to the average case

presented above,

- all GnutellaNet participants are capable of supporting the bandwidth rates incurred,
- search queries and responses represent the only relevant and bandwidth-significant activity on the GnutellaNet.

So why should the above charts be taken with a grain of salt? Well, the real GnutellaNet that exists today is certainly not ideal, and has been occasionally observed persisting as several smaller, fractured GnutellaNets. Also, there's a great deal of transience in the GnutellaNet; observations yield only roughly 30-40% of participants remain for 24 hours or more. And it should be obvious to even the most casual observer that query rates are not constant, and are more likely to burst and lull as the topology shifts and usage varies.

One important factor in evaluating the usefulness of the above is to consider the usage demographic. Current usage may show 3-5 queries per second with anywhere between 4,000 and 8,000 users, but if Gnutella were to ever grow in size, both by users and consequentially by files, search rates would likely increase dramatically. This would be for at least two reasons: more users equates to more people interested in locating content equates to more aggregate queries per second, and more content equates to wider variance in type of material equates to, quite simply, more to search for. So, applying query rates involving only thousands of users to GnutellaNet populations orders of magnitude greater in size is probably inaccurate; instead, at greater sizes, the above computed bandwidth rates are probably much too small. Indeed, one can extrapolate from the above, using the test case of 1,000,000 users:

- 8,000 users generate 5 queries per second, which simplified means
-
- 1,600 users generate 1 query per second, which then leads to
-
- 1,000,000 users / 1,600 users per query per second == 625 queries per second

Therefore it is more likely that, given an ideal GnutellaNet and a capable Internet, Gnutella would generate 625 queries per second with one million users instead of our test case of 5, which generates 4GBps worth of traffic just by itself. So how much data does a query rate of 625 qps generate? The calculation is left as a thoughtful exercise to the reader.

Most important of all, though, the above numbers assume a capable network connection exists for all participants. If networks weren't capable of relaying the amounts of traffic discussed above, traffic jams would occur and query rates would drop, query response rates would drop, and overall traffic rates, as a result, would drop. And we know they aren't capable; we know that a significant percentage of participants are dialup users, and their low bandwidth capabilities cause significant traffic congestion and topology fragmentation when improperly configured.

Conclusions

Even though many assumptions were made throughout the course of these calculations, some of which are provably unrealistic, these exercises still yield a useful perspective. In an ideal world, Gnutella is truly a "broadband killer app" in the most literal of senses -- it can easily bring the Internet infrastructure to its knees. And it should also be noted that only search query and response traffic was accounted for, omitting various other types of Gnutella traffic such as PING, PONG, and most importantly, the bandwidth costs incurred by actual file transfers. 2.4GBps is just search and response traffic, but what about the obnoxiously large amount of bandwidth necessary to transfer files between clients?

Those reading this paper should be careful to note that non-intended uses of the GnutellaNet also incur noticeable bandwidth hits: using search queries to chat with other participants, SPAM placed inside search queries and results to advertise various things, and gibberish, typically resulting from misbehaving users or clients. Futhermore, with individuals writing their own clients and protocol extensions, we may begin to see loop detection being rendered useless. Depending on how individual clients implement loop detection (comparing message ID's versus comparing message ID's + a checksum of the packet's payload), protocol extensions may interfere with legacy clients and result in more traffic than necessary being generated and relayed.

The main argument against this paper is that GnutellaNets are never ideal, and as adoption and usage grows, are statistically less likely to be ideal, given the increase in complexity of the topology as the number of participants increase. I would agree with this principle, but I believe it only serves as better proof of the premise: if an ideally distributed and fully capable network generates 2.4GBps to accomodate 1M users (and we already know this figure to be unrealistic in terms of what the modern Internet is capable of), then a poorly distributed network with insufficient bandwidth will certainly not be able to support the same number of participants or the traffic they generate. In other words, again, Gnutella can't scale.

Another key argument against these computations is that they are all focused on the center of an ideal GnutellaNet, and applying this generalization to all configurations of nodes is misleading and inaccurate. Traffic is measured and generalized from a maximizable point; this is to say that the "center" node will always generate the most amount of traffic given the same configuration throughout, whereas a leaf node in an ideal GnutellaNet generates only a fraction of that bandwidth. However, empirical analysis yields the observation that, in practice, leaf nodes don't generally have only one connection into the GnutellaNet. As a matter of fact, leaf nodes don't tend to occur naturally at all, since it is rarely in a participant's best interest to limit themselves to one connection, in maximizing bandwidth capacity versus search depth. To date I've only observed this happening on a large scale with Reflectors, or strategically placed Gnutella "proxies" at high bandwidth locations on the Internet aimed at serving dialup and other small capacity clients. So, the inaccuracy of these numbers likely lies in their being, again, much too small. Also, regardless of how intertwined and convoluted the connection paths are, the data path is effectively rendered semi-ideal through loop detection, so the methodology turns out to be more realistic than first thought.

Yet another valid question to raise against the premise is, What is a reasonable size? Is it 100 users? Is it 1,000? Or 100,000? Or 1,000,000? Nothing short of global domination? Discerning what's reasonable is assuredly a subjective comparison, however, I use the phrasage interchangably with original statements like "Gnutella will kick Napster in the pants." Common sense dictates that in order to accomplish that, Gnutella would have to perform more efficiently, scale higher, and be more capable. These exercises prove that, on a perfect level, Gnutella just can't rise to meet the challenge. Consequentially, they prove that on an imperfect level Gnutella has no hope of performing on the same level.

In the final assessment, it's painfully obvious that Gnutella needs a complete overhaul. Major architectural flaws are fundamental in nature and cannot be mitigated effectively without redesign at the most basic level. Some intelligent caching could likely benefit the Gnutella architecture, since observations yield that many searches and responses result in repetitive, duplicate transmissions. However, given the transience of GnutellaNet participants, and the wide variety of participating clients, it would be difficult to predict with any amount of accuracy how effective technology like this would be.

Various efforts claim to be underway to redesign the protocol; among them, gPulp stands out as the farthest along, with message boards and mailing lists set up for those wanting to get involved. But, with its mission of consentual changes implemented through a working group, I harbor significant doubt as to whether they will ever be timely and effective at producing an alternative. GnutellaWorld, another revamp effort recently publicized by CNet's news.com, takes the lead on the initiative for developing Gnutella2. J.C. Nicholas, apparently representing GnutellaWorld, claimed in an interview with CNet that Gnutella2 technology would be out "soon". Characterized as an "Internet Earthquake" and promised to be "the greatest revolution since Linux", Gnutella2 sounds more like the same old hype than anything else. And with only 8-9 months under their collective belt as an organization, I personally wonder how far along efforts could be. If the fact that this open-source project's CVS repository remains quite empty, or that its mailing lists appear dormant presents any indication of progress, the Internet probably has some time to go before experiencing the next internet cataclysm. Considering GnutellaWorld's intentions of supporting 20 million people or more, I can only hope that it's nothing like the original Gnutella.

This article is re-printed with permission. The originals can be found at:

http://www.darkridge.com/~jpr5/doc/gnutella.html

The Gelato Federation Team at UNSW Look at the Linux Kernel

Lucy Chubb <lucy@chubb.wattle.id.au>

[Editor's Note: Many of you will likely remember Lucy Chubb, and the contributions that she and Peter Chubb have made to AUUG over the years, as well as to low-level Unix kernel work. Here's a quick synopsis of the project that Lucy and Peter are currently undertaking at UNSW. She has promised a more in-depth piece for our next issue.]

Hewlett-Packard is partnering a number of institutions in a group known as Gelato Federation, which in addition to HP, involves the BioInformatics Institute in Singapore, Groupe ESIEE in France, the National Center for Supercomputing Applications (NCSA) in the U.S., China's Tsinghua University, University of Illinois in the U.S., the University of New South Wales, and the University of Waterloo in Canada. Each member of the group is targeting different aspects of Linux running on Itanium based systems aimed at making Linux on Itanium a platform of choice for high end computing and research.

The team at the University of New South Wales, headed up by Gernot Heiser of the School of Computer Science and Engineering, is to focus on strengthening and enhancing the Linux kernel for the Itanium. The UNSW team currently consists of Peter Chubb and Lucy Chubb. During the project we hope to describe our progress and explain some of our developments to the members of AUUG.

This is a quick look forward at the first aspect that I will be looking at. Firstly, a quick summary of the important memory management concepts that you will need.

So that programs are not restricted in size by the amount of real memory in a system, the program resides in a large virtual memory space. Only the parts of the virtual memory that have something in them (code, data, or stack) need to have some sort of storage associated with them. When the program is running, some of the storage may be real memory and some may be blocks of slower storage such as disk (the swap space).

When a program instruction uses a virtual memory address, the operating system and hardware combine to work out where the contents of that address is held and then make it accessible. If it is in real memory the virtual address is translated into a real memory address using the page tables. If the contents is in the swap space, it has to be read into real memory first before the address translation is completed.

To make virtual to real address translation faster a translation lookaside buffer (TLB) holds the real addresses of some of the memory blocks corresponding to in-memory virtual memory blocks. Addresses in the TLB can be translated without having to use the page tables, which makes it faster. If an address is not in the TLB, a TLB miss occurs. The number of entries in the TLB is limited and a large number of TLB misses can be costly in terms of performance.

One way of reducing the number of misses (increase the TLB coverage) is to make page sizes larger. The Itanium supports page sizes of 4K to 256MB, so this is feasible. The most obvious problem with increasing the page size across the whole system is that swapping has to deal with the larger pages as well. It is much more expensive to swap a 256MB page with a few dirty bytes than a 4K page with the same dirty bytes.

Initially I will look at using various sizes of pages within the same kernel, rather than a single size. The extra cost of moving page contents around should not wipe out the savings from reducing TLB misses. It may also be worth while to look at what size of page to allocate when a page is created, and when it is worth while aggregating or splitting pages.

More information on the Gelato foundation can be found on the Gelato web site: (http://www.gelato.org/). By the time this article appears the UNSW Gelato project site (http://gelato.unsw.edu.au/) should be up.

For further reading on concepts related to superpages see "Transparent Support for Superpages" Juan E Navarro, Rice University (available at http://citeseer.nj.nec.com/496959.html).

Writing Gnome Applications with Glade and Python

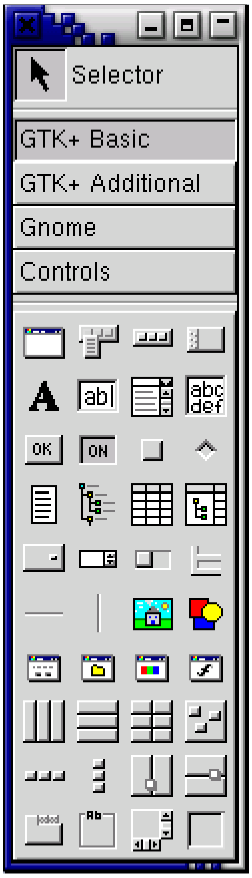
Robert Laing <zapr@icon.co.za>

If you've ever set out to learn programing hoping to dive straight into writing an ambitious application but then got scared off by all the hard ground that needed to be covered first, developing Gnome Apps using Glade and Python is for you. We'll have a fully fledged application window up faster than most programming tutorials can print "Hello World!" on a command line.

GLADE: AN OPEN SPACE IN A FOREST

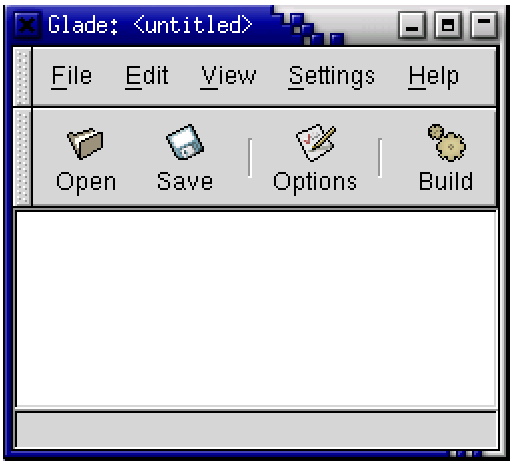
Step one is to open Glade. It's a standard feature in Gnome setups found in the "Development" sub-menu of "Programs". The following three windows pop-up on the screen:

Palette



This is the window that makes app programming with Glade as fun and easy as playing with Meccano, Lego, Barbies ... whatever construction set you liked as child. There are currently four "layers" of widgets to chose from in the palette, giving an intimidatingly large selection of pieces to get started with.

Project window



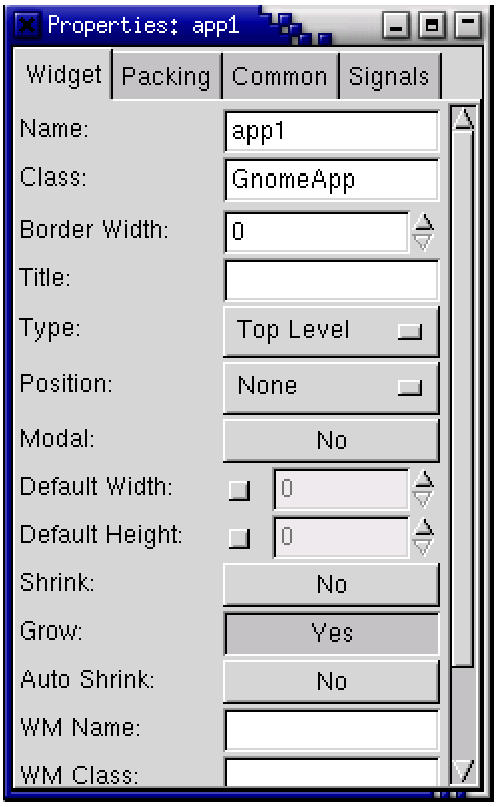
Clicking "Build" reveals that, at the time of writing, Python is conspicuously absent from the list of programming languages Glade can write its output as. But thanks to libglade, the choice of language is academic since we're not going to convert Glades's XML output into source code. While you could instruct Glade to produce C or Perl code to use as a starting point for the final application, leaving Glade's output as a separate foo.glade file holds advantages for both application developers and end users. Gooey toolkits like Gtk offer a plethora of cosmetic options. If you used Glade to create C code, the look and feel seen by end users is set in stone as the code compiles. By leaving Glade's output as data to be loaded at runtime by libglade, the end user can manipulate the application's foo.glade file to tweak things such as menu names and hot-keys to personal taste. And developers get to experiment WYSIWYG with their application's look without needing to constantly edit and recompile.

As will become clear as project1 grows, a combination of Glade and Python lets you bash out applications more artistically than traditional programing techniques tolerate. We can make incremental improvements, adding just a little to project1.glade and project1.py, to gradually feel our way to our dream app.

Properties

I've jumped ahead here and created a "Gnome Application Window' to put some interesting things in the Properties window. This is a notebook with four pages of options for every widget. The pages are titled "Widget", "Packing", "Common" and "Signals", and the bewildering number of choices found on each page is another strong argument in favour of doing this kind of development WYSIWYG.

Creating app1



Name: app1
Class: GnomeApp
Border Width: 0
Title:
Type: Top Level
Position: None
Modal: No
Default Width: □ 0
Default Height: □ 0
Shrink: No
Grow: Yes
Auto Shrink: No
WM Name:
WM Class:

Glade's toolbox opens with "GTK+ Basic". Since we're diving directly into the deep end, toggle that to "Gnome", and then click on "Gnome Application Window" which is the icon on the top left hand corner. This causes a prototype of our work in progress which looks confusingly similar to the Project Window to pop-up.

Click "Save" -- in the Project Window, not the prototype.

The default ~/Projects/project1/project1.glade is as good a directory and filename as any.

Then in the ~/Projects/project1/ directory, create the four lines of code below with your favourite text editor.

```
#!/usr/bin/env python
import gtk, gnome.ui, libglade
widget_tree1 = libglade.GladeXML
("project1.glade","app1")
gtk.mainloop ()
```

I saved my file as project1.py. Bring this application to life by entering at the command line:

```
python project1.py &
```

You'll see we've now created a fully fledged windows, icons, mouse and pull-down menu (WIMP) application. Not only do the menus open to the mouse, they also respond to Alt-f or whatever is underlined. But perspicacious users will note a few flaws. There's no "meat" in the app: just menus and bars that don't actually do anything. The "Exit" option in the File menu doesn't even work, so the only way to close the window is to use whatever kill option the windows manager provides. You'll also have to get another command line to "ps -A" and kill the python pid to get the prompt back.

We'll make a proper way to quit this app a priority, but first lets digress a little into Python style.

Dotty Python

Readers who've worked through introductory Python tutorials may be wondering why I didn't write the above program in this more familiar style:

```
#!/usr/bin/env python
from gtk import *
from gnome.ui import *
from libglade import *
widget_tree1 = GladeXML ("project1.glade","app1")
mainloop ()
```

The subtle difference between Python's "import module" and "from module import *" conventions is that the latter loads all of a module's function names into the current symbol table while the former requires programmers to address them using the syntax module.function ().

Beginner Python tutorials tend to avoid the dot convention, preferring to keep things simple by treating "add on functions" imported from modules exactly like built-in Python functions. The problem is it taxes the memories of both computers and humans. Often the only reference material for a function is the code in its module file, so the module.function() style syntax gives a hint where to start looking for help.

Programming Classes

Readers who've attempted to learn object oriented programming in other languages may be tempted to bolt at the sound of classes and methods. Take courage, one of the joys of Python is you don't have to delve deeply into the mysticism of OOP to use its techniques. Once you've acclimatised yourself to dots using the module.function(paramaters) syntax, flinging instance.method(parameters) lingo is no trouble at all.

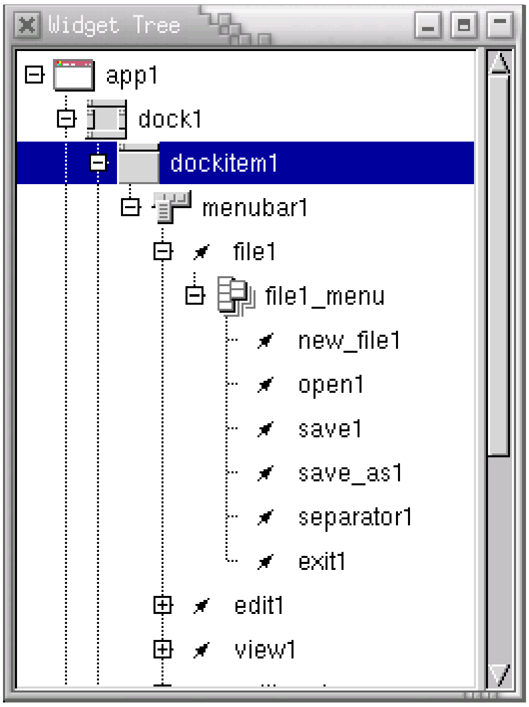
Much of application programming boils down to writing procedures which spring to life on a signal. A signal is something like a button getting clicked or a menu item selected. Glade provides a menu listing the choice of signals for the selected widget, so all we have to do is write signal handling procedures. C programmers tend to call these procedures callback functions while Python programmers tend to call them handlers .

LIBGLADE

This module file is located on my system at /usr/lib/python2.1/site-packages/libglade.py. Browsing that file reveals it defines one class, GladeXML. We've already set up an instance of libglade.GladeXML with the line

```
widget_tree1 = libglade.GladeXML
("project1.glade","app1")
```

widget_tree1 is a compound data type best explained visually. To get a diagram, return to Glade's menu window, open "View," and select "Show Widget Tree". After climbing the branches by clicking on "+" a few times, you should see something like this:



Class GladeXML needs two parameters to create an instance of itself: first the name of the file to read the data from, and second the "root" of the desired tree out of a potential forest that file might house. As the project grows, the number of trees project1.glade will multiply, and each one needs its own

```
instance_name = libglade.GladeXML("file_name",
"root_name")
```

statement to bring it to life.

The libglade.py file reveals class GladeXML has four methods:

```
signal_connect(handler_name, handler, *args)
signal_autoconnect(dict)
get_widget(name)
get_widget_by_longname(longname)
```

Readers who've read the libglade.py files will notice I've left out the first argument, self, for each of these. I'll explain why in another digression into Python style.

Parameter differences between methods and procedures

Using instance.method(parameters) is nearly the same as old fashioned function(parameters) but some things are a little weird at first, such as the vanishing first parameter of methods which is conventionally called self. If you invoke a method by adding it after a dot to your instance -- your instance being the name you've selected as the variable to hold a class -- you

treat this parameter1 "self" much as the "for internal office use only" entry in tax forms -- ie forget about it and write ***instance.method(parameter2, parameter3)***.
Python offers an alternative syntax for the same job where the first parameter doesn't vanish: ***class.method(instance, parameter2, parameter3)***. But I prefer the first style because it has a more OOP tone.

You can't ignore "self" entirely however. You need to leave it a space when you write handlers. So even if you don't intend passing any parameter to your procedure, you can't say "def procedure():" -- you need to say "def procedure(widget):" so as to have a "place holder" for self. This variable is often called widget or obj.

SIGNAL_CONNECT(HANDLER_NAME, HANDLER, *ARGS)

We'll bring this GladeXML method into play adding the following line to our program:

```
#!/usr/bin/env python
import gtk, gnome.ui, libglade
widget_tree1 = libglade.GladeXML
("project1.glade","app1")
widget_tree1.signal_connect("on_exit1_activate",
      gtk.mainquit)
gtk.mainloop ()
```

Now when you run project1.py, you'll be able to quit the program by clicking "Exit" in the "File" menu or using its pre-defined accelerator Ctl+Q.

HANDLER_NAME

The handler_name for the Exit menu item is "on_exit1_activate" because that's the default name Glade puts in its template file for GnomeApps. You could edit it into something else by selecting in Glade -- an operation which can be done by clicking on the Exit menu item in Glade's prototype application. The title in Glade's Properties Window changes to "Properties: exit1". By bringing the "Signals" folder to the top, you bring up the dialog in which you can graphically create and name signals. To get a list of handler_names Glade has already created for us, enter "grep handler project1.glade" at the command line. While the default GnomeApp has handlers ready to go for its menu items, the "New", "Open", and "Save" icon buttons in the toolbar are currently dead to the world. If you select the "New" button and then select "clicked" from the "Signal:" menu, Glade provides a default handler_name "on_button1_clicked". Since we want this button to do the same job as the "New File" menu item, we may as well give it the same handler_name, "on_new_file1_activate". The Open and Save buttons can similarly be given the same handler names as their corresponding menu items.

HANDLER

Quitting is an easy operation to add because the gtk module contains this procedure ready made, mainquit(). But note that when you call functions from signal_connect's handler parameter, you don't put in the brackets. If you attempt to run the above program with

```
widget_tree1.signal_connect("on_exit1_activate",
     gtk.mainquit())
```

you would get a "fatal error, segmentation crash" message. This doesn't mean, however, that we can't write procedures invoked by signal_connect that have parameters passed to them.

*ARGS

The "*args" nomenclature means signal_connect's paramters could look like (handler_name, handler), or (handler_name, handler, arg1), or (handler_name, handler, arg1, arg2), and so on.

To demonstrate, lets create our own handler which we'll call from either the "New" icon button or "New File" menu item since we gave the button the same handler name a few paragraphs ago. Expand project1.py as follows

```
#!/usr/bin/env python
import gtk, gnome.ui, libglade
def handler1(widget, message1):
              print message1
widget_tree1 = libglade.GladeXML
("project1.glade","app1")
widget_tree1.signal_connect("on_exit1_activate",
gtk.mainquit)
widget_tree1.signal_connect("on_new_file1_activate
"
 ,
   handler1, "Hello World!")
gtk.mainloop ()
```

Phew! I've satisfied the introductory tutorial law which says all first programs must print "Hello World!" on the command line by the end of Chapter 1. As explained above, I can pass as many arguments as I wish to a handler:

```
#!/usr/bin/env python
import gtk, gnome.ui, libglade
def handler1(widget, message1,
  message2):
              print message1
              print message2
widget_tree1 = libglade.GladeXML
("project1.glade","app1")
widget_tree1.signal_connect("on_exit1_activate",
gtk.mainquit)
widget_tree1.signal_connect("on_new_file1_activate
", handler1, "Hello World!", "Hello Again!")
gtk.mainloop ()
```

GET_WIDGET(NAME)

Since we're learning to write gooey apps, printing to the command line doesn't really pass muster. To make "Hello World!" appear in the "appbar" at the bottom of our application and get it cleared when the "Clear" menu item is selected in the "Edit"menu, expand the program as follows:

```
#!/usr/bin/env python
import gtk, gnome.ui, libglade
def handler1(widget, message):

     appbar1.push(message)
def handler2(widget):
              appbar1.pop()
widget_tree1 = libglade.GladeXML
("project1.glade","app1")
widget_tree1.signal_connect("on_exit1_activate",
gtk.mainquit)
widget_tree1.signal_connect("on_new_file1_activate
", handler1, "Hello
  World!")
widget_tree1.signal_connect("on_clear1_activate",
   handler2)
appbar1 = widget_tree1.get_widget("appbar1")
gtk.mainloop ()
```

GladeXML's method get_widget is used to create instances of widget objects. The name "appbar1" is the default name in the template file for a class called GnomeAppBar. The instance of this object can be any legal Python variable name you like. You can just use Glade's default name except when it's something like "combo-entry1". If you try that as a variable name, the Python interpreter tries to subtract two unitialised variables and then stumbles and falls. So remember to give things like "combo-entry1" variable names like combo_entry1. The methods that can be used on GnomeAppBar instances are found in the gnome.ui module, a file called /usr/lib/python2.1/site-packages/gnome/ui.py on my system. Note how Python also uses dots to keep "module packages" together: gnome.ui corresponds to the module file gnome/ui.py, gnome.xmhtml to gnome/xmhtml.py and so on.

By reading that file I discovered GnomeAppBar has a method called push which prints strings and one called pop which, ehr, pops strings. So if you click "Save" lots of times, you'll have to push clear lots of times to get the app bar blank again.

SIGNAL_AUTOCONNECT(DIC)

If we keep adding "app1.signal_connect(handler_name, handler)" lines for every menu entry, our code is going to get very cumbersome. Unless we want to pass additional arguments to a handler, we can use GladeXML's signal_autoconnect(dic) method to group all our handler_names and handlers into one data structure. The "dic" refers to a data structure called a dictionary in Python. Perl or Awk programmers may prefer to think of "dic" as an associative array. I'll call the variable containing this data structure handler_dictionary and bung in every handler_name that "grep handler project1.glade" churns up. With the exception of "on_exit1_activate" which is needed to close the program, I've made everything call a procedure called my_name. It in turns calls a function in libglade which I'll explain later:

```
#!/usr/bin/env python
import gtk, gnome.ui, libglade
def my_name(widget):

 appbar1.push(libglade.get_widget_name(widget))
handler_dictionary = {"on_new_file1_activate" :
        my_name, "on_open1_activate" : my_name,
        "on_save1_activate": my_name,
        "on_save_as1_activate": my_name,
        "on_exit1_activate": gtk.mainquit,
        "on_cut1_activate": my_name,
        "on_copy1_activate": my_name,
        "on_paste1_activate": my_name,
        "on_clear1_activate": my_name,
        "on_properties1_activate": my_name,
        "on_preferences1_activate": my_name,
        "on_about1_activate": my_name,
        "on_new_file1_activate": my_name}
widget_tree1 = libglade.GladeXML
("project1.glade","app1")
appbar1 = widget_tree1.get_widget("appbar1")
widget_tree1.signal_autoconnect(handler_dictionary
)
gtk.mainloop ()
```

WIDGET NAMES

If you run project1.py now and click on the various menu items and buttons, you'll see each one prints its name in the app bar. The name of the widget is passed to GnomeAppBar's push method as a string by libglades's get_widget_name(widget) function. Besides the GladeXML class and its four methods, the libglade module makes these three functions available:

1. get_widget_name(widget)
2. get_widget_long_name(widget)
3. get_widget_tree(widget)

But what is "widget"? The fact you don't have to state every variable's type in Python is a two edged sword for novices. Not having to "earmark" space for variables and cast their type in stone before you can use them does make programming simpler. But on the other hand, you sometimes have to do some detective work to figure out what a variable is supposed to represent. Fortunately, the language provides plenty of tools to help you do that. One of these is a built-in function type(object) which we can make "printable" using another built-in function, repr(object), whose job is to make whatever put inside its parentheses representable by print. To get and idea of what "widget" is, alter the contents of the my_name procedure to:

```
appbar1.push(repr(type(widget)))
```

Experimenting with running project1.py leads to the problem there isn't enough space in the app bar to read the string issued by repr(type(widget)). Go into Glade, select appbar1, and in the "Widget" folder of the "Properties" window toggle "Progress:" to No. Once you've saved project1.glade and re-run project1.py you'll see the appbar1.push(string) method now has all that real estate to itself. But all this reveals is that everything has the same type, "<type 'instance'>". To get more detail than that, edit the above to

```
appbar1.push(repr(widget))
```

Now the status bar will provide information like "<gtk.GtkButton instance at 0x82a7344>", revealing what class whatever we clicked is an instance of and its memory address. This information is passed to handlers when they're invoked from either the signal_connect or signal_autoconnect. Libglade's fet_widget_name(widget) and get_widget_long_name(widget) functions help use these as I'll demonstrate later.

To see the difference between widget_name and widget_long_name, alter the contents of the my_name procedure to this:

```
appbar1.push(libglade.get_widget_long_name(widget))
```

Now if you click on the "Cut" menu item, instead of just printing out "cut1" it prints out "app1.dock1.dockitem1.menubar1.edit1.edit1_menu. cut1". Glade's Widget Tree window graphically lineage involved in this name.

libglade gives us the option using long_name to create instances. Instead of

```
appbar1 = widget_tree1.get_widget("appbar1")
```

we could write

```
appbar1=
widget_tree1.get_widget_by_longname("app1.appbar1"
)
```

You might like to do that if you intend to call all your Cancel buttons "cancel" and prefer to differentiate them by the name of their parents. But I think just letting Glade number them is easier. While the need for get_widget_by_longname(widget_long_name) isn't obvious, the get_widget_long_name(widget) function is very handy as we'll see.

Now that we've played with the basics, lets move on to creating a "proper app": a Glade/Python version of the Gnome Less utility.

(Which will appear in AUUGN's next issue --Ed)

This article is re-printed with permission. The originals can be found at:

http://www.icon.co.za/~zapr/Project1.html

Press Release: Wind River Announces Transfer of FreeBSD Sponsorship to FreeBSD Mall, Inc.;

Sponsorship of Open Source Effort Returns to Roots

ALAMEDA, Calif.--(BUSINESS WIRE)--Jan. 14, 2002--Wind River Systems, Inc. (Nasdaq:WIND), a leading provider of software and services for connected devices, today announced a definitive agreement to transfer its FreeBSD operations to FreeBSD Mall, Inc. FreeBSD is an advanced open source UNIX operating system, derived from BSD UNIX and developed at the University of California. Well known for its performance and reliability, FreeBSD technology is widely used as a server operating system by many large Internet sites including Yahoo!, Hotmail, Sony Japan, Apache, Pair Networks and Whistle Communications. Like Linux, the source code for FreeBSD is freely available.

Wind River assumed stewardship for the FreeBSD open source project in May 2001 when it acquired assets of Berkeley Software Design, Inc. (BSDi), the former sponsor of FreeBSD. In addition to the FreeBSD sponsorship, Wind River acquired the proprietary BSD UNIX-based OS (BSD/OSO). Wind River's interest in the BSDi assets continues to focus on BSD/OS. Divesting the FreeBSD business further sharpens that focus and provides continuity and increased support for FreeBSD.

FreeBSD Mall Inc. is led by its founder, Bob Bruce. Bruce's involvement with FreeBSD dates back to 1993 when his former company, Walnut Creek CDROM, was the first and primary distributor of FreeBSD.

"The FreeBSD community will be well served by this transaction," said Larry Macfarlane, senior director of Wind River's Application Platforms product division. "When we decided to divest the FreeBSD business, we looked for a successor organization that could meet high standards of customer service and maintain a mutually beneficial relationship with the FreeBSD community. After carefully evaluating many interested organizations, we decided that the best way to ensure the continuity and vitality of FreeBSD was to return it to its roots."

Bob Bruce enthusiastically welcomes FreeBSD back home, and commented, "As we go forward, we will be able to build on strong relationships and friendships. We have all worked together before. FreeBSD has a promising future, and I am committed to helping it reach its full potential."

Jordan Hubbard, co-founder of the open source FreeBSD Project, also endorses this change. "I'm happy to see that the FreeBSD Mall will be continuing without interruption," said Hubbard. "Through this transfer, FreeBSD will be back under the direction of the same people who started and know the FreeBSD CD product line perhaps better than anyone." Hubbard is an engineering manager at Apple, where he helps develop the highly acclaimed Mac OS X, and works on the open source Darwin Project. Both Mac OS X and Darwin are based on FreeBSD.

FreeBSD Mall plans to aggressively promote and market FreeBSD. In addition to providing the standard FreeBSD distribution on CDROM, they will continue to offer a subscription service, snapshots of the current development branch, and published hardcopy editions of the FreeBSD Handbook. They will also offer several levels of professional support and services for FreeBSD. FreeBSD Mall has placed FreeBSD products in many mainstream retail stores, including Best Buy, CompUSA, Borders Books, Barnes & Noble and Amazon.com. In the near future, they plan to expand their retail presence, especially in Europe and Asia.

Current support obligations and software subscriptions at Wind River will be transferred to FreeBSD Mall in this transaction. All current Wind River employees working with FreeBSD will be employed by FreeBSD Mall.

Wind River will continue to develop and support BSD/OS, a professionally engineered and supported

BSD operating system widely used in embedded systems. A major upgrade of BSD/OS that will focus on meeting the needs of networked embedded devices is scheduled for release later this year. Wind River will also continue to support the BSD open source community, and plans to cooperate with the FreeBSD BSD open source community, and plans to cooperate with the FreeBSD Mall and other BSD organizations to help promote wide adoption of BSD technology.

The transaction is expected to close at the end of January. Financial details are not being disclosed at this time.

About Wind River
Wind River is a worldwide leader in integrated embedded software solutions for creating reliable and innovative connected devices. Wind River provides development tools, real-time operating systems, and advanced connectivity software for use in products in carrier and enterprise networking, consumer electronics, automotive, industrial measurement and control, and aerospace/defense markets. Wind River is How Smart Things Think(TM). Founded in 1983, Wind River is headquartered in Alameda, California, with operations worldwide.

About FreeBSD Mall
FreeBSD Mall Inc. is a publisher and distributor of FreeBSD software, and a provider of FreeBSD services and support. Founded in 1991, FreeBSD Mall is located in Concord, California.

Quick Toots: Ceres

Dave Phillips <dlphilp@bright.net>

[Editor's Note: This piece is a part of a series which the author is contributing to the Demudi project, www.demudi.org]

These tutorials are meant to be quick and interesting hands-on exercises. I will not explain basic installation and configuration details of the packages presented here (unless a specific need exists), nor will I take the time to explain subjects such as DSP theory or the ALSA audio API (even if I could). The Web hosts some wonderful sites for that kind of background information, and interested readers are encouraged to investigate the relevant URLs found on the Linux Sound & MIDI Software pages (http://sound.condorow.net/).

Transforming Sound With Ceres3

Ceres3 (http://www.music.columbia.edu/~stanko/About_Ceres3.html) is a spectral domain editor for audio files. Its display represents the frequency content of a sound as it changes over time, and the program's toolkit provides the means for directly editing and transforming that content. Ceres3 is a fantastic tool for studying, editing, and creating sounds, and in this tutorial we'll take a look at what can be done with some of Ceres3's powerful Transforms. You'll also be able to hear the results via downloadable MP3 files (encoded with BladeEnc (http://bladeenc.mp3.no/)at 64 kbps bitrate and 44.1 kHz sampling rate).

Preparation

The examples here were made under certain conditions. I used a 29-second monaural AIFF file (Ceres3 reads and writes only mono AIFF soundfiles), and I started the program with this command sequence:

```
ceres3 4096 4096 128 800
```

The command options set the FFT and window sizes, the window step factor, and the display width. Higher values for the first three options will result in higher resolution analysis (and a much larger analysis file).

Spread

Spread applies a granulation effect to the sound. Figure 1 displays the spectral content of the unaltered file, a reading by Ezra Pound:

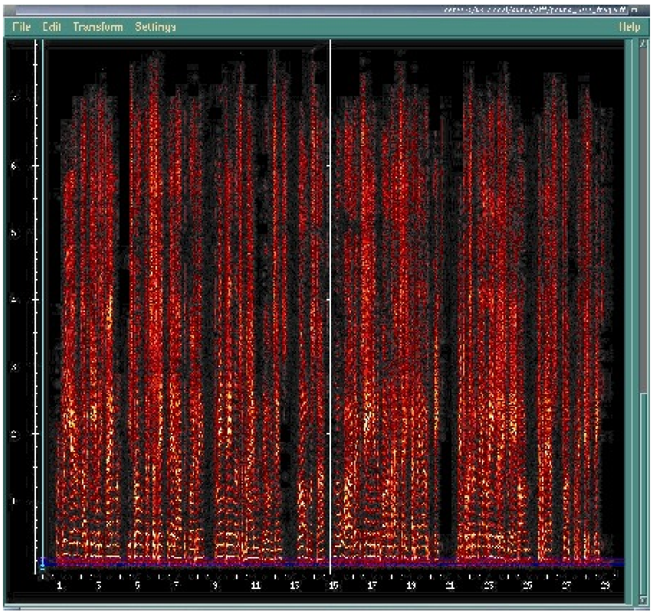
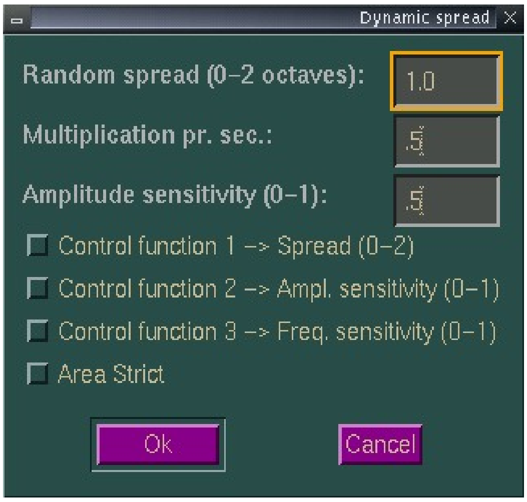


Figure 1: Ceres3 with soundfile loaded

You can download and hear this file at Original MP3 (http://linux-sound.org/quick-toots/1-ceres3/sounds/pound-original.mp3)

After selecting Spread from the Transforms menu we apply these values to the transform parameters:


Dynamic spread

Random spread (0-2 octaves): 1.0

Multiplication pr. sec.: .5

Amplitude sensitivity (0-1): .5

□ Control function 1 -> Spread (0-2)

□ Control function 2 -> Ampl. sensitivity (0-1)

□ Control function 3 -> Freq. sensitivity (0-1)

□ Area Strict

Figure 2: Spread parameter settings

Figure 3 shows the results:

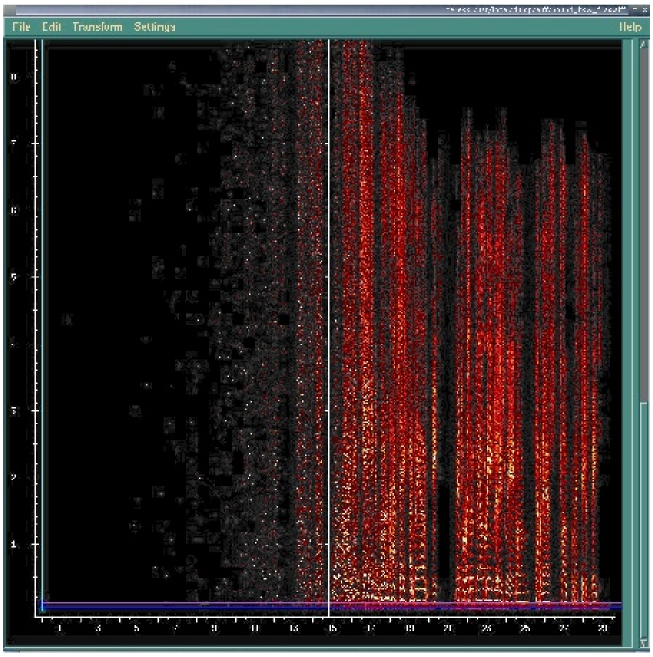


Figure 3: The soundfile after applying the Spread transform

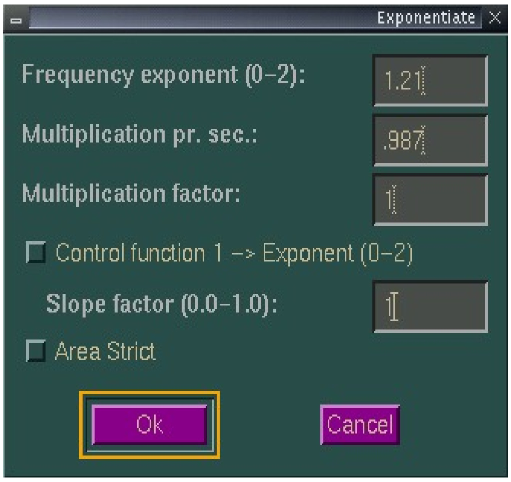
As the Spread MP3 (http://linux-sound.org/quick-toots/1-ceres3/sounds/pound-spread.mp3) demonstrates, the actual sound is an audio analog of the visual display. At first only a few scattered grains are heard, but by the mid-point of the time-scale the sound is rapidly coalescing into something more like the original file. By the end-point the sound is (more or less) identical to the original.

The transform's parameters are rather sensitive, and you may find that only a rather narrow range of values is useful for some parameters. For example, altering the random spread yields relatively subtle differences within a range from .50 to 2.0, but applying a similar range to the multiplier has a dramatic effect. The multiplier "advances" the granulation into the soundfile, i.e., at .25 the granulation occurs over only the first quarter of the file, at .50 it reaches the halfway point (as seen in Figure 3), but at .75 almost the entire file is granulated.

I leave it as an exercise for the reader to discover the effect of varying the amplitude sensitivity. Note also that this Transform can be dynamically controlled by Ceres3's function curves (see the Edit/Extract dialog) and that the effect can be restricted to within a delimited area (see Settings/Edit Display).

Exponentiate

Figure 4 shows some useful settings for the Exponentiate transform:


Exponentiate

Frequency exponent (0-2): 1.21

Multiplication pr. sec.: .987

Multiplication factor: 1

□ Control function 1 -> Exponent (0-2)

Slope factor (0.0-1.0): 1

□ Area Strict

Figure 4: Exponentiate transform settings

These values will have a dramatic bending effect on the sound, as can be seen in Figure 5 and heard in the Exponentiate MP3 (http://linux-sound.org/quick-toots/1-ceres3/sounds/pound-expon.mp3).

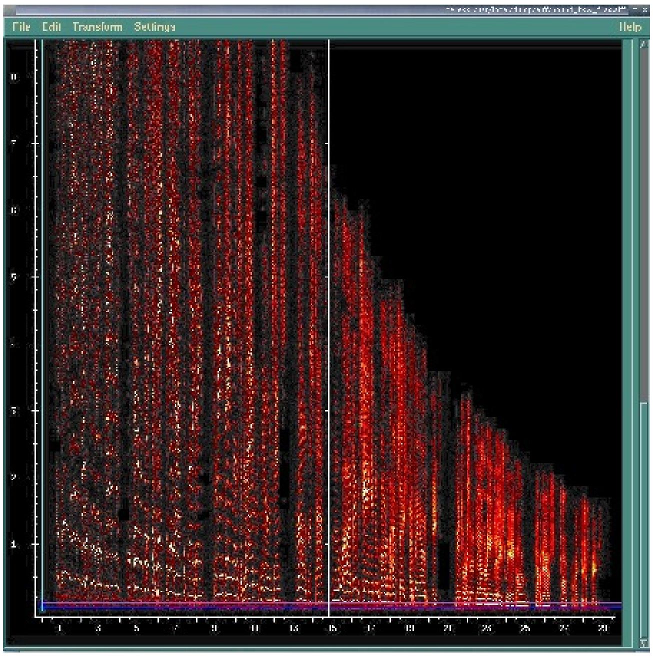


Figure 5: Soundfile after Exponentiate transform

This transform is especially sensitive to values for its multiplier and multiplying factor; however, the frequency exponent can be adjusted over a fairly wide range and a control function can be applied for dynamic modulation of the exponent value. Like Spread, the transform's effect can be delimited to a restricted area of the sound. Finally, the initial frequencies produced by the example may be filtered out or shaped by the Ceres3 paint brush. Yes, you can directly edit the spectral content with a resizable paint brush tool, and yes, it is very cool.

Combining Transforms

Our last example will combine three transforms to create a rather interesting "musicalizing" effect from the spectral content of our original soundfile.

First apply the Sieve and the Blur tramsforms with their default values. These actions will result in the display shown in Figure 6:

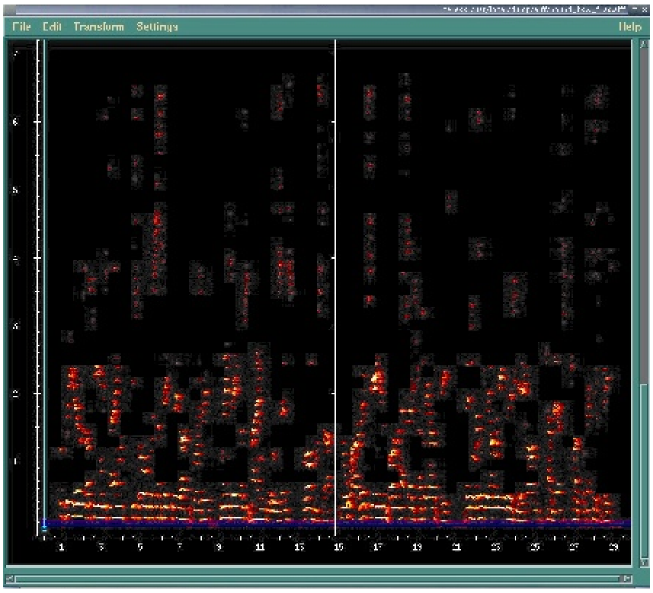
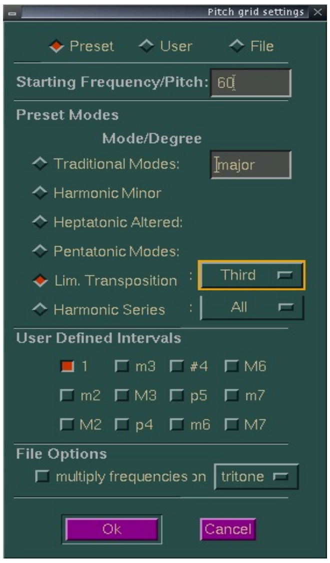


Figure 6: Soundfile after Sieve and Blur transforms

Next open the Settings/Pitch Grid dialog and set its values to those in Figure 7:


Pitch grid settings

● Preset ○ User ○ File

Starting Frequency/Pitch: 60

Preset Modes

Mode/Degree

○ Traditional Modes: major

○ Harmonic Minor

○ Heptatonic Altered:

○ Pentatonic Modes:

● Lim. Transposition : Third

○ Harmonic Series : All

User Defined Intervals

☒ 1 □ m3 □ #4 □ M6

□ m2 □ M3 □ p5 □ m7

□ M2 □ p4 □ m6 □ M7

File Options

□ multiply frequencies on tritone

Figure 7: Pitch Grid settings

From this point we will apply the Move To Pitch Grid transform (using the defaults) over three sections of the soundfile, changing the Pitch Grid settings each time to create a sense of harmonic motion in the resynthesized sound. The first part was set for Messiaen's third mode of limited transposition (base frequency at 60 Hz), the second section was set for the odd-numbered members of the harmonic series starting from a base frequency of 90 Hz, and the last part used a pitch grid for a heptatonic altered scale with a base frequency of 120 Hz. Figure 8 shows the results, you can hear them in the Combo MP3 (http://linux-sound.org/quick-toots/1-ceres3/sounds/pound-combo.mp3)

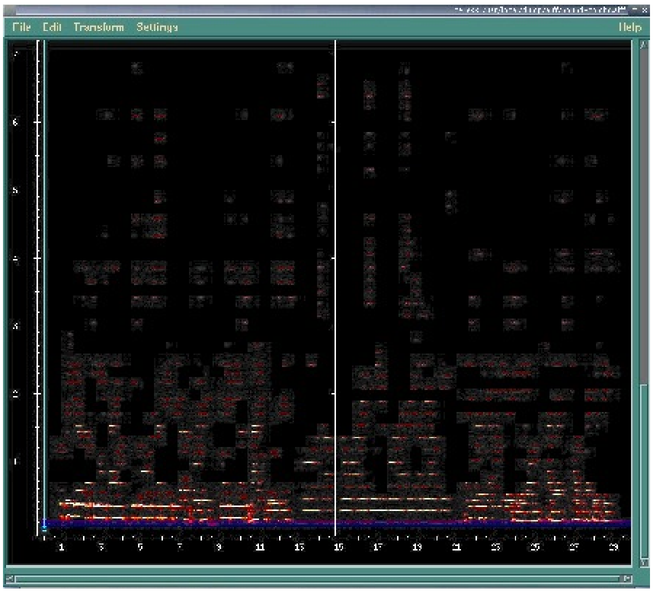


Figure 8: Soundfile after combined processing

Going Out...

You can keep piling on transform after transform (the Mirror and Spectrum Shift effects might be interesting), and Ceres3 contains many other interesting aspects (including the translation of sound into a Csound score), but I must stop somewhere. Fortunately the program's interface is well-designed and invites experimentation, so download it, build it, and make some joyful noises of your own. Enjoy, and feel free to let me know how you fare with Ceres3.

This article is re-printed with permission. The originals can be found at:

http://linux-sound.org/quick-toots/1-ceres3/quick-toot-ceres3.html

Chrooting All Services in Linux

Mark Nielsen http://www.tcu-inc.com/mark/

Abstract:

Chrooted system services improve security by limiting damage that someone who broke into the system can possibly do.

Introduction

What is chroot? Chroot basically redefines the universe for a program. More accurately, it redefines the "ROOT" directory or "/" for a program or login session. Basically, everything outside of the directory you use chroot on doesn't exist as far a program or shell is concerned.

Why is this useful? If someone breaks into your computer, they won't be able to see all the files on your system. Not being able to see your files limits the commands they can do and also doesn't give them the ability to exploit other files that are insecure. The only drawback is, I believe it doesn't stop them from looking at network connections and other stuff. Thus, you want to do a few more things which we won't get into in this article too much:

- Secure your networking ports.
- Have all services run as a service under a non-root account. In addition, have all services chrooted.
- Forward syslogs to another computer.
- Analyze logs files
- Analyze people trying to detect random ports on your computer
- Limit cpu and memory resources for a service.
- Activate account quotas.

The reason why I consider chroot (with a non-root service) to be a line of defense is, if someone breaks in under a non-root account, and there are no files which they can use to break into root, then they can only limit damage to the area they break in. Also, if the area they break into is owned mostly by the root account, then they have less options for attack. Obviously, there is something wrong if someone actually does break into your account, but it is nice to be able to limit the damage they can do.

PLEASE REMEMBER that my way of doing this is probably not 100% accurate. This is my first attempt at doing this, and if it just partially works well, it should be easy to finish out the rough edges. This is just a roadmap for a HOWTO I want to create on chroot.

How are we going to chroot everything?

Well, We create a directory, "/chroot" and we put all of our services under there in the following format:

- Syslogd will be at chrooted with each service.
- Apache will be at /chroot/httpd.
- Ssh will be at /chroot/sshd.
- PostgreSQL will be at /chroot/postmaster.
- Sendmail will be chrooted, but it won't be running under a non-root account, unfortuantely.
- ntpd will be chrooted to /chroot/ntpd
- named will be chrooted to /chroot/named

Each service should be completely isolated.

My Perl script to create chrooted environments.

Config_Chroot.pl.txt (http://www.linuxfocus.org/common/src/article225/Config_Chroot.pl.txt) should be renamed Config_Chroot.pl after you download it. This perl script lets you list the services being installed, view the config files, configure a service, and start and stop the services. In general, this is what you should do.

- Create the chroot directory.
  `mkdir -p /chroot/Config/Backup`
- Download Config_Chroot.pl.txt to /chroot/Config_Chroot.pl
- Change the $Home variable in the perl script if you are not using /chroot as the home directory.
- Download my config files.

Now, the important thing here is: ***I have only tested in on RedHat 7.2 and RedHat 6.2.***

Modify the perl script for your distribution.

I ended up making a huge gigantic article on Chroot, but with my Perl script, it became much smaller. Basically, I noticed after chrooting many services, they all have very similar files and configurations that needed chrooted. The easiest way to figure out which files need copying for a particular service is to look at the manpage and also type "ldd /usr/bin/file" for programs that use library files. Also, you can chroot the the service you are installing and manually start it to see what errors you get or look at its log files.

In general, to install a service do this:

```
cd /chroot
./Config_Chroot.pl config  SERVICE
./Config_Chroot.pl install SERVICE
./Config_Chroot.pl start   SERVICE
```

Chrooting Ntpd

Ntpd is just a time service that lets you keep your computer and other computers in sync with the real time. It was a simple thing to chroot.

```
cd /chroot
# Uncomment the next line if you don't use my
config file.
#./Config_Chroot.pl config  ntpd
./Config_Chroot.pl install ntpd
./Config_Chroot.pl start   ntpd
```

Chrooting DNS or named

Already done, check out
http://www.linuxdoc.org/HOWTO/Chroot-BIND8-HOWTO.html
or
http://www.linuxdoc.org/HOWTO/Chroot-BIND-HOWTO.html

Or, if you want to use my script,

```
cd /chroot
 # Uncomment the next line if you don’t use my
config file.
#./Config_Chroot.pl config  named
./Config_Chroot.pl install named
./Config_Chroot.pl start   named
```

Chrooting Syslog with services and my complaints.

I want to chroot syslogd. My problem is, syslogd uses /dev/log by default, which can’t be seen by chrooted services. Thus, I can’t syslogd easily. Here are the possible solutions:

- Chroot syslogd with every service. I actually tested this, and yes, I was able to log stuff. I don’t like this since I have a root running service.
- See if we can connect to an offsite logging facility.
- Just log files to a file and not through syslogd. This is probably the best security option, although if someone breaks, they could play around with the logs.
- Configure the main syslogd to look at several locations to get all the services. You use the -a option with syslogd to do this.

My only solution was to make sure syslogd is chrooted with every service. I would like some sort of solution which would log stuff in a non-root account using its own chrooted environment, like maybe a network port. It can probably be done, but I am going to stop where I am at and figure out a better solution later.

If you do not want to make a separate syslogd for each service, then with the main syslogd that you are running on your system, add the following command when syslogd starts:

```
syslogd -a /chroot/SERVICE/dev/log
```

If I had ssh and dns running, it might look like,

```
syslogd -a /chroot/ssh/dev/log -a
/chroot/named/dev/log -a /dev/log
```

Last note on Syslogd, I wish I could make it run under a non-root account. I tried a couple of simple things, but it didn’t work and I gave up. If I could run syslogd under a non-root account with each service, that would satisfy my security issues. Possibly, even have it log offsite.

Chrooting Apache

This was extremely easy to do. Once I got it setup, I was able to execute Perl scripts. Now, my config file is rather long because I had to include Perl and the PostgreSQL libraries into the chrooted area. One thing to note, if you are connecting to a database, make sure your database service is running on the 127.0.0.1 loopback device and you specify the host to be 127.0.0.1 in your Perl scripts for the DBI module. Here is an example of how I connect to a database using persistent connections in apache:

```
$dbh ||= DBI->connect(’dbi:Pg:dbname=DATABASE’
,"","", {PrintError=>0});

if ($dbh ) {$dbh->{PrintError} = 1;}
else
  {$dbh ||=
DBI->connect(’dbi:Pg:dbname=DATABASE;host=127.0.0.
1’,"","",
      {PrintError=>1});;}
```

Source: http://httpd.apache.org/dist/httpd/
Compile and install apache on your main system at /usr/local/apache.

Then use the perl script.

```
cd /chroot
 # Uncomment the next line if you don’t use my
config file.
 # ./Config_Chroot.pl config  httpd
./Config_Chroot.pl install httpd
./Config_Chroot.pl start   httpd
```

I changed my httpd.conf file to have this stuff:

```
ExtendedStatus On <Location /server-status>
    SetHandler server-status
    Order deny,allow
    Deny from all
    Allow from 127.0.0.1
</Location> <Location /server-info>
    SetHandler server-info
    Order deny,allow
    Deny from all
    Allow from 127.0.0.1
</Location>
```

Then, just point your browser at
http://127.0.0.1/server-status
or
http://127.0.0.1/server-info
and check it out!

Chrooting Ssh

First off, ideally, you should port forward ssh on port 22 to port 2222. Then, when you start ssh, have it listen to port 2222 under a non-root account. For the initial ssh connection, we want to have secure accounts with passwords just to let the people in, but not do anything else. After they log in, then have a second ssh program running on port 127.0.0.1:2322 which will let them connect to the real system -- the second ssh program should ONLY listen on the loopback device. Now this is what you should do. We aren’t going to do it. The only thing we are going to do is chroot ssh for this example. Exercises which are left up to the reader include putting sshd under a non-root account and to install a second sshd which listens on the loopback device to let people into the real system.

Again, we are going to just chroot ssh and let you worry about the consequences of doing that (you won't be able to see your entire system if you just do this). Also, ideally, it would be nice to set this up to record logs offsite. Also, we should use OpenSSH, but I am using the commercial SSH for simplicity (which is not a good excuse).

Source:
http://www.ssh.com/products/ssh/download.cfm
Install ssh at /usr/local/ssh_chroot. Then use the Perl script.

```
cd /chroot
 # Uncomment the next line if you don't use my
config file.
 # ./Config_Chroot.pl config  sshd
./Config_Chroot.pl install sshd
./Config_Chroot.pl start   sshd
```

I suppose one really good thing with putting ssh under a chrooted environment is that if you want to use it to replace an ftp server, people will have limited access to your area. Rsync and SCP go very well together for letting people upload files. I don't really like to put an ftp server up for people to log into. A lot of ftp servers are also chrooted, but they still transmit passwords in the clear, which I don't like.

Chrooting PostgreSQL

This was almost as simple as perl, except it required a few more libraries. Overall, it wasn't that hard to do. One thing I had to do was put PostgreSQL open to the network, but only on the loopback device. Since it was chrooted, other chrooted services couldn't get to it, like the apache web server. I did compile Perl into PostgreSQL, so I had to add a lot of Perl stuff to my config file.

Source:

ftp://ftp.us.postgresql.org/source/v7.1.3/postgresql-7.1.3.tar.gz
Compile and install apache on your main system at /usr/local/postgres. Then use the Perl script.

```
cd /chroot
 # Uncomment the next line if you don't use my
config file.
 # ./Config_Chroot.pl config  postgres
./Config_Chroot.pl install postgres
./Config_Chroot.pl start   postgres
```

Chrooting Sendmail

Go ahead and execute my script.

```
cd /chroot
 # Uncomment the next line if you don't use my
config file.
 # ./Config_Chroot.pl config  sendmail
./Config_Chroot.pl install sendmail
./Config_Chroot.pl start   sendmail
```

Now are there catches? Yes. It is still running as root. Darn. Also, certain files are recreated by the /etc/rc.d/init.d/sendmail file when it is started. Mine script doesn't handle that. Anytime you make changes to sendmail under /etc/mail, please copy the changes to /chroot/sendmail/etc also. Also, you will have to point /var/spool/mail to /chroot/sendmail/var/spool/mail so that the sendmail program and the users (when they log in) can see the same files.

The good thing is, you can always send mail out, it is just receiving it that is the problem. Thus, I was able to install sendmail with apache without any problems. Some of my perl scripts send mail out, and so, I needed the sendmail files copied into the chroot area for apache.

Other things to chroot.

Here is my philosophy:

1. Everything should be chrooted, including sendmail, ssh, apache, postgresql, syslog, and any service running on the computer.
2. Everything should be put under a non-root account (you might need to port forward protected ports to a non-protected port). This includes sendmail and syslog by the way.
3. Logs should be sent offsite.
4. A partition should be setup for each service to limit the amount of diskspace a hacker can use up if they decide to write files. You could use a loopback device to mount files as filesystems for some of these services if you run out of partitions.
5. Root should own all files that do not change.

Now, when it comes to sendmail and syslogd, I still think they should be run under a non-root account. For sendmail, this should be possible, but I found it extremely difficult to run as a non-root account.

I haven't been successful getting sendmail to run as a non-root account, and I think it is a serious mistake for it not to be. I know there are problems doing that, but I think they can ALL be taken care of. As long as file permissions are taken care of, I don't see why sendmail needs to be run as root. There might be some reason I am overlooking, but I doubt any of the obstacles can't be overcome.

For syslog, I haven't even tried, but I would say logs should should be logged under a non-root account and I don't see why that shouldn't be possible. At least I was able to get syslog to be chrooted for each service.

All services should be setup as non-root accounts. Even NFS. Everything.

Suggestions

- Use two logins for ssh and have two running sshd daemons.
- Figure out how to get sendmail or some other mail program running as non-root.
- Strip out the unnecessary libraries under /lib. I just copied everything to make it easy on myself. Most of it you don't need.
- Do remote logging of syslogd and find out if we can

attach syslogd to a network port and get all the services to connect to that network port on the loopback device. See if we can get syslogd to run as a non-root account.

Conclusion

I think chroot is cool for all services. I believe it is a big mistake not to chroot all services under non-root accounts. I wish a major distributions would do that, or a smaller distribution: ANY distribution. Mandrake started off by taking stuff from RedHat and expanding off of it, so perhaps, someone should take Mandrake and expand chroot off of them. Nothing prevents people from redoing other people's work in GNU/Linux, so I think it is possible. If some company wanted to chroot everything and create a systematic easy environment for people to manage their chrooted services, they would have a fantastic distribution! Remember, now that Linux is going mainstream, people don't want to see the command line, so if everything is done at a gui level, they don't need to see the guts and they really don't need to know what is going on -- they just need to be able to configure it and know that it just works!

I am in 100% complete support of the idea that all services should be chrooted with non-root accounts and that any distribution that doesn't do this is less than proper for me to use in a production environment.

I am going to chroot everything, as much as possible -- eventually I will get there.

I plan on creating s HOWTO about chrooting. I am submitting a request to have someone help me convert this article into LyX format so that it can be put in the HOWTOs for Linux.

References

If this article changes, it will be available here
http://www.gnujobs.com/Articles/23/chroot.html

This article is re-printed with permission. The originals can be found at:

http://www.linuxfocus.org/English/January2002/article225.shtml

QCAD: Technical drawing

André Pascual <apascual@club-internet.fr>

Abstract:

QCad is a 2D CAD system with which you can draw and modify plans easily.

General Notes

A "plan" is any precise plane representation of a real object for study or for production purposes. The dimensions of each element (entity) of which the drawing consists of, must be exact no matter what scaling is used. This differentiates a CAD program from a vectorial drawing tool such as Sketch, Illustrator or Corel Draw, which is a more or less faithful representation of reality. With CAD a plan first of all has to be exact. This is in contrast to illustrations (Drawing) where the aesthetics of a picture are of more concern.

Installation of QCad

The version qcad-1.4.x used for this article is on the applications CD of the Redhat and Mandrake distributions as ready-made package. Other distributions surely have similar packages. You can download the newest version at http://www.qcad.org/. Qcad needs QT 2.2 as GUI library.

A little bit of theory

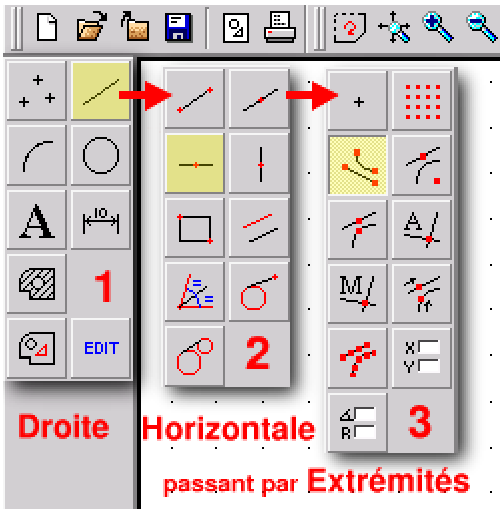
Before beginning with your first document you should have understood certain CAD concepts and definitions.

The entities

An entity is a layout element that is "known" to the program by its form (segment, arc...), in the geometric characteristics of its position (vertical, tangent...), in its start and end positions which determine its dimension (fixed at intersections, coordinates, center...), in its attributes (color, thickness, types of characteristics) and its membership to a layer (blue print). Generally speaking to build an entity, it is necessary:

- to define your view on the working layer
- to define the attributes
- to choose the nature: straight line, segment, circle, ellipse, point, curve, hatch, text...
- to indicate the geometric framework of the construction: horizontal, oblique, concentric, vertical...
- to indicate the constraints

It results in building a virtual but exact sentence of this kind: circle of radius X from the center passes through the endpoints of an entity which was indicated by a right click, etc. The points will have to be indicated by a left mouse click near the desired points which have to be chosen among those that the system offers. It should be noted that the concept of fixation is found elsewhere under the name <snap> For example the sentence horizontal straight line, which touches the outer lines of an entity is composed with the following menus and sub-menus:



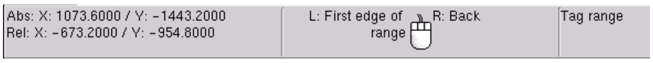
Note: That you get to the first start menu by clicking with the right mouse button onto the "paper".

The Layers

Elsewhere called levels, plans, blue prints. The layers describe in fact a virtual pile of celluloids. Each celluloid contains a part of the drawing, recognizable as a whole if you look on the pile from above, thanks to its transparency. A layer can be moved in the pile, removed (which affects only the part of the drawing that it contains), frozen or made invisible. The layer on which you work is the only active one at the moment. The operations that you carry out affect only it. When you assign attributes of color, line types or line thickness then all entities that you draw will have them by default. However you could assign blue to an entity that is on a red layer by modifying its properties. For a complex drawing you will work on one layer after the other which allows to make a certain subset visible or invisible, print only one piece, modify nothing but this.

The Status Line

The status line is located in the lower part of the main window. It is not specific to CAD software programs but nevertheless essential. As a command requires several successive operations carried out in a defined order, the program shows in the status line the operations that should follow and what it expects from you and this until the end. It is therefore absolutely necessary to read the information that is displayed in that lower line if you do not want to risk that the CAD session ends with the declaration that this is a +–@–#!! program. In CAD the result is precise if the designer is working exactly and systematically.



Abs: X: 1073.6000 / Y: -1443.2000
Rel: X: -673.2000 / Y: -954.8000

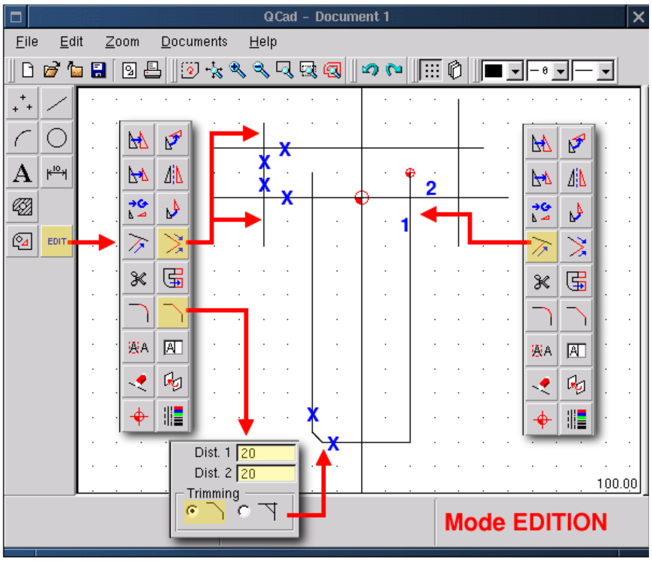
L: First edge of range R: Back

Tag range

Methods of Drawing

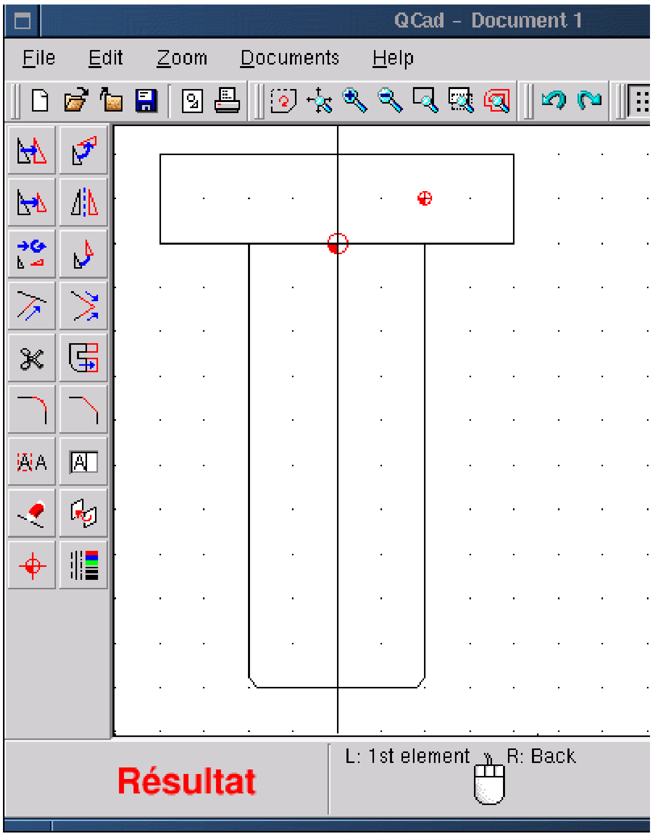
There are several ways how to do it, with at least two of them being excellent. Both use the concept of drafts based on not dimensioned (very long) straight lines but with precise relative positions (distance of one compared to the other). These straight lines, horizontal and vertical are called lines of construction in DTM or SoildWork and geometries in TSCadDraw.

The first method consists of defining a profile based on these straight lines as points of support. The exercise which we will follow in the rest of this article will demonstrate this. The second method consists of defining a profile by adjusting the lines at fitting intersections. To do this with QCad you have to right click to get back to the main menus and then choose <edit><Trim two object> then click on the line that you would like to trim (cut) next click on the line where your first line should stop. Here are 3 examples of editing objects:



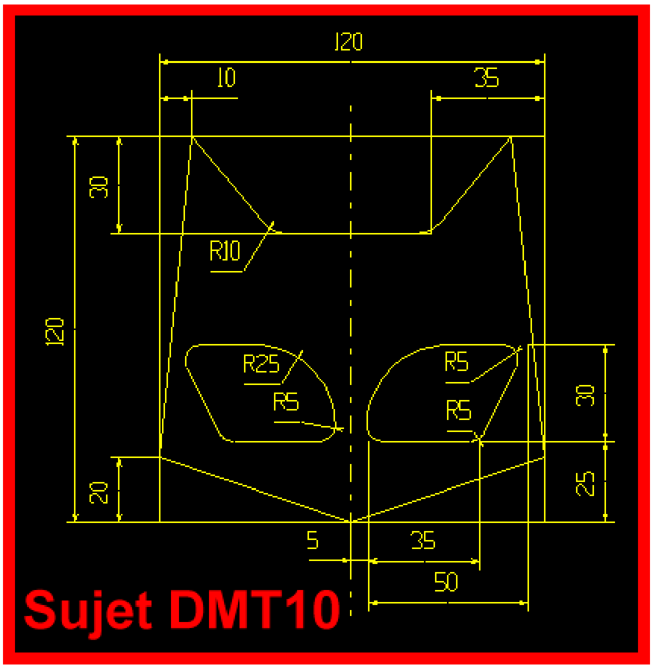
Dist. 1 20
Dist. 2 20
Trimming

In this figure as in the following ones the yellow boxes show the selected functions, footnote: not colored by Qcad itself, and the blue crosses show corner points on which you can click. With the function <Trim objects> one makes an element fit to another. It is important to click first (1) on the part that you want to fit and second (2) on the entity that intersects the first. For the function <Bevel> it is important to determine the X and Y values of the edge before, no matter if it is trimmed or not and finally to click on the entity to be beveled. The steps are the same when working with intersecting lines and the function <Round>. One should also mention that QCad tries to be quicker than the user or tries to help you with your decision, in fact when a function is activated which needs the selection of a second entity to go on, then QCad modifies the color of that entity which is near the pointer and indicates to you that you can select it with a left mouse click. It works the same way with the fixation points which are colored red. The right mouse click cancels an operation and allows you to go back to the main menu. The following figure shows the result of these various adjustments:



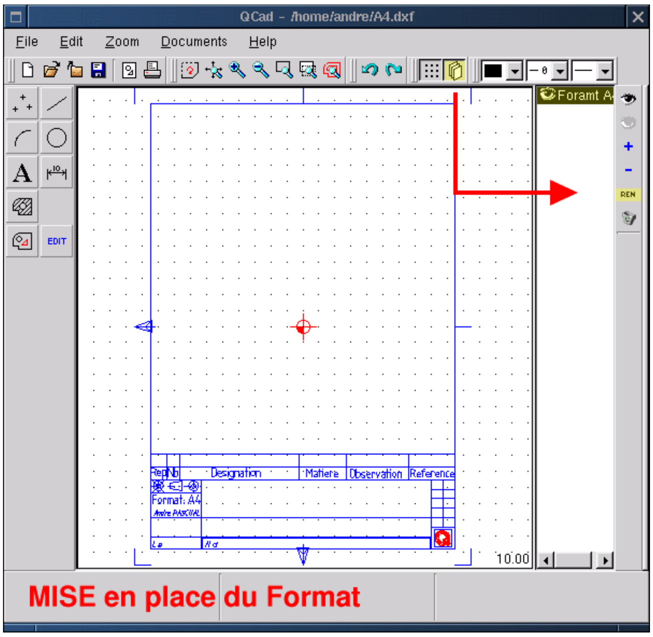
Example application

To get to know a program there is nothing better than to use it. Let's try to draw an object which is inspired by the logo of SEV Marchal which I usually use for my beginner's courses in numerical control. With DMT 10 by Mécasoft it can be drawn in less than 5 minutes, annotations included.



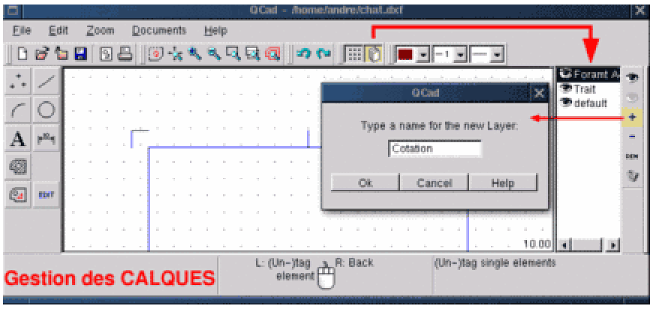
Set up of a page format

This is not strictly necessary for the exercise but a technical drawing obeys to standards which define among other things the view and the aspect of the format (frame and data block) in which they are contained. Here I use a format coming from DMT10 transferred to DXF, the only file type that QCad can read and generate which in turn guarantees the exchange in two directions with all CAD programs in the world. Once the format is loaded you see a mark off of the drawing zone with a zero reference in the middle of the area. It is recommended to delete unnecessary layers, to rename those which contain the format frame and to add those that are described in the following paragraphs.



Managing the layers

By clicking on the icon representing several piled up pages you open a side window called "Layer List". The selected blue print becomes the active blue print, it appears with intensified brightness. The eye besides the name of the layer allows to make it visible or invisible. The open eye to the very right of the layer list window makes all layers visible while the closed eye makes them invisible. The plus sign adds a layer to the list, the minus sign eliminates the selected layer, the symbol REN allows to rename the selected layer and the trash can deletes all empty layers. Now we need a layer <Format A4> that contains the frame, a layer <Trait> that contains the drawing in a front view and a layer <Annotation> that contains the annotations to the drawing.



QCad - /home/andre/chat.dxf

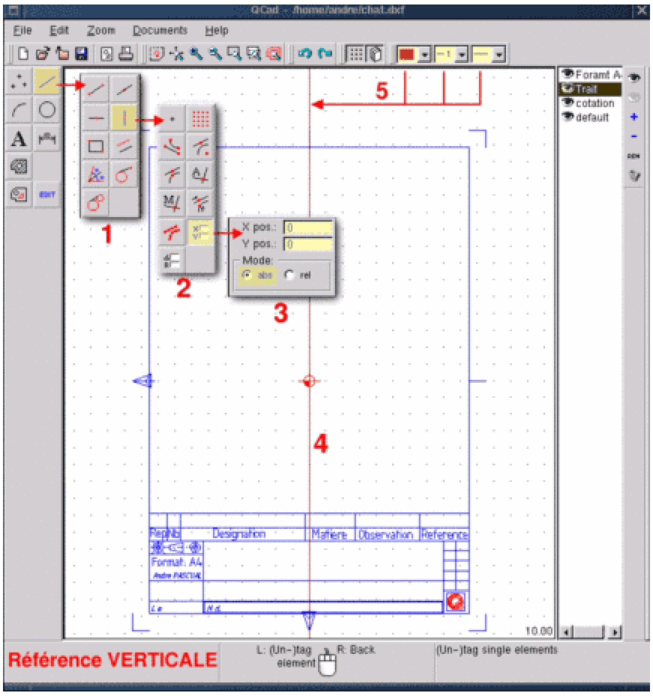
QCad

Type a name for the new Layer:

Cotation

Defining a vertical reference

This straight line passes through the origin (zero) and allows the construction of parallels on the X-axis. Our drawing will be symmetrical to this line. At first you activate the layer <Trait>; then you choose a continuous width type with a thickness of 1 and the color red (point 5) then you construct the "sentence" <straight line><vertical(1)><passing through absolute coordinates(2)> <enter 0,0(3)> <click left(4)>



QCad - /home/andre/chat.dxf

X pos.: 0

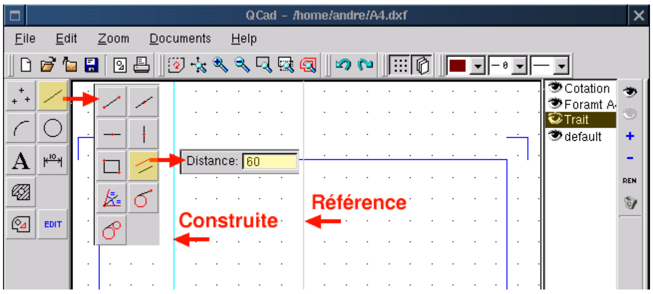
Y pos.: 0

Mode: ● abs ○ rel

Construct a parallel to 60

If necessary you remove the menus by several right mouse clicks and construct the following: <straight line>< Create parallels><enter 60> and approach the position marks of the reference straight line. It will become grey in turn and according to the position of the pointer QCad suggests to construct the parallel either to the right or to the left of the reference. Position the pointer slightly to the left and make a left mouse click. A straight line in cyan is created.

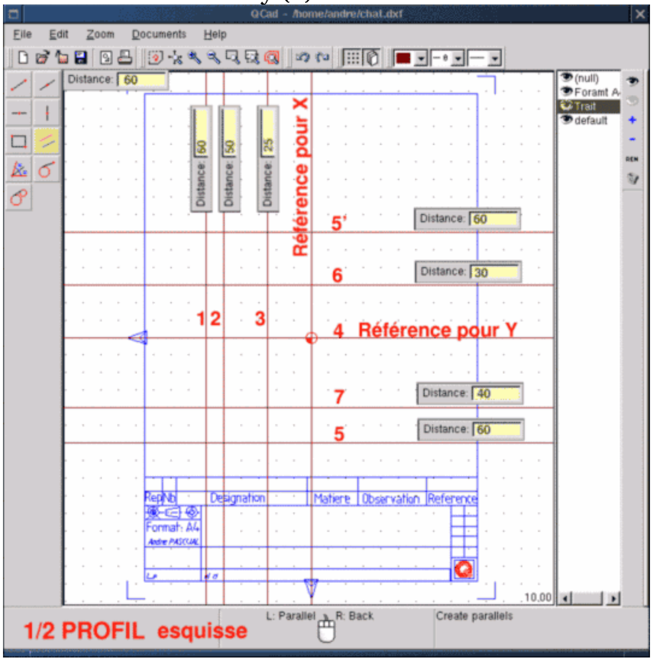
Construct the other straight lines



QCad - /home/andre/A4.dxf

Distance: 60

Most functions of QCad are repetitive, that is, they are active as long as they aren't replaced by another. Therefore <straight line><Create parallel><Distance of > is still activated. It is sufficient to replace the value 60 by 50 (Vertical 2) and to click and then to replace 50 by 25 (Vertical 3) and then to click. Go on with this for the horizontal straight lines which are defined with regards to the reference at 0 (Horizontal 4). Draw the horizontals (5) and (5') with distance 60, then (6) with distance 30 and finally (7) with distance 40.

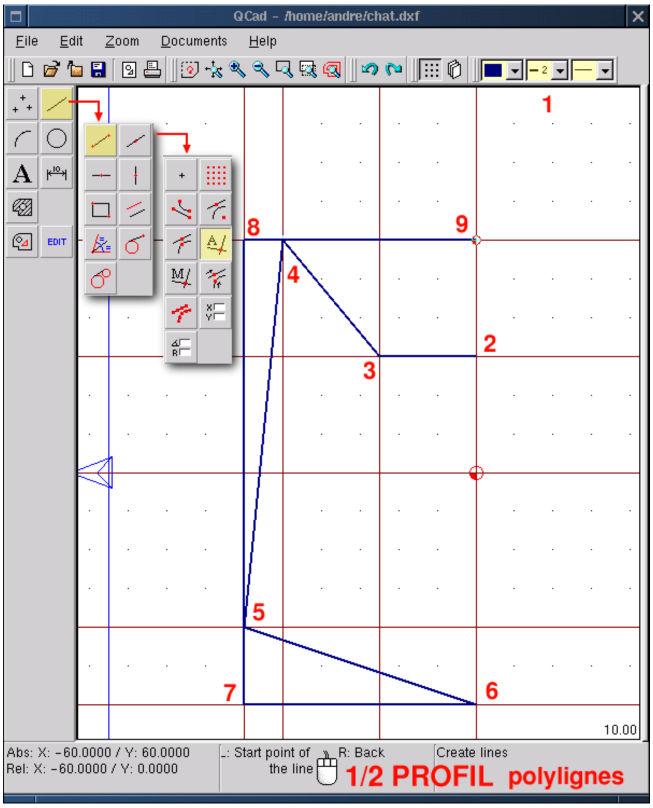


QCad - /home/andre/chat.dxf

Distance: 60

Construct the left half of the logo

For the construction we rely here on the straight lines which we have just drawn. You have to go back to the main menu with a right mouse click and choose <lines><multiple lines (button: create lines)>< passing through the intersection (Snap automatically to..)>. From this moment on when we position the pointer near the intersection of the straight lines it will be marked with a red circle. When this intersection is suitable as the beginning of a character segment, then make a left click, move to the next intersection and make a left mouse click again. The segment is drawn. But as the function is modal this last point which is the end of the segment that we have just drawn will at the same time be the beginning of the next segment. This allows the drawing of closed contours. If you don't need it for an additional segment a right click will interrupt the active function but doesn't cancel it. So for this half side of the profile choose a continuous line type, thickness 2 and color blue like in point 1 in figure QCad10 (below). Then left click on 2, 3, 4, 5, 6, 7, 8, 9 and finally a right mouse click. The left half of the profile is ready. Easy!



Abs: X: -60.0000 / Y: 60.0000
Rel: X: -60.0000 / Y: 0.0000

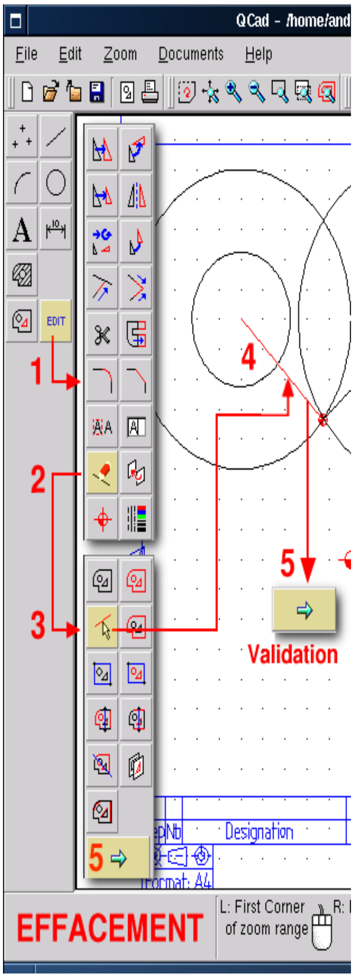
L: Start point of the line R: Back

Create lines

Editing the result

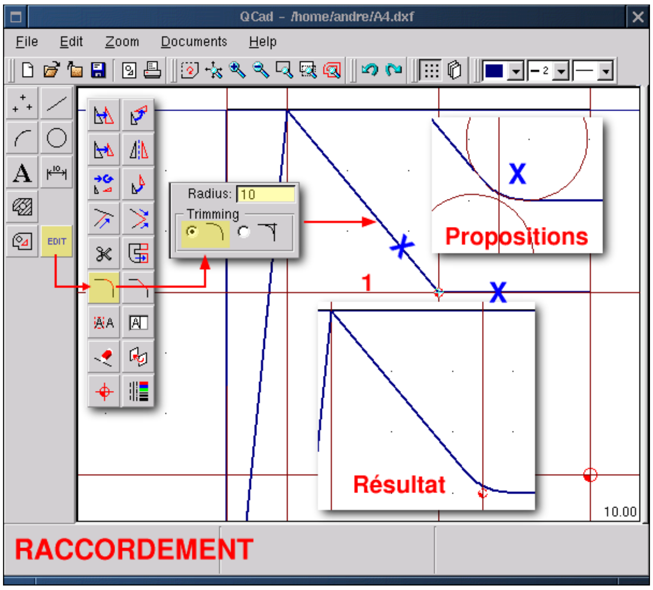
Editing is a modification of something existing. To add an adjustment or a beveling to a drawn profile or to delete a segment, that are modifications. Whatever the changes to make are there is a general approach. First you choose the function <Edit (1)> which opens a sub−menu of all possible modifications. Select the desired function, e.g. <delete objects(2)>, which opens a sub−menu for selection: contour, all entities, tag single element... This allows you to choose the borders of the modification. If you choose for example <Tag single element(3)> design the element (4) and then acknowledge the action by a left mouse click on the arrow icon (5) then the chosen element is deleted. Please note that the function <(un−)tag single element;> is a toggle, if you click on an element then it is selected, another click and it is deselected. This allows to remove certain elements from a global selection.

Adjusting the basis of the ear



L: First Corner of zoom range R: I

Going back to the main menu we delete the construction straight line called 1 in figure QCad12 (below) and choose <Edit><Round><Radius 10> <trimming> We determine the entities to trim, then we move the pointer near to the adjustment/link that has to be made. Qcad then suggests possible solutions (radius 10). If a fitting point is suggested a left mouse click keeps the linking/adjustment and trims the line.

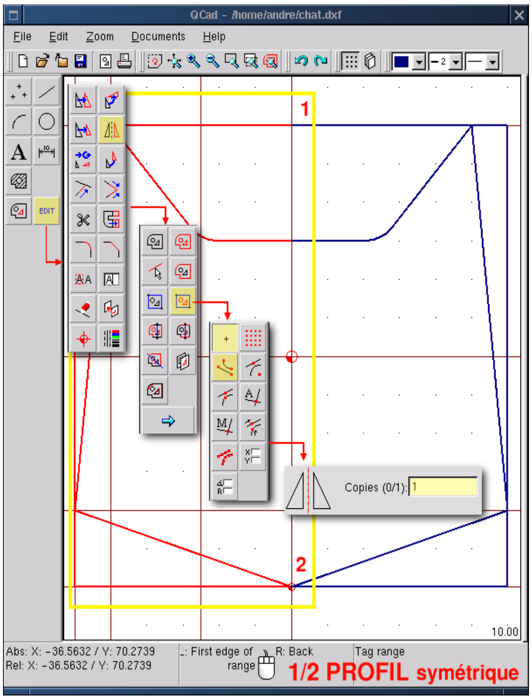


Radius: 10
Trimming

To construct the mirrored half of the profile

With the existing 1/2 profile it would be nonsense to draw another one, therefore it is sufficient to duplicate the first one symmetrically to obtain a complete profile. We choose <Edit><mirror

objects><Tag Range><Point (Snap to nothing)> and draw a window around the 1/2 profile (yellow frame in figure QCad13). The profile is selected: it becomes red. We make a right mouse click: we get back to the selected sub-menu. We acknowledge by clicking on the arrow icon. We get back again to the fixation sub-menu: we choose <Extremity (Snap to endpoints)> and determine the points 1 and 2 as shown in figure QCad13. A dialog box "Mirror" appears. If you type in a value of 0, the 1/2 profile is moved, if you give in a value of 1 it is duplicated. Therefore you have to give in a value of 1 and click <Ok>. And the profile is ready.

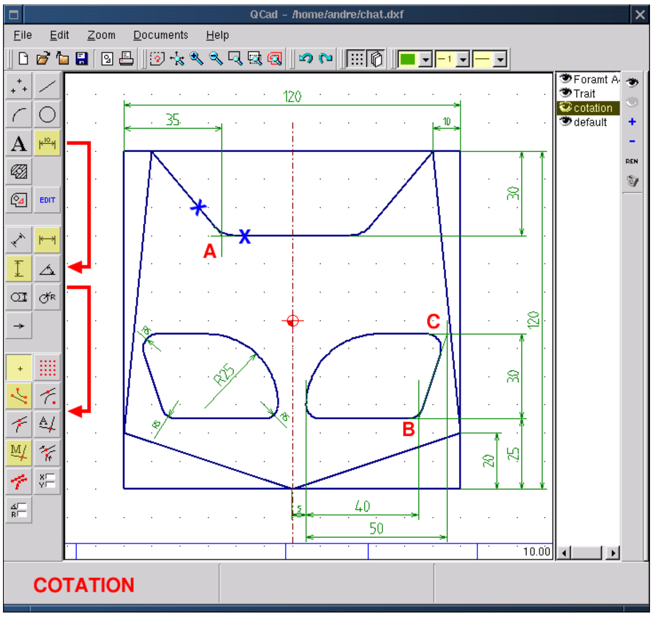


Drawing the eyes

By using what we have seen already we can draw the left eye as well through the horizontal and vertical construction lines (point 1). Then you draw the profile by building upon this straight line with a polyline. You trim it with a radius 5 (green crosses) and a radius 25 (magenta cross, points 2 and 3); finally you delete the construction line and duplicate the left eye by mirroring it to the right (point 4). All necessary commands for this operation have already been explained above.

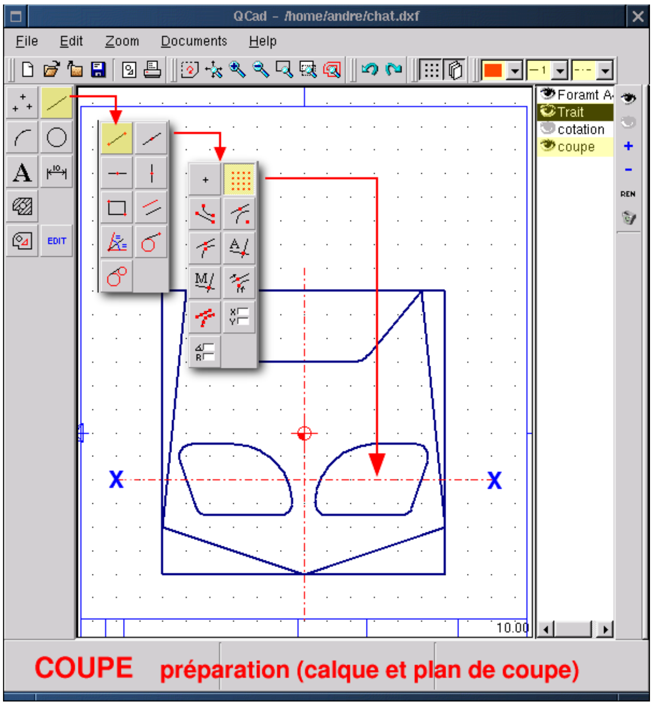
Adding dimensions

Annotations are no strength of QCad: it is impossible to give tolerances or to write somewhere else than in the middle of the lines that indicate the dimensions. The consequence of this last point is that the size of the characters is changed depending on the available space between two reference points. This give the whole drawing a strange aspect. Well, no matter how it is, to make annotations you have: to position yourself on the annotation layer, choose a fitting line attribute, especially a thickness of 1 and a color different from the other lines, unique if possible. But this isn't obligatory. Select <Annotation (Sub-menu dimensions)><Type of annotation horizontal or vertical or radius...><Endpoints to determine the position of the construction lines or certain intersection (Snap manually o..)>< Point (Snap to nothing) for positioning the dimension> To change from one way of fixation to another you may use the short cut keys: F for <point>, E for <Extremity>, X for < automatic intersections> etc. The points A, B and C (image below) are difficult to annotate with a dimension.



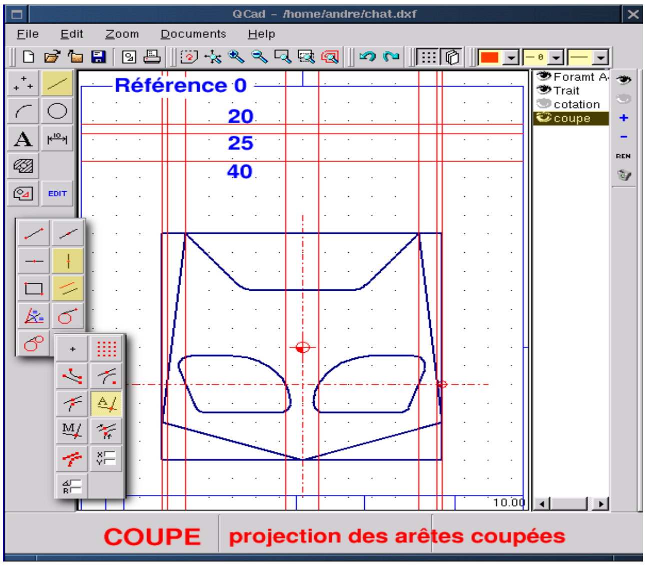
A sectional view (cut): preparation

The representation of a three dimensional object in 2D makes it necessary to order several views according to certain drawing norms even if it only is to show the thickness of the object. Our drawing represents an object of 20mm thickness worked on in a depth of 5mm. To just say this isn't explicitly enough and a cross sectional view becomes necessary. To do this: Make the layer <Annotation> invisible with a double click on the icon with the open eye. Add a layer <cross section> with the option + in the management menu for the layer. According to the norm a cross sectional cut is indicated by a line with a points and hyphens. Activate this as a style attribute and draw a line of width 1 between the eyes of our logo (<straight line><Polyline><snap to grid points>).


QCad - /home/andre/chat.dxf
File
Edit
Zoom
Documents
Help
Foramt A
Trait
cotation
coupe
X
X
10.00
COUPE
préparation (calque et plan de coupe)

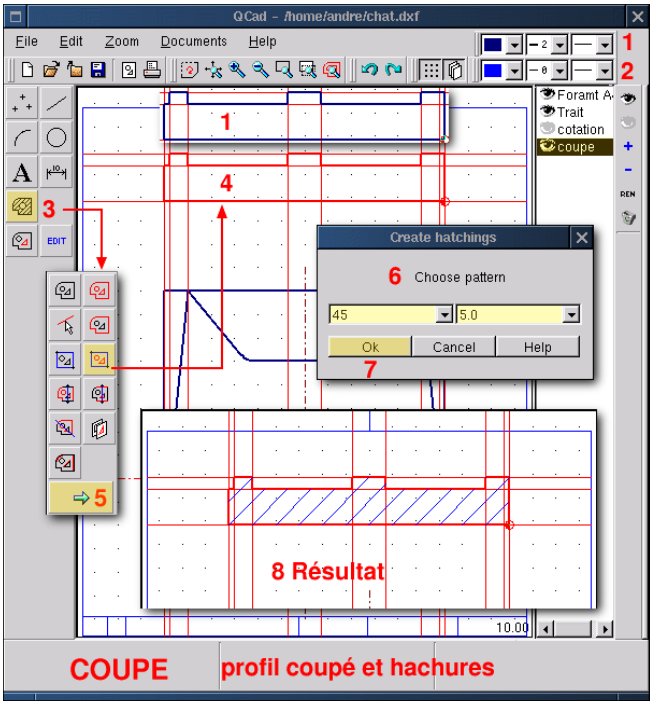
Sketch the cross sectional view

In industrial drawing whether with pencil and paper or with CAD there must always be a correspondence between the views. The cross sectional view is a projection along the line that indicates the cross sectional cut through the object.



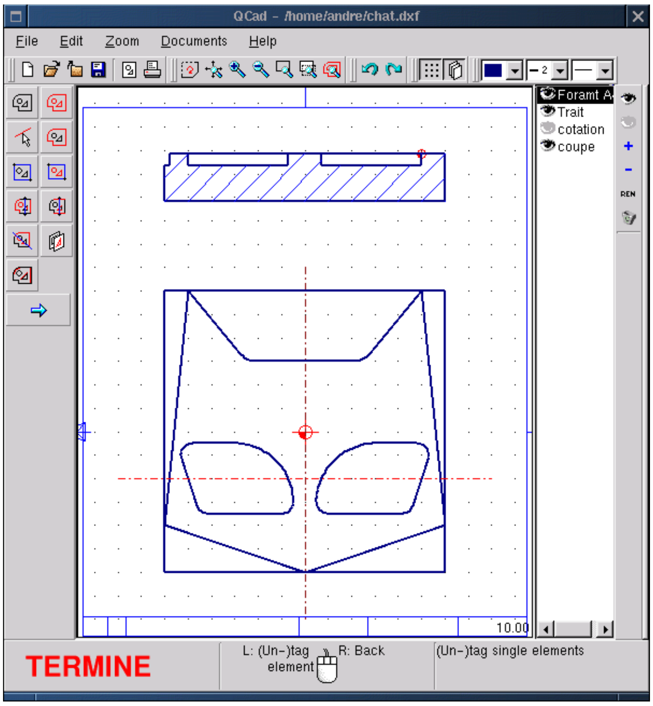
Drawing and hatching of the cross sectional view

We use again straight construction lines to draw the cross sectional profile with a polyline (figure QCad18, below). Modify the properties of the lines for the hatch (2). Select <Create hatchings(3)> <Tag range> <Passing through the Point (Snap to nothing)> and draw a square around the cross sectional view to select the area (4). Acknowledge (5). The dialog box "Create hatchings" appears. We choose a fitting hatch parameter(6). We press OK (7) and the hatched view (8) is ready.



And finally the finish

The cross sectional view as it is represented here is, according to the rules of industrial drawing, is a projection. Since obtain this view by virtually cutting our profile at the height of the eyes we have to indicate the depth of the eyes. You add this as follows <Straight polyline line (button line)><create lines> Now the drawing is ready. Only the frame of our paper (the data block) has to be filled in with text. I leave it to you to discover the < text function>.



TERMINE

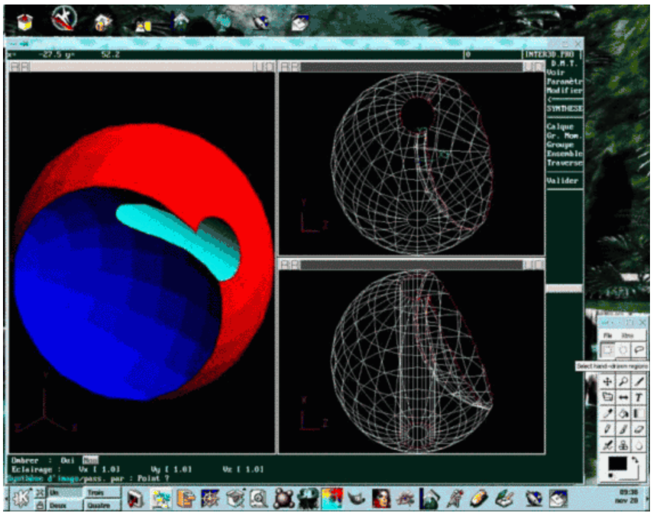
L: (Un-)tag element R: Back

(Un-)tag single elements

Conclusion

Linux CAD-Applications under the GPL aren't numerous. It is therefore appropriate to honor the authors of QCad and to thank them for the useful application they give to the community. Even though it isn't replacing industrial applications of the type of Cadkey, AutoCad or DMT, it remains a good educational tool and a tool for not too complex plans. One can regret the weaknesses of the annotation function, the absence of covering/lining/boarding functions (rowness, geometrical tolerances, sectional views) and the numeric limitation of the geometric border conditions. But you can congratulate the authors for the ease of handling, for the simple and convenient user interface, the powerfulness of linking/fitting and hatch functions, to the choice of the DXF format and not a proprietary format. Possibly as well to many other good things in QCad which I have not yet discovered. QCad has a help system but the documentation is English and remains therefore totally obscure to me. This proofs how easy the handling of QCad is. I have discovered everything by just playing around with the program.

Having said that, the optics of CAD have changed dramatically in the recent years. It is less a question to produce 2D drawings to represent three-dimensional objects. But to work out a 3D model completely defined in form and dimension with the help of performant tools and volume modelers. The program generates then automatically plans, annotations and the listing for numerically controlled machines. These programs are ProEngineer, SoldConcept, Catia, Solid Edge or Think3D.... When will these tools be available to Linux? At the moment we have QCad and CAM Expert, its commercial brother, equipped with two dimensional CAD and old MS-DOS products such as DMT10 by Mécsoft which runs perfectly in dos-emulation under Linux as you can see in the last screen shot (below).



This article is re-printed with permission. The originals can be found at:

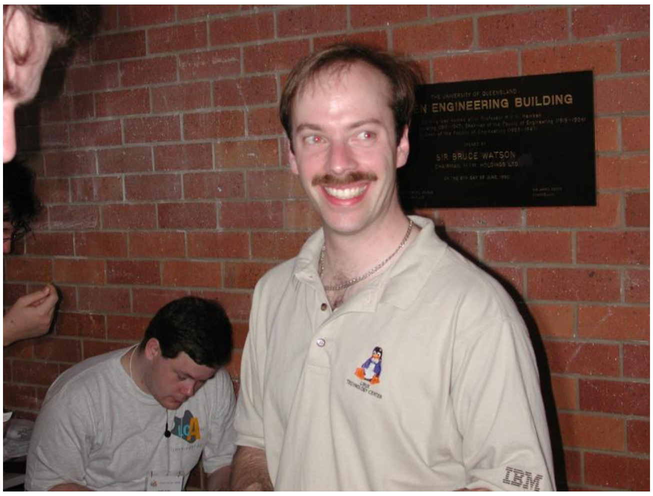
http://www.linuxfocus.org/English/January2002/article132.shtml

linux.conf.au -- The Photos

Photographer: Anthony Rumble <anthony@everythinglinux.com.au>

For those that missed it, here's a sequence of photos from the recent national Linux conference, held in Brisbane. It was a great event, and we can look forward to similar interesting material and speakers at the forthcoming AUUG 2002 conference, in early September, Melbourne. Mark the date down in your diares now! Our thanks go to Anthony Rumble of Everything Linux for the happy–snaps

Registration.



Rusty

Greg Lehey

Terminal Room

Tridge

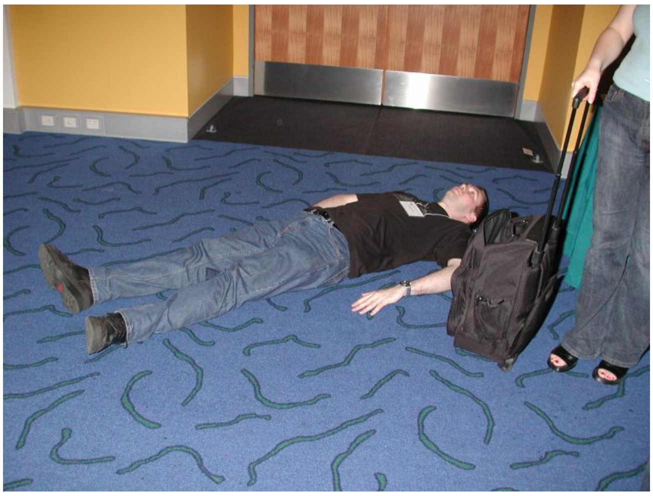
Bdale Garbee from HP

Networking reception

Brad Hards talking about Linux USB

Your AUUGN editor at left, with ***Slashdot's*** Chris Di Bona at right. We're either arguing over the finer points of various open source licences, or what's going to be on the menu of the conference dinner (about to soon start.) David Axmark, co-founder of MySQL is in the background in the white t-shirt.

Some if IBM guys at the conference



Contemplating illegible Perl code

Food!

These photographs are re-printed with permission. The originals can be found at:

http://www.linuxhelp.com.au/~smilie/hashslug/lca1.shtml

Irene Graham, Electronic Frontiers Australia, with the soon to be auctioned 'autographed t-shirt'

Organising Committee member, Anand Kumria: The 'autographed t-shirt' will soon be mine! :-)

Not so fast, says the real winner, with his $1000+ investment.

Official Gadgeteer Hands On Review: Sharp Zaurus SL-5000D

Julie Strietelmeier <julie@the-gadgeteer.com>

Product Requirements:

Desktop:
Windows 98, or 2000, USB Port

PDAs (Personal Digital Assistants) have sure come a long way since the chicklet keyed Sharp Wizard clamshell devices. Way back in 1988, when the Wizard was first introduced, these devices weren't even called PDAs. Instead, they were called Electronic Organizers. At that time, the Wizard was king, and a PalmPilot was still just an idea yet to be thought of by Jeff Hawkins. In 1994, the Sharp Wizard evolved into the Zaurus, another clamshell organizer. Then eventually the Zaurus grew into a color handheld device that was only available in Japan (why do they always get the cool stuff?). Now, Sharp has once again updated the Zaurus, this time into a small form factor Linux / Java PDA.

Review disclaimer: This is a review of the SL-5000D. This is the developers version of the SL-5500, which will go on sale early next year. As of this writing, the main differences between the developers version and the consumer version of this device will be the amount of included RAM and different operating system changes / tweaks /additions. The SL-5500 will have 64mb of RAM, while the developers version has 32mb of RAM. The rest of the hardware will be identical between the two devices.

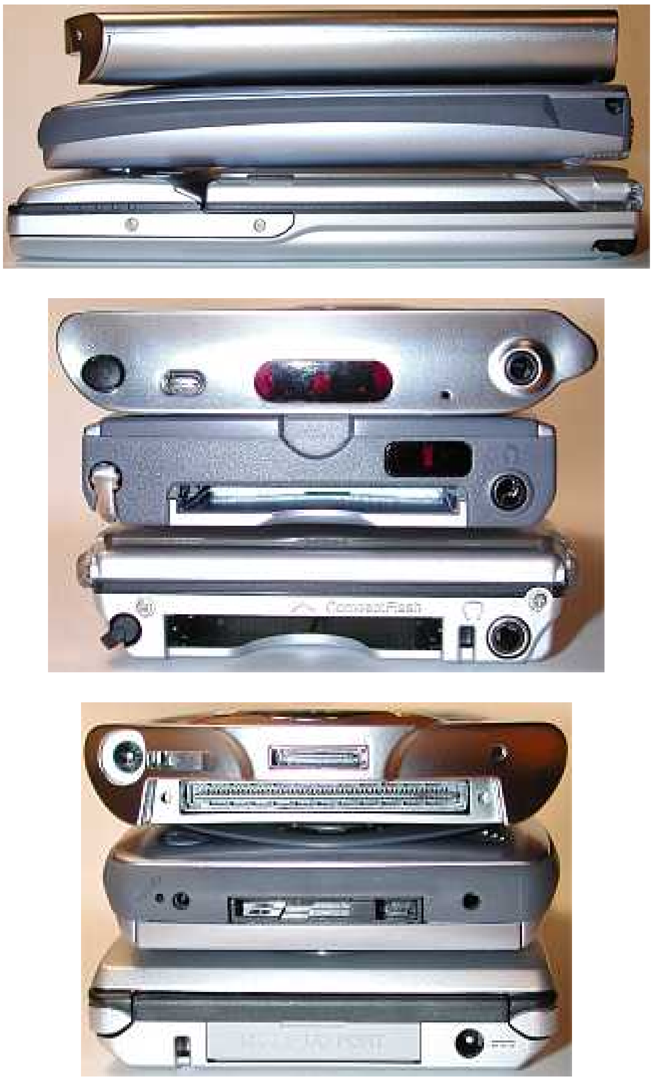
The big difference though will be in the software (operating system). The SL-5000D that I was given still has some rough edges as far as I'm concerned, and I didn't think it would be fair to write a full review on a product that will most likely change quite a bit before it is sold to the general public. So, I have decided to write a strictly hardware review of this device since the hardware features will remain unchanged. Once the consumer version is available, I will update this review to finish it up.

With that out of the way, let's dig into this interesting PDA and check out the hardware specs.

Hardware Specifications:

- Processor: StrongARM (206 MHz 32-bit SA-1110)
- Operating System: Linux 2.4 (Embedix)
- Memory: 32 MB SDRAM, 16 MB Flash ROM,
- Display: 3.5in 240 x 320 pixel, Color Reflective TFT LCD, 16 bits (65,536 colors)
- Power: Removable, rechargeable 3.7V Lithium-Ion battery pack, Built-in 3.0V
- back-up battery, 5.0V AC adapter
- Communications: USB Docking Station, IrDA infrared port
- Expansion slot: One CompactFlash Type I / Type II slot, One Secure Digital slot
- Audio: Stereo headphone jack
- Size: 2.90 x 5.40 x 0.80in. (74 x 138 x 21mm)
- Weight: 7.3 oz (206g)

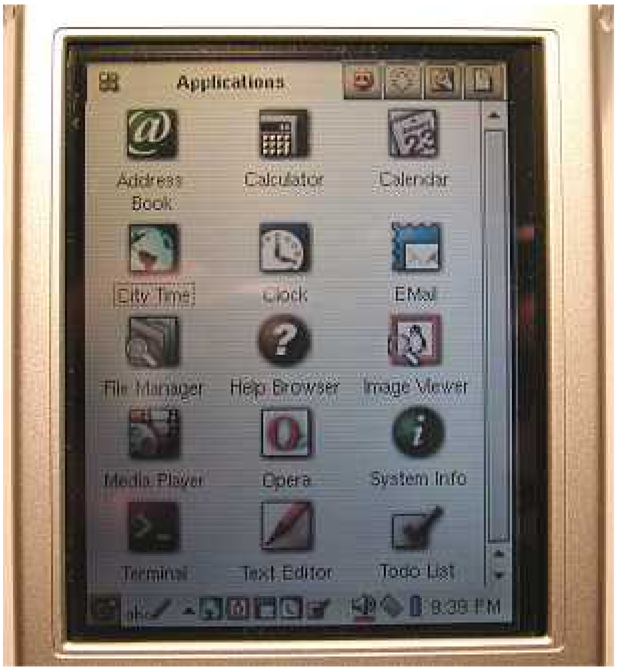
The Zaurus is a very sexy looking device in my opinion. The casing is made of silver frosted plastic that is very similar in color, appearance, and feel to the Casio E-100, 105, 115, and 125 Pocket PCs. The body is very solid and does not creak, crack or flex when squeezed or handled. Size-wise, the Zaurus is bigger and heavier than the iPAQ, but is pretty close to the HP 565 in both overall size and weight. It feels good in my hand and is remarkably small and light considering all of the features that have been packed into it.



(Top to bottom: iPAQ, HP 565, Zaurus)

The Zaurus display is protected by a flip-up translucent frosted lid that is reminiscent of the Palm III series and Jornada 565 style screen covers. The lid opens to a maximum of ~ 135 degrees and can be removed if desired. The Zaurus logo is printed in the middle of the lid.

The display is a 3.5in (diagonal) 240 x 320 pixel, color reflective TFT LCD capable of displaying 65,536 colors. The physical screen size is 2.32 x 2.90in (59 x 74mm) which is the same width as an iPAQ but the same length as a HP 565. However, the screen has a black border around the edges so that the actual viewable / useable area is only 2.1 x 2.7in (53.3 x 69mm). That sounds small, but in everyday use, the screen 'feels' big enough to me.



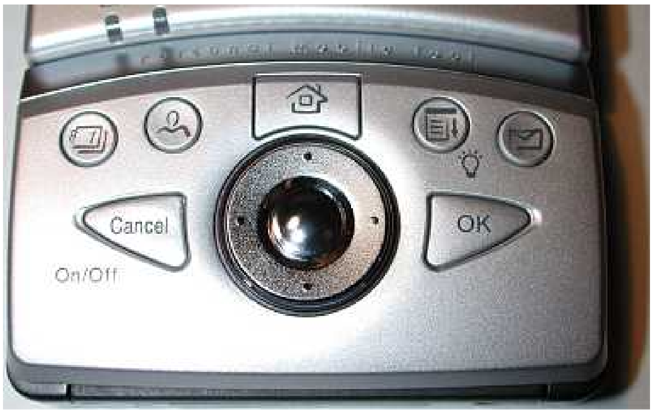
I compared the Zaurus screen to the HP 565 and the iPAQ 3670 screens. Viewing the same JPG image on all three devices, the Zaurus seems to have the richest colors. It also is similar to the HP display in that it has a 'warm up' period. When you first turn the PDA on, the display is not as bright as it is will be after being on for several seconds. I don't notice this on iPAQ displays. Other than that, the display looks good indoors and outdoors in full sunlight. It also has a slick texture so that it is easy to tap and write on with a stylus.

(Left to right: HP 565, Zaurus, 3670 iPAQ)

You might be wondering if the Zaurus is another PDA with the dreaded dust affliction. I'm sorry to say that it might be. The first unit that was given to Judie had several large specks that were clearly visible. At the moment, mine appears to be clear.

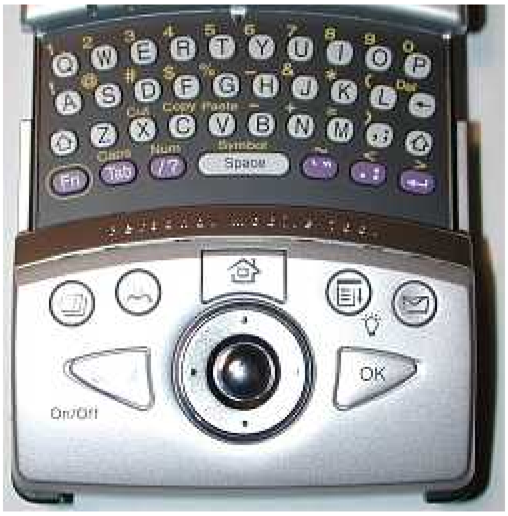
Below the display are two LED indicators. The leftmost LED is for email notification and the right LED is for battery charging status. The email LED glows green during email operations. I was unable to test email operations for this review. The battery status LED glows amber while charging and attached to the AC adapter. This LED will turn off once the battery is fully charged.

Next we have the application button area. Wow, the Zaurus has more buttons on the front than any other PDA that I can think of. The top row of buttons launch the Calendar, Address Book, Home, Menu / front light toggle, and E-Mail applications. Below the top row of buttons is the On/Off button which doubles as a Cancel button, the Cursor pad, Select button, and the Ok button. The small round buttons are slightly concave and sit a little higher than the casing around them. The other buttons are more convex. They all have great tactile feedback so you don't have to guess when you've pressed them. The Cursor / Select button is just plain great. It is a two piece button with the outside being the cursor control and the inside being the select button. The outside collar can be pressed in 4 directions. The inside button is used as a Select. This cursor / select combo button is my favorite style of all the Pocket PCs that I've used to date. It is a good size and is easy to manipulate with one hand.



There is one thing that I hope they 'fix' with the consumer version. When you press the On/Off button, there is a slight pause and then the Zaurus will power on with the front light turned off. After 1 or 2 seconds, the front light will then come on. This only takes 2-3 seconds total, but it just doesn't feel right after using Pocket PCs and Palm devices that have no lag in powering on and off. Same goes for turning the unit off. You have to actually hold down the On/Off button for a couple seconds. Then the unit will click, the front light will turn off and then the unit will power off.

Of course, you're wondering about the built-in keyboard right? Well, hidden under the application and cursor pad buttons, is where you will find the yummy candy center. To get to it, you can grip the ridges on the sides with one hand and pull down, or you can use the tips of your thumbs to pull down the sliding cover. The cover slides down and clicks securely in place revealing a nifty thumbtype-style keyboard underneath.



This keyboard has 37 hard white and purple plastic keys that are arranged in the typical QWERTY format and give really good tactile feedback when pressed. The easiest way to use this keyboard is to cradle the Zaurus between your two your hands and use your thumbs to press the keys. This works remarkably well and allows for pretty quick and reasonably comfortable typing. I think this keyboard feels a little easier to use than the RIM Blackberry 950; I would always seem to get cramps in in my hands while using it. I didn't have this problem with the Zaurus. Probably because there is more to hold on to.

While you are using the keyboard, the application buttons and cursor pad are still active. I actually found out that you can use the cursor pad in conjunction with the SHIFT keys to select blocks of text. You can then use the FN C and FN V key combinations to copy and paste text. Another interesting feature that I came across is the fact that the Select button (middle of the cursor pad) can function the same as the Space key.

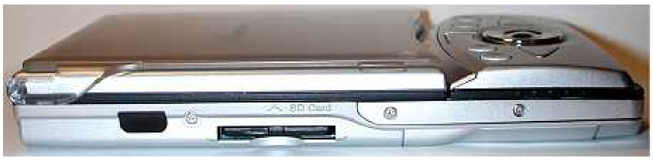
There are a couple little things that I don't like about the keyboard though. One is that the keys are hard and slick. I think rubber keys might have been a better choice. I tend to use my thumbnails to press the keys and sometimes I slide off. I also don't like the location of the "?" key. It is on the left side of the bottom row. For touch typists like myself, this is the opposite of where I'm used to it being located. Other than those two little personal annoyances, I find the addition of the keyboard to be terrific! It gives people yet another method of inputting data, and I'm all about options.

The bottom of the Zaurus has the AC adapter port, lanyard hitch and serial connector. The AC adapter can be plugged directly into the bottom of the PDA for charging, or it can be plugged into the cradle so that when the Zaurus is in the cradle, it will charge thru it.

The lanyard hitch location is actually one of two. There's another one at the top of the unit. Lanyards, which are more popular in Japan than the US allow you to attach a hand strap to the PDA so that you can wear it around your neck (not comfortable!) or as a hand grip. Unfortunately, there wasn't a lanyard included with the Zaurus.

The serial connector has a plastic door covering it that can be opened and slid into the bottom of the PDA. You have to uncover the connector when you want to charge or sync in the included cradle. The cradle is a light weight blob of plastic. It really doesn't differ from most USB PDA cradles. There is a sync button on the front and an AC adapter connector on the back. There is also an I/O port on the back which I'm not exactly sure of its function. I'll assume that it could be used for an optional serial cable for those people that don't want to use USB.

The left side of the Zaurus is the location of the IR port and Secure Digital (SD) / MultiMedia (MMC) card slot. I'm not too thrilled with the IR port location, I much prefer it to be at the top of the unit. Having it on the side makes it harder to beam data to people and to use it for such things as a TV remote control. At the time of this review, I was unable to test the IR strength due to the fact that none of the built in apps actually support it yet.



The SD / MMC slot is one of two expansion slots on the Zaurus. It's great that it has been included because it allows you to use this slot for memory expansion while using the CF slot for other things such as CF wireless cards, modems, etc. The slot itself is spring loaded. You just press the card into the slot and it locks in place. Press it again and it ejects.

The other expansion slot is the Compact Flash slot which is on the top of the unit. This is a Type I/II slot, so you can use the thicker MicroDrives. I tested several of my CF cards, including a 256mb Mr. Flash card, and they all worked just fine with the Zaurus.

Also at the top of the PDA is the stylus silo, a lanyard hitch and the earphone jack. The stylus silo is the typical friction insert type. The stylus is also your run of the mill plastic toothpick. This one is somewhat shorter and fatter than average though.

The earphone jack is a standard 3.5mm sized jack that actually does dual duty. Besides being a stereo earphone jack, it is also a mono microphone jack. You will need to buy a combination earphone / mic device to take advantage of this feature though. Unfortunately, I was unable to test the recording quality because the software does not support it at the moment.

Unlike the current crop of Pocket PCs, the Zaurus can not play MP3's or other 'real' audio thru the internal speaker. You must listen thru headphones. The internal speaker is a piezo buzzer which means that it will really only play beeps, boops and clicks. To me this is very disappointing because I want to be able to be able to set alarm sounds that are more interesting then the lame phone ringer one that is included. I also want to play games that have great sound without having to wear earphones. That said, the stereo output thru headphones sounds great! I think the Zaurus has the best sound quality for playing MP3's of any PDA that I've tested so far. The volume level is also quite good. Compared to my HP 565 and iPAQ 3670, it is about 10% louder than the HP and about 20% quieter than the iPAQ. I never listen to MP3's on the highest volume setting anyway, so I find the levels to be perfect.

The back of the Zaurus is plain except for the removable lithium-ion rechargeable battery pack and the battery replacement switch. The switch is a lock for the battery cover and also functions as a soft reset switch. If you take off the cover, you then will see the battery and a full reset switch. Pressing the full reset switch will erase any information that you have saved directly on the PDA.

There is also a built-in rechargeable back-up battery inside the Zaurus. It isn't something that can be replaced though like a coin cell. It prevents the memory contents from being erased when you replace the main battery. It is charged along with the main battery when in PDA is in the cradle and attached to AC power.

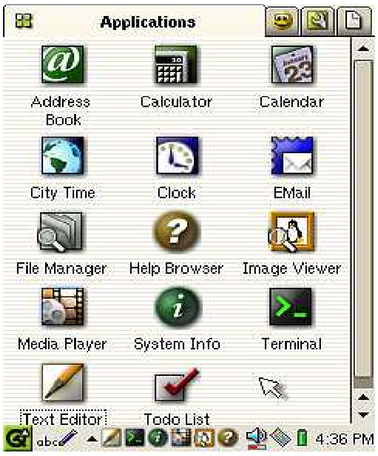
So far, I've noticed that battery life is close to that of my 3670 iPAQ. It really depends on what you use the PDA for as to how much life you should expect per charge. Just playing MP3's yielded me approximately 3hrs of use. By using the MENU button, you can turn off the display's front light, but there isn't a feature to turn the display totally off like you can on a Pocket PC.

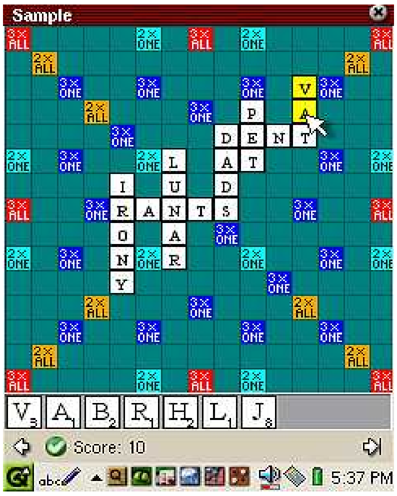
Overall system speed seems to be on par with other StrongArm devices such as the iPAQ and HP 565. I took a few minutes to sit and open apps one after another on my iPAQ and HP and then did the same with the Zaurus. I didn't notice any real differences in launch speed between all of the devices.

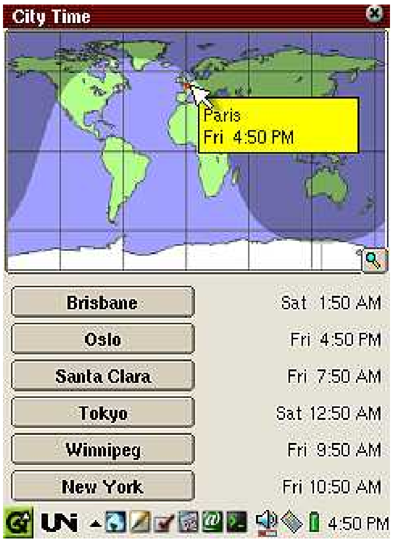
As far as the hardware goes, Sharp has a real winner in the Zaurus SL-5000D. It is a solid PDA packed with great features, while not being overly bulky. Including both CF and SD/MMC slots is a big plus, and the built-in keyboard gives this device a real gee whiz flare. If I had to change anything regarding the hardware, it would be to give the Zaurus a real internal speaker on par with the Pocket PC, move the IR port to the top of the unit and take away the lag with powering the unit on and off. Other than that, I could honestly say that I would love for this device to be my main PDA. But, there's that little matter of the operating system....

The OS as it is on this developers unit is still rough. It doesn't feel as polished as the Pocket PC or Palm. The main PIM apps just aren't ready for the masses as far as I'm concerned. They seem flat and sorely lacking in advanced features. However, I do like the style of the interface. For Linux users, it has that KDE look to it and is called Qtopia from Trolltech. Qtopia is the GUI and a core set of applications which include: an Address book, To-Do List, Appointment Calendar, E-Mail client, Opera Web Browser, a multimedia player capable of playing MPEG1, MPEG2, and MP3 format files, image viewer, Command Line Terminal and File Manager, Text Editor, Calculator, City Time app, and several games, including Asteroids, Go, Mindbreaker, Mine Hunt, Patience, Snake, Tux and Word Game (Judie and I both loved this Scrabble clone!).

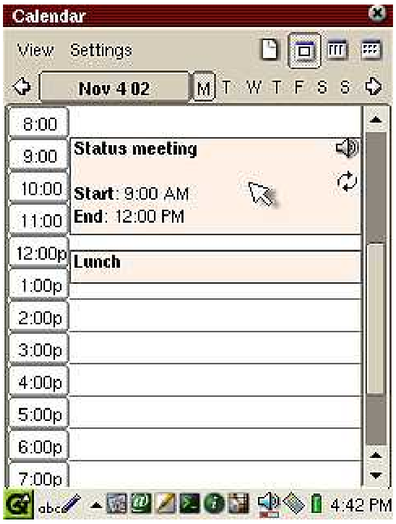
Below are some screenshots that I lifted from Trolltech's website. Visit it to see more.






City Time

Brisbane	Sat 1:50 AM
Oslo	Fri 4:50 PM
Santa Clara	Fri 7:50 AM
Tokyo	Sat 12:50 AM
Winnipeg	Fri 9:50 AM
New York	Fri 10:50 AM


Calendar

8:00
9:00 Status meeting
10:00 Start: 9:00 AM
11:00 End: 12:00 PM
12:00p Lunch
1:00p
2:00p
3:00p
4:00p
5:00p
6:00p
7:00p

It's always fun to play with a new PDA, and this one was no different in that respect. But after playing with it for several hours, I realized that it couldn't be my main PDA until the software becomes a little more robust. Sure, if this device takes off, there is going to be a large community of developers that will rally behind it and create better applications. Right now, it has a big appeal for hackers and Unix geeks which is great, but it doesn't feel like a consumer device for the average person. If this device were running Pocket PC 2002, I would say that Sharp might have created the next golden child of the PDA world. But since they decided to go with Linux and Java, it makes me wonder if they will be able to succeed. The Pocket PC is only just now gaining on Palm in the battle of the PDA OS's after being on the frontline for several years now. Bringing yet another OS into the fray seems like a mistake to me. But I will withhold my judgment until I see the actual consumer version. I will be anxiously waiting to get my hands on one to review.

Price: $399

Pros:
- Built-in keyboard
- SD/MMC and CF slots
- Great stereo output thru headphones

Cons:
- Internal speaker inferior. Needs earphones to listen to MP3's etc.
- Operating system needs work
- Pause in powering up and powering down
- Core apps need more polish and features

This article is re-printed with permission. The originals can be found at:

http://www.the-gadgeteer.com/zaurus-sl-5000d-review.html

New UCITA Revisions – First Reactions

Cem Kaner <kaner@kaner.com>

A few months ago, Professor Phil Koopman, Sharon Roberts, Professor Don Gotterbarn and I went to the 17th meeting of the Uniform Computer Information Transactions Act drafting committee (I've attended 16 of these meetings).

The drafting committee is under intense pressure to work a political compromise, because, after passing in Virginia and Maryland, UCITA has been rejected in every state that has considered it and three states have passed "bomb shelter" laws designed to keep UCITA-governed contract rules out of their states. Additionally, the National Association of Attorneys General recently published a letter (signed by 33 Attorneys General) saying that UCITA is so fundamentally flawed that it should be abandoned rather than amended. Additionally, the UCITA process is under the scrutiny of a Task Force appointed by the American Bar Association. The ABA has not yet committed itself for or against UCITA. Some of its Sections (comparable to SIGs) appear to favor UCITA; others appear to oppose it. One of the

Sections actively opposes UCITA and triggered the study by the ABA. It is likely that UCITA will have no further legislative success if ABA recommends against its adoption.

The committee met privately, after the official meeting, and adopted 19 of the amendments.

A couple of things that I was advocating were passed, especially a ban on "self-help" (ability of a vendor to remotely shut down your system if there's a contract dispute between you and the vendor). This shuts down a serious security flaw that UCITA was encouraging large-system vendors to build into every significant piece of commercial software.

Here is my analysis of the amendments that were passed. Overall, I think we are still seeing a big trend favoring large companies over small companies and individuals. In this case, though, large customers are scoring some wins and smaller customers are picking up a little bit as a side-benefit.

The National Conference on Uniform State Laws published an announcement today of 19 amendments to UCITA. These were written in response to a series of amendments proposed at the UCITA drafting committee meeting this November. These amendments are available at http://www.nccusl.org/nccusl/UCITA-2001-comm-fin.htm For the text of UCITA, see http://www.law.upenn.edu/bll/ulc/ucita/ucita01.htm.

For a detailed analysis (of mine) of UCITA, see http://www.badsoftware.com/engr2000.htm

The Attorneys General letter is at http://www.affect.ucita.com/pdf/Nov132001_Letter_from_AGs_to_Carlyle_Ring.pdf

Here are my first impressions of those amendments. Please feel free to circulate them.

1) Consumer protection

UCITA defines the typical consumer software transaction as an intangible license, the purchase of a right to use the software, rather than the sale of a copy of the software. So, when you buy a copy of Microsoft Word and a book on how to use Microsoft Word at your local computer store, you buy two things that contain copyrighted intellectual property. The sale of the book is a sale of goods under UCITA but under UCITA, the sale of the software is not. If you download that same book from Barnes & Noble, instead of buying the paper copy at Barnes & Noble, the book is treated like software under UCITA.

By defining consumer purchases of software as licenses, rather than sales, UCITA pulls consumer software out of the scope of all of the consumer protection statutes that protect buyers of "consumer goods." All of the consumer warranty laws, for example, are "consumer goods" laws.

The revisions to UCITA still pull software outside of the scope of the consumer warranty laws. The changes offer very little protection.

2) E-SIGN

In the second amendment, UCITA supercedes E-SIGN, except in certain listed sections. In general, I think that E-SIGN is more consumer-friendly than UCITA. I have not had time to analyze the new relationship between the two statutes.

3) Choice of Forum

The change proposed will make it slightly harder for vendors to make an outrageous choice of forum (where the customer must sue the vendor, if the customer wants to bring suit).

4) Electronic Self-Help

I am glad to see that UCITA has been revised in the way that Sharon Marsh Roberts (Independent Computer Consultants Association) and I recommended, with the support of the Society for Information Management. Electronic self-help is banned, but a vendor retains extensive power to protect its rights under UCITA. For example, the software can come with a built-in automatic termination, stopping performance after a specified number of days or uses. In the event of a dispute, the vendor can simply refuse to renew the license. The vendor can also get an injunction.

5) Public Criticism & Contract Laws

The amendment (section 105(d)) appears to address the public criticism issue, but leaves open a wide loophole. People are allowed to criticize a product that has been "offered in its final form to the general public." But anything that is not "in its final form" is not open to criticism. Let's consider Viruscan, published by McAfee. McAfee has issued licenses that ban publication of benchmarks or other reviews of Viruscan without McAfee's permission. Viruscan is updated frequently. I don't think it is ever in "final form." So it appears to be outside of the scope of this consumer protection. Anything that is sold with the promise of frequent automatic updates (think of the dot-NET business model) is, arguably, never in its "final form". Any vendor who wants to ban criticism of its products has an obvious way around 105(d).

6) Known Defects

This amendment specifically states that UCITA does not displace the laws of "fraud, including fraudulent inducement, misrepresentation, or unfair and deceptive practices." This amendment does nothing whatsoever. UCITA already does not displace these laws. To the best of my knowledge (which is fairly extensive on this point), every software publisher in the United States releases software with known defects, and many of those known defects are serious. It is very difficult to hold vendors accountable for this under current law. UCITA shields vendors further, by making it easier for them to disclaim warranties, harder for a customer to establish that a product demonstration upon which the customer relied actually created an express warranty, easier for the vendor to limit remedies, and harder for the customer to recover a "minimum adequate remedy.

What was proposed, time after time after time in the UCITA meetings, was that the drafting committee provide an affirmative incentive to manufacturers to reveal their known defects. This was in return for the many vendor protections being written into the statute. This amendment does not address that proposal and is no better than the unmodified UCITA.

7) Presentation of Later Terms

"Later terms" are contract terms that you see only after you pay for the product.

Amendment 7, new Section 216, appears to add nothing to UCITA's rules. The question is not whether some of the terms in the click-wrapped licenses will be enforced. Most people know that some contract terms will be presented in the box in some form or another. The question is which terms will be enforced and how much notice customers will have of those terms.

The new UCITA requirement is satisfied merely by putting a notice on the box that says, "Terms inside" or a statement when you start to download a product that contract terms will be presented later. This is trivially easy to satisfy. The only people who will have difficulty satisfying it will be the open source / free software community because so much of their software is already circulating and will continue to circulate. That software was not packaged in a way that will meet the new, fairly formal, UCITA requirements.

What was repeatedly requested was a requirement that customers could get a copy of the terms before the sale if they asked for the copy. This is one of the basic tenets of the consumer warranty laws that UCITA helps software publishers evade.

Under this amendment, customers will still have to pay for the software and start installing it (if that's how the vendor chooses to structure the deal, which most software vendors seem to want to do) before being able to discover the terms of the contract.

The "right of return" under UCITA is the same extremely weak "right" that it was before, more marketing fluff than a consumer benefit. Remember: even though this is promoted regularly as a consumer benefit, it was brought to the UCITA drafting committee by the representative of the Business Software Alliance and it has (to the best of my knowledge) never been endorsed by any consumer protection advocate.

8) Retention of Terms

Amendment 8 provides that the license must be provided to the customer in a form in which it can be printed and/or retained by the customer. That this is an improvement on the current UCITA is an illustration of the extent to which the current UCITA is poorly drafted. Of course the customer is entitled to a copy of the license that can be printed and retained. How can you hold the terms of a license against someone who can't even refer to it? What court would enforce the terms of a contract that the customer is allowed to see once and never again? Vendors need this rule as much as customers. Without it, they might sometimes be tempted to make terms irretrievable or to allow a product to ship with terms that happen to be irretrievable. In either case, they would face severe problems in the courts under current law, (including UCITA) because judges would be so unlikely to enforce such terms.

9) Open Source Software--Noncontractual Permissions

As the Reporter of the UCITA Drafting Committee pointed out in the November meeting, UCITA already does not cover permissions that are not intended as contracts. However, all of the open source and free software licenses / permissions that I have seen are in fact contracts. This amendment provides zero or almost zero protection to the Open Source / Free Software communities.

10) Warranties for "Free" Software

UCITA provides an important protection for free software and broadens it in a way that will also often serve vendors of non-free commercial software. It eliminates warranties for software when there is "no contract fee for the right to use, make copies of, modify, or distribute" the software. The critical word here is OR, which should be AND. With the OR in place, the vendor need only satisfy one of these conditions in order to claim that the software is free.

Here's an example: under this new definition of free software, Internet Explorer is free software because there is (currently) no contract fee for the right to use the software. That's all that is needed. You don't have to have the right to make copies of the software or modify it or reverse engineer it or obtain source code to it or distribute it, as long as you get a free right to use it.

So, if Vendor X sells you installation and support services and "throws in" the software "for free", the Vendor achieves free software status and no warranties apply. This is an easy way for a traditional software vendor to escape all warranty liability.

Warranty liability cannot be excluded, under this amendment, if the licensee is a consumer. Thus, genuinely free software is fully subject to consumer warranties. This is still going to be a big problem.

A point was made at the UCITA meeting that no one would sue free software developers anyway, because they don't have any assets. But universities and libraries and many businesses post free software at their websites. That makes them distributors, under UCITA, even if they are giving away software that was written to be given away. Universities, libraries, and many businesses do have deep pockets (i.e. they have insurance policies) -- if a credible threat of liability can be made against them, they will stop distributing free software.

So, what do we have? Microsoft gets to completely avoid warranty protection for business users of some of its products, and organizations that distribute free software (which Microsoft now appears to consider a competitive threat) can still be targeted for consumer lawsuits and thus might be successfully intimidated out of distributing the free software.

This is not a victory for the Free Software community.

We could solve part of this problem by fixing the definition of "merchant." George Graf (one of the ABA Advisors who helped write UCITA) had an important idea, and I was surprised not to see this amendment. He said that we should change the definition to merchant to be someone who is in the business of licensing software. I like this, but it might exclude consultants too much. Here's a slight variation that I think should be adopted:

(46) "Merchant" means a person that received consideration in this transaction or a transaction associated with this one:

(A) that deals in information or informational rights of the kind involved in the transaction;

(B) that by the person's occupation holds itself out as having knowledge or skill peculiar to the relevant aspect of the business practices or information involved in the transaction; or

(C) to which the knowledge or skill peculiar to the practices or information involved in the transaction may be attributed by the person's employment of an agent or broker or other intermediary that by its occupation holds itself out as having the knowledge or skill.

Supplement this with a Comment that public institutions and others who are not the developers and are also not receiving fees for distribution should not be warrantors in a consumer transaction.

11) Transfer

Software that comes with a computer can be transferred WITH THE COMPUTER as a gift to a library or K-12 school or from one consumer to another. This still allows the vendor to kill the market in used software and it allows only a minimal number of transfers of software. The general rule under UCITA will be that if you buy a copy of the software, you will not be able to sell it when you are done with it, or give it away unless you are willing to give away your computer with it.

12) Express Warranty by Sample, Model or Demonstration

This amendment improves the current UCITA by stating that the product must conform (rather than "reasonably conform") to the sample, model or demonstration. However, even as modified, UCITA section 402 provides that the following does not create a warranty: "a display or description of a portion of the information to illustrate the aesthetics, appeal, suitability to taste, subjective quality, or the like of informational content." It is not a breach of contract if there are differences in the user interface and usability (or in the aesthetics, appeal, suitability to taste or subjective quality) between the demonstrated model and the model shipped, even if these are material to the consumer.

13) Infringement and Hold Harmless Duties

I'm not sure of the effect of this amendment and therefore will not comment
on it.

14) Implied Warranty Scope

The amendment specifies that the implied warranty runs from the licensor to
ITS end-user licensee and to ITS distributor.

I'm not sure, but it looks to me as though UCITA is re-establishing a privity rule. I am unsure of the intent, but I expect that we will see the argument in court that Vendorsoft provided no warranty to Consumer because Consumer is the licensee of Distributorsoft, who distributes Vendorsoft's software. Given the other sections of UCITA, I don't think this argument would prevail, but if it is not to make room for an argument like this, I don't understand why this restrictive language is here.

15) Delete Section 308

In Section 308, current UCITA allows a vendor, after the sale, to terminate a license by determining that the duration of the license, as long as that duration has been "a reasonable time". It was never clear to me that this was a big deal (in comparison to the rules that would apply under Article 2) nor that this deletion offers a big advantage over what the courts will do in the absence of specific terms.

16) Delete Section 307(c)

Current UCITA 307(c) states that "(c) An agreement that does not specify the number of permitted users permits a number of users which is reasonable in light of the informational rights involved and the commercial circumstances existing at the time of the agreement." I'm not sure that deleting this will offer any advantage over what the courts will do in the absence of specific terms.

17) Section 605 Automatic Restraints

This is a clarifying amendment that closes a loophole that was apparently not intended by the drafting committee.

18) Corrects a typo, no policy impact

19) Reverse engineering

This is very narrow and not very useful. It is narrower than the provisions in DMCA that allow reverse engineering. It does not permit reverse engineering in order to detect security holes or defects or to enable repair of the security holes or other defects. Additionally, if "the elements" to be reverse engineered were ever previously "readily available to the licensee" (when he didn't need them) then the licensee can't reverse engineer to discover them now, when he does need them.

K) Scope

As the comments point out, the electronics manufacturers (who will be able to opt their goods within the scope of UCITA under the current scope) support the current scope. And no wonder! They get to apply UCITA's rules to their customers instead of Article 2's.

We proposed a rule that addressed safety-critical software, rather than one that tried to distinguish between embedded and nonembedded software. The drafting committee did nothing to restrain UCITA's application to safety-critical embedded software. Never during the UCITA drafting meetings did we discuss the potential consequences of applying UCITA to embedded software or, especially, safety critical software. There will undoubtedly be unintended consequences of the application of UCITA to this domain. Where lives are involved, I think it is grossly irresponsible to press forward with the application of a new body of law to an ill-considered domain.

This article is re-printed with permission. The originals can be found at:

http://www.interesting-people.org/archives/interesting-people/200112/msg00255.html

The version printed was sent to AUUGN as an update, by the author on Wed, 26 Dec 2001.

The Open Cluster Framework Project

Alan Robertson <alanr@unix.sh>

We are a group of people who are in the process of defining standard clustering APIs for some basic capabilities. At this writing (Jan 2002), this project is in fairly early stages and is very much open to participation.

Who are we?

We are a group of people who have interests in cluster software – both proiders and consumers of clustering services. We have periodic meetings in person, and an ongoing conversations via a mailing list.

Our Approach

We have two basic thrusts to our work:

- Define standard APIs for basic clustering functions
- Create and support an open source development project which acts as the reference implementation for the OCF APIs.

It is also our intent to create APIs which are usable on both High-Performance and High-Availability clusters. It is ***not*** our intent to replace or redefine de-facto standards (like MPI or PVM) which are already in common use, and serve their intended purpose well.

IP stance

It is our stance that the OCF APIs themselves must be royalty-free (RF) standards. It is acceptable for individual implementations to use patented or otherwise encumbered techniques, but the standard itself must be reasonably implementable without the use of patented techniques.

The Standards

We are working towards becoming a working group of the Free Standards Group. The standards themselves are intended to be largely platform-independent, capable of being implemented on most POSIX-compliant OSes, but there will be certain sections (kernel APIs for example) which will be platform-specific. Although the standards are intended to be OS-independent, the primary interest of the majority of the group is Linux, and the OS-specific APIs will likely only be defined for Linux initially.

The Reference Framework Development Project

Areas of Interest (scope)

There are many kinds of APIs which have been or might be defined for services in a cluster. Since our scope is necessarily limited, we are currently only considering working in the following general areas:

- Node services
- Group services
- Resource services
- Lock Services
- External Interfaces

Moreover, the areas we are going to concentrate on first are node liveness and membership, and resource agents. This will help us keep focused and enable us to make good progress while keeping the whole of the task in mind.

These areas are further defined below:

Node Services
Examples of node services which are being considered for standard APIs include:

- Node liveness services
- Node membership services
- Node communication services (relaible, not globally ordered)

Group Services
Examples of group services which are being considered for standard APIs include:

- Group Membership services
- Group communication services (reliable, ordered)
- Group Barriers
- Group Transactions
- Group Voting
- Group membership

Resource Services
Examples of resource services which are being considered for standard APIs include:

- Cluster (resource?) management
- Resource Agents
- Resource monitoring
- Resource fencing
- Remote instantiation (RIF)

Lock Services
Examples of lock services which are being considered for standard APIs include:

- Lock creation
- Lock manipulaiton
- Lock destruction

External Interfaces
Examples of external interfaces which are being considered for standard APIs include:

- User Interface (GUI, CLI, etc.)
- Management (SNMP, CIM, etc) intefaces
- Logging interfaces

More information on some of these areas can be found in Greg Louis' notes from the Enschede clustering workshop. (http://opencf.org/enschede2001/Enschede.summary.txt)

Other Documents

The following documents are likely also of interest.
Draft OCF charter document in PDF (http://opencf.org/OCF.pdf) or StarOffice (http://opencf.org/OCF.sdw) formats.

An outline of the framework development project in HTML, (http://opencf.org/HAFramework.html)
PDF (http://opencf.org/HAFramework.pdf)
or StarOffice formats.
Information on the 2001 Enschede clustering workshop (http://opencf.org/enschede2001/).

Information on the 2001 Ottawa clustering Working Group where the idea for the OCF was first formally presented. (http://opencf.org/ottawa2001/)

A talk on the OCF being prepared for the January 2002 Linux World Conference and Expo in New York City.(http://opencf.org/talks/LWCE-NYC-2002/LWCE-NYC-2002.html)

Free Standards Group policy for forming new working groups. (http://freestandards.org/policy/fsg102-newworkgroup-draft.txt)

Draft answers to the FSG 102-1 questions in html or text. (http://opencf.org/OCF-fsg102-1.html)

OCF Supporters

The following organizations and companies are currently supporting the OCF effort.

- IBM
- COMPAQ
- SGI
- SuSE
- Red Hat Software
- Conectiva
- BigStorage
- MSC Software
- Bald Guy Software
- OSCAR
- Linux-HA

This article is re-printed with permission. The originals can be found at:

http://opencf.org/

Oracle 9i EE Installation on Red Hat Linux 7.1 and on Red Hat Linux 7.2

Author: Werner Puschitz <webmaster@puschitz.com>

Here is a summary (HOWTO) of how I installed Oracle 9i (9.0.1) Database on Red Hat 7.1 (kernel 2.4.2-2, glibc 2.2.2-10) and on Red Hat 7.2 (kernel 2.4.7-10, glibc 2.2.4-13).

For more information regarding configuration and performance check the following links:

- Oracle9i Database Documentation for Linux (Official Oracle documentation, http://otn.oracle.com/docs/products/oracle9i/content.html)
- Oracle Performance Tuning on Linux (Part I) (Simple ways to achieve Oracle performance improvements, http://www.linuxjournal.com//article.php?sid=5840)
- The RAW Facts on Filesystems (Part II) (Ways to achieve Linux performance improvements for databases in general, http://www.linuxjournal.com/article.php?sid=5841)

NOTE: Red Hat 7.1 has been validated for Oracle9i Database and for Oracle9i Application Server, see " Red Hat Announces Validation of Red Hat Linux For Oracle ." But as of February 2002, there has been no validation for Red Hat 7.2 yet.

Downloading and Installing Red Hat Linux 7.1 and/or 7.2

To download Red Hat Linux, check the links at http://www.puschitz.com/RedhatDownload.html

You can find the installation guides for installing Red Hat Linux under Red Hat Linux Manuals.

Unpacking Downloaded Oracle9i Installation Files and Burning Oracle9i CDs on Red Hat Linux 7.1 and 7.2

Download Oracle9i for Linux from the following web site: http://otn.oracle.com/software/products/oracle9i/htdocs/linuxsoft.html

Oracle does not provide the checksums for these files to make sure if the downloaded files are ok. I successfully decompressed (gunzip) and extracted the downloaded files, and here are the MD5 checksums I got:

```
$ md5sum Linux9i_Disk1.cpio.gz Linux9i_Disk2.cpio.gz
  Linux9i_Disk3.cpio.gz
f1a99eb8c8aca1d69a9eeaa8858570d7  Linux9i_Disk1.cpio.gz
f2444c0fa53c898e7d2f78c184829d7d  Linux9i_Disk2.cpio.gz
ec655402d8bc547ed031f14122da574b  Linux9i_Disk3.cpio.gz
```

Now uncompress and unpack the downloaded files. There are two ways to do this:

One step procedure (uses less disk space and is faster):

```
zcat Linux9i_Disk1.cpio.gz | cpio -idmv
zcat Linux9i_Disk2.cpio.gz | cpio -idmv
zcat Linux9i_Disk3.cpio.gz | cpio -idmv
```

Two step procedure:

```
# Uncompress
gunzip Linux9i_Disk1.cpio.gz Linux9i_Disk2.cpio.gz
    Linux9i_Disk3.cpio.gz

# Unpack the downloaded files:
cpio -idmv < Linux9i_Disk1.cpio
cpio -idmv < Linux9i_Disk2.cpio
cpio -idmv < Linux9i_Disk3.cpio
```

Now you should have 3 directories containing installation files:

```
Disk1
Disk2
Disk3
```

I executed the following commands to burn the 3 CDs with my external USB CD Burner HP-8230e:

```
mkisofs -r Disk1 | cdrecord -v --eject dev=0,0,0
speed=4     -
mkisofs -r Disk2 | cdrecord -v --eject dev=0,0,0
speed=4     -
mkisofs -r Disk3 | cdrecord -v --eject dev=0,0,0
speed=4     -
```

(You can get the dev numbers when you execute cdrecord -scanbus).

Swap Space

In order to perform a typical Oracle 9i installation and to create a simple prototype database, Oracle says that you need a minimum of 512MB of RAM for the Oracle9i Server, and the amount of disk space (swap space) should be equal to twice the amount of RAM or at least 400 MB, whichever is greater.

When I installed Oracle 9i, I used 600 MB of swap space on a PC with 256MB of RAM, which worked fine. But when I used less swap space I ran out of memory. I definitely would recommend to use more RAM and/or more swap space, especially when you have other programs running on your Oracle server.

NOTE: ***If you do not have enough swap space or RAM during the Oracle installation, in particular during the database creation, your Oracle server will temporarily become unresponsive to any events for several minutes.***

Check your memory by executing:

```
grep MemTotal /proc/meminfo
```

Check swap space by executing:

```
cat /proc/swaps
```

You can also add temporary swap space by creating a temporary swap file instead of using a raw device. Here is the procedure:
As root:

```
dd if=/dev/zero of=tmpswap bs=1k count=900000
chmod 600 tmpswap
mkswap tmpswap
swapon tmpswap
```

/tmp Space

The Oracle Universal Installer requires up to 400 MB of free space in the /tmp directory. If you do not have enough space in the /tmp directory, you can temporarily create a tmp directory in another filesystem. Here is how you can do this:

As root:

```
mkdir /<AnotherFilesystem>/tmp
chown root.root /<AnotherFilesystem>/tmp
chmod 1777 /<AnotherFilesystem>/tmp
export TEMP=/<AnotherFilesystem>/tmp
    # used by Oracle
export TMPDIR=/<AnotherFilesystem>/tmp
    # used by Linux programs like the linker "ld"
```

When you are done with your Oracle installation, shutdown Oracle and remove the temporary directory:

```
rmdir /<AnotherFilesystem>/tmp
unset TEMP
unset TMPDIR
```

Oracle Disk Space

You will need about 2.5 GB for the database software. If you perform a typical database installation and not a customized database installation, then you will need about 3.5 GB of disk space.

"binutils" Issue

The binutils package that comes with Red Hat 7.1 and 7.2 doesn't work with Oracle 9i Universal Installer. No new version of binutils seems to work (e.g. you will fail with binutils-2.11.90.0.8-9).

You have 2 options:

Wait for the following Oracle installation error to make a minor change in an Oracle file (it's very easy): ***"Error invoking target install of makefile /opt/oracle/product/9.0.1/plsql/lib/ins_plsql.mk"*** See Running Oracle Installation and Oracle Installation Errors (below) for more information. I recommend this approach. This obviates the need to change your binutils at all.

Download the following binutil RPM version and downgrade binutil on the Oracle server: ftp://ftp.redhat.com/pub/redhat/linux/7.0/en/os/i386/RedHat/RPMS/binutils-2.10.0.18-1.i386.rpm
As root:

```
rpm -Uvh --force --nodeps binutils-2.10.0.18-
1.i386.rpm
```

When you are done with the Oracle installation, you upgrade your binutil RPM back to the version you had before you downgraded:

E.g. on my Red Hat 7.2 server:

```
rpm -Uvh --force --nodeps binutils-2.11.90.0.8-
9.i386.rpm
```

I do not recommend this approach.

Install JDK

Download JDK 1.3.1 or Blackdown 1.1.8_v3: (I always use Blackdown)

http://www.blackdown.org
http://java.sun.com

According to JDK documentation, install JDK under /usr/local .

Then create a symbolic link to the JDK under /usr/local/java :

As root:

```
bzip2 -dc jdk118_v3-glibc-2.1.3.tar.bz2 | tar xf
-  -C  /usr/local
ln -s /usr/local/jdk118_v3 /usr/local/java
```

Create Oracle User Accounts

As root:

```
groupadd dba
groupadd oinstall
useradd -g oinstall -G dba oracle
passwd oracle
```

For more information on the "oinstall" user account, see ***When to use "OINSTALL" group during install of Oracle*** (http://metalink.oracle.com/oracleinstall/oracle8i/genericunix.html#Uoui)

Create Oracle Directories

As root:

```
mkdir /opt/oracle
mkdir /opt/oracle/product
mkdir /opt/oracle/product/9.0.1
chown -R oracle.oinstall /opt/oracle
mkdir /var/opt/oracle
chown oracle.dba /var/opt/oracle
chmod 755 /var/opt/oracle
```

Set Oracle Environments

As oracle: (in e.g. ~oracle/.bash_profile)

```
# Oracle Environment
export ORACLE_BASE=/opt/oracle
export ORACLE_HOME=/opt/oracle/product/9.0.1
export ORACLE_SID=test
export ORACLE_TERM=xterm
```

```
#export TNS_ADMIN= Set if sqlnet.ora,
tnsnames.ora,
 etc.   are   not    in
$ORACLE_HOME/network/admin
export NLS_LANG=AMERICAN;
export
ORA_NLS33=$ORACLE_HOME/ocommon/nls/admin/data
LD_LIBRARY_PATH=$ORACLE_HOME/lib:/lib:/usr/lib
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/usr/local/lib
export LD_LIBRARY_PATH
# Set shell search paths
export PATH=$PATH:$ORACLE_HOME/bin
# CLASSPATH:
CLASSPATH=$ORACLE_HOME/JRE:$ORACLE_HOME/jlib:$ORAC
LE_HOME/rdbms/jlib
CLASSPATH=$CLASSPATH:$ORACLE_HOME/network/jlib
```

Start runInstaller

Oracle no longer supports a character mode installer. Therefore, in order to execute runInstaller directly from a console of a machine you are logged into (in this example the node name where Oracle is running is called "oracleserver"), you need to set the DISPLAY environment variable. Before you do that, make sure you also allow runInstaller on " oracleserver " to display X information to your Linux desktop machine (in this example, the PC name where you are running X Windows like KDE or GNOME is called "yourdesktop"), because programs running on remote machines cannot display information to your screen unless you give them the authority to do so. Note that the X display relink mechanism does not work for NT desktop machines unless you use Exceed.

If you install Oracle on your desktop PC and not on a remote node, then you can skip step 1 and 2.

Step 1: E.g. allow "oracleserver" to display X information to your desktop PC "yourdesktop":

```
yourdesktop:user$ xhost +oracleserver
```

Step 2: From the console of your Oracle server "oracleserver" you are logged into, execute the following command as user " oracle ":

```
oracleserver:oracle$ export
DISPLAY=yourdesktop:0.0
```

Step 3a: From your burned CD Disk 1, execute runInstaller (do not cd to /mnt/cdrom !):

As root:
```
oracleserver:root# mount /mnt/cdrom
```

As oracle:
```
oracleserver:oracle$ /mnt/cdrom/runInstaller
```

Step 3b: Or wherever you unpacked your downloaded files:
```
oracleserver:oracle$ Disk1/RunInstaller
```

Running Oracle Installation

This is how I answered the questions in the runInstaller:

- What would you like as the base directory (Inventory Location):

```
/opt/oracle/oraInventory
```

UNIX Group Name (permission for updating Oracle software):
```
oinstall
```

You could also use "dba" which I do not recommend for security reasons. For more information on the "oinstall" user account, see

When to use "OINSTALL" group during install of oracle.

- Full path name of the Oracle Home:

```
/opt/oracle/product/9.0.1
```

- JDK Home Directory:

```
/usr/local/java
```
etc.

NOTE:
If you did not downgrade the binutils package, which I recommend (see "binutils" RPM Issue), then you will get the following error message when the Oracle installer is at the third Oracle CD:

"Error invoking target install of makefile /opt/oracle/product/9.0.1/plsql/lib/ins_plsql.mk"

To solve this problem, see Oracle Installation Errors for more information.

Sometimes the "Oracle Net Configuration Assistant" will hang, see Oracle Installation Problems, Important Tips and Hints for more information.

I would recommend that you also check the other issues at Oracle Installation Problems, Important Tips and Hints and Oracle Installation Errors .

Startup and Shutdown of Oracle 9i Database

sqlplus:

svrmgrl is not supported any more. You can now do everything with sqlplus.

E.g., to startup the database, execute the following commands:

```
dba$ sqlplus /nolog
SQL> connect / as sysdba
SQL> startup
```

The slash connects you to the schema owned by sys. And as far as I know, "sysdba" gives you the following privileges:

```
sysoper privileges WITH ADMIN OPTION
create database
recover database until
```

```
$ORACLE_HOME/bin/dbstart and
$ORACLE_HOME/bin/dbshut :
```

You can also use $ORACLE_HOME/bin/dbstart to startup the database, and $ORACLE_HOME/bin/dbshut to shutdown the database. You caN place $ORACLE_HOME/bin/dbstart into the

/etc/rc.d/rc.local boot script to automatically bring up the database at system boot time. To get $ORACLE_HOME/bin/dbstart and $ORACLE_HOME/bin/dbshut working, you need to change the third field for your Oracle SID in /etc/oratab from "N" to "Y".

E.g. for the Oracle SID "test" I changed the line in /etc/oratab from test:/opt/oracle/product/9.0.1:N to read test:/opt/oracle/product/9.0.1:Y

Oracle Installation Problems, Important Tips and Hints

If you are having problems with gunzip on any of the downloaded files Linux9i_Disk1.cpio.gz,Linux9i_Disk2.cpio.gz , or Linux9i_Disk3.cpio.gz, then try to run MD5 checksum on these files with md5sum. I successfully unzipped and extracted these files, and my downloaded files have the following checksums:

```
$ md5sum Linux9i_Disk1.cpio.gz
Linux9i_Disk2.cpio.gz
   Linux9i_Disk3.cpio.gz
f1a99eb8c8aca1d69a9eeaa8858570d7
Linux9i_Disk1.cpio.gz
f2444c0fa53c898e7d2f78c184829d7d
Linux9i_Disk2.cpio.gz
ec655402d8bc547ed031f14122da574b
Linux9i_Disk3.cpio.gz
```

Do not cd to /mnt/cdrom to run ./runInstaller! If you do so, the installation will fail.

If you forgot to set the DISPLAY environment variable (e.g. export DISPLAY=oracleserver:0.0), or if you forgot to give the remote console - your Oracle Server - authority to display X information on your desktop PC (e.g. xhost +oracleserver), you will get the following error:

```
Xlib: connection to ":0.0" refused by server
Xlib: Client is not authorized to connect to
Server
```

In this case, I always had to kill runInstaller which was still running in the background! If you don't do this, runInstaller will not completely come up any more and you will not see any error messages that runInstaller is having problems.

You might also want to clean up /tmp/OraInstall:

```
rm -rf /tmp/OraInstall
```

When runInstaller starts to configure the tools ("Configuration Tools"), the "Oracle Net Configuration Assistant" will sometimes hang. Simply stop the Assistant and restart it, or continue the installation. When the rest of the installation is finished, do a "Retry" for "Oracle Net Configuration Assistant". This always worked for me.

When the system stops to respond during the Oracle installation, in particular during the database creation, then it is probably because you don't have enough RAM or enough swap space. I noticed that the whole system will not respond (or "hang") for several minutes when I did not have enough swap space. If this happens, simply wait until the system starts to respond again.

The Oracle installation also runs make etc. In a production environment you might not have compilers and other development packages installed. Therefore make sure you have temporarily the following packages installed:

```
cpp, egcs, egcs-c++, glibc-devel, kernel-headers.
```

(I'm not sure though if all of these packages have to be on the system during the Oracle installation.)

If for any reason the Oracle installation didn't finish successfully, you might want to clean up the following files and directories before you restart over again:

```
rm -rf /etc/oraInst.loc /etc/oratab
/tmp/OraInstall

rm -rf $ORACLE_BASE/*
/tmp/<OtherOracleOwnedFiles>
```

Oracle Installation Errors

Here is a list of Oracle installation problems and solutions that have been posted by other people. Since I did not experience all of these problems, I am not able to verify the correctness of all the solutions. If you had other problems and you were able to resolve them, please send me an email at webmaster@puschitz.com so that I can add it to the list here.

First check always the error logs in /tmp/OraInstall. And when you get make problems check also $ORACLE_HOME/install/make.log .

```
"Error invoking target install of makefile
/opt/oracle/product/9.0.1/plsql/lib/ins_plsql.mk"

"Error invoking target install of makefile
/opt/oracle/product/9.0.1/precomp/lib/ins-
precomp.mk"

"Error invoking target install of makefile
/opt/oracle/product/9.0.1/precomp/lib/ins-net-
client"

"Error invoking target install of makefile
/opt/oracle/product/9.0.1/precomp/lib/ins-
oemagent"
```

Edit the file $ORACLE_HOME/bin/genclntsh and change the following line (people have sent me emails pointing out that this also works for Mandrake 8.1):

```
LD_SELF_CONTAINED="-z defs"
```

to read:

```
LD_SELF_CONTAINED=""
```

Then run this script $ORACLE_HOME/bin/genclntsh :

```
$ $ORACLE_HOME/bin/genclntsh
Created /opt/oracle/product/9.0.1/lib/libclntst9.a
$
```

Then hit retry in the error popup. This always worked for me.

```
"Error in setting permissions of file/directory
/opt/oracle/jre/1.1.8/bin/i686/native_threads/.ext
ract_args"
```

First of all, make sure you really installed the right version of Java (JDK 1.3.1 or Blackdown 1.1.8_v3) in /usr/local/java. If not, see Install JDK and try to install Oracle again. While the error dialog is open, manually find and copy the .extract_args file from your installed jre to where runInstaller complains it is missing.

```
"jre was not found in
/tmp/OraInstall/jre/bin/i586/green_threads/jre"
```

You are probably running runInstaller on a 586 machine, or your AMD CPU gets recognized as 586 (e.g. AMD K6-III-400).

You can check your machine (hardware) type by executing uname -m .

To rectify this problem, create a link for lib and bin from i586 to i686 and make the i686 directories read only:
E.g.

```
ln -s /tmp/OraInstall/jre/bin/i686
/tmp/OraInstall/jre/bin/i586
ln -s /tmp/OraInstall/jre/lib/i686
/tmp/OraInstall/jre/lib/i586
chmod u-w /tmp/OraInstall/jre/bin/i686
/tmp/OraInstall/jre/lib/i686
```

Now restart runInstaller.

Oracle Links

I also have some Oracle Linux links on my Home Page (http://www.puschitz.com/)

I tried to cover only Linux related Oracle topics. I did not go into configuring Oracle itself since there are enough web sites covering this topic. If you have any questions or comments, feel free to drop me an email at webmaster@puschitz.com

This article is re-printed with permission. The originals can be found at:

http://www.puschitz.com/InstallingOracle9i.html

AUUG Inc. Annual General Meeting Minutes

Location Carlton Crest Hotel, Sydney
Date 26 September 2001
Meeting started at 5:00 pm

Agenda

1. Apologies

David Purdue
Lucy Chubb

2. Approval of the minutes of the last Annual General Meeting.

Motion to accept: Catherine Allen. Seconded: Mark White. Carried.

3. Returning officer's report.

Peter Chubb represented the returning officer. Since there was no contest for the committee posts, there was no election and thus no report.

4. Approval of appointments to the Management Committee: Michael Paddon to the vacant office of Vice-President, Malcolm Caldwell, Peter Gray, Conrad Parker and Warren Toomey to vacant Ordinary Committee Member positions.

Motion to accept: Peter Chubb. Seconded: Lawrie Brown. Carried.

5. President's report

Michael Paddon (vice-president) represented the president.

We have had a year in which AUUG has done a lot of things right and a lot of things wrong.

What we have done right is to exercise fiscal responsibility. The Management Committee has managed to cut expenses to the minimum-- as the Treasurer will report, although on balance we spent more than we took in, on a cash flow basis more cash arrived at AUUG's door than left it. This can mainly be ascribed to the fact that we did not run a conference in this financial year.

We have also developed a model for running symposoia that deliver value to AUUG members as well as financial rewards to AUUG.

However we have continued to lose members, as the Secretary will report. We are at a stage where AUUG is not supportable on membership alone, and we rely on events to give us the income to survive. This means that the committee can not take risks on events or member services that may not turn out to be profitable, which in the end restricts what services can be offerred to members and hence the value for money of AUUG memberships. In essence, we have no room for experimentation.

And so we find ourselves having to focus on a strategy for the year to come.

Our primary goal is to increase membership. AUUG's long term survival de- pends on us having an income stream that can be relied upon and that matches our regular outgoings. This means having a membership roughly double what it is now. So every member needs to recruit one new member.

We think that the best way to increase membership is to provide value for money for the membership fee.

So our strategy for the coming year will be to provide better member ser-vices without increasing the cost of membership. There are two ways we will achieve this.

The first will be to look at the events we organise. As stated previous-ly we will be looking to arrange the events that our members require.

The other strategy we will persue is the deployment and development of electronic services--services that can be delivered to members over the Internet.

We believe that these services will help AUUG to attract and, more impor-tantly, retain members--as well as making an AUUG membership more valuable and seen to be more valuable.

Motion to accept: Frank Crawford. Seconded: Adrian Close. Carried.

6. Secretary's report.

I took over the office of secretary in July of this year, and I am still trying to find older documentation. I do not have a report from the out-going secretary, and he also did not supply a report to the last AGM, so this report will concentrate mainly on the last three months.

Current membership statistics

The oldest membership statistics available to me are from November 2000. At this time, we had 561 members on our books. On 31 July 2001 we only had 449 members.

These figures are alarming. We are seeing a serious decline in member-ship, and we must make it one of our highest priorities to reverse the change.

The main reason for the dramatic drop is that we are now taking non-re-newal of memberships more seriously. Previous reports included members whose memberships had expired in the previous 6 months. This is a one-time change, however: we can't close our eyes to the fact that we are experiencing a dramatic decline in membership. In less than a year we havehad a decline of 20% in memberships.

Looking at the details of the decline, there seems to be little difference across the country. Victoria shows the strongest decline with 27% fewer members, and the less populous states do relatively well. South Australia has increased memberships by 41%, but this should be viewed in perspective: the current membership statistics represent 16 members per million population, while Victoria has 19 and New South Wales has 22.

What are the causes of the decline? A number of reasons spring to mind:

- Decline of the chapters. It's clear that there is a strong correlation between chapter activity and healthy membership figures, but it's not clear which way round this works. If people are leaving at this rate, they won't be interested in chapter activities.
- General economic climate. This could be a possibility: the number of corporate members has dropped noticeably more than the number of indi- vidual members. There's little we can do about this one except hope.
- Lack of interest. Is AUUG becoming boring? Are we not supplying enough for our members?
- Competition. Are the Linux user groups, for example, taking our place?
- Change in focus. The Canberra chapter is relatively quiet, but a new group has sprung up and will have its first meeting in a couple of hours' time. I spoke to some of the organizers of the group, many of them AUUG members, and asked why they needed a new group. It seems that the focus is different, more towards commercial use of UNIX.

We should also note that the campaign to get more student members has been an unmitigated flop. The numbers have declined by 25% since November.

The management committee has been discussing this problem at each meet-ing, but we haven't found the silver bullet yet. We welcome suggestionsfrom the membership.

Table 1. Membership overview

November 2000	February 2001	May 2001	July 2001	Decline since November 2000
Individual Member				
368	376	344	299	19%
Corporate Member				
155	158	145	119	27%

November 2000	February 2001	May 2001	July 2001	Decline since November 2000
Student Member				
16	16	15	12	25%
Freebies				
17	17	16	15	
Subscription				
2	2	2	2	
Life Member				
2	2	2	2	
Corporate Sponsor				
1	1	1	1	
Total				

561	572	524	449	
	+2%	-8%	-14%	
NSW				
177	176	170	141	20%
VIC				
123	130	115	90	27%
ACT				
83	84	76	63	24%
QLD				
87	87	74	69	21%
WA				
36	35	26	29	19%
SA				
17	20	25	24	-41%
TAS				
15	17	13	14	7%
NT				
6	6	6	6	0%
OVERSEAS				
17	17	16	16	6%

Table 2. Membership renewals due

May 2001	July 2001			
December 2000	85	17%	(removed)	
June 2001	189	37%	97	20%
December 2001	217	43%	221	50%
June 2002	13	3%	110	24%
December 2002	0	2		
December 2005	1	1		
December 2008	1	1		
Perpetual	2	2		

Correspondence of note

We received a response from the Office of the Premier of South Australia in reply to the petition against the proposed Classification (Publications, Films and Computer Games) Miscellaneous Amendment Bill 2000, which was submitted by the members of the SA in April 2001. The letter is available in PDF and PostScript forms at http://www.auug.org.au/correspondence/.

We received a large number of replies to the Email about the inclusion of CD-ROMs in AUUGN. Without exception, these were positive. In addition, a good 10% of the membership replied. I think it's the biggest reply quota we've ever had. A summary was published in the July 2001 AUUGN.

(end of secretary's report)

There was a lively discussion of the reasons for the drop in membership. Peter Chubb observed that we are a generalist organisation, and that many people prefer specialist organisations.

Question from David Mandala (guest speaker): Why do people leave?

Reply: Many people make a choice of organisation because of cost reasons.

Question: What are our goals? What is our charter?

Reply: We exist to promote the use of UNIX and Open Systems.

Question: What will we do next?
Reply: The management committee has had many ideas, but a large number have failed due to lack of time. It was noted that many other UNIX groups, notably in Europe, have closed due to lack of interest, and that only USENIX was doing well.

Observation from the audience: USENIX is doing well mainly because of the SAGE membership.

Observation (unrecognised delegate): He had a lot of contact with Open Source and UNIX, and came to the conference specifically to hear about UNIX topics.

Reply: AUUG tries to cater for all types of UNIX users. The programme committee listens very much to the view of the membership, but it could do with more input.

Observation from Catherine Allen: In the early days of the AUUG, it wasn't easy to get support, and AUUG played the role of a support group. Now that has changed, and we need to redefine our role.

Observation from an unidentified delegate: AUUG is not very well known. We need to do more to make ourselves known.
Reply: The management committee needs assistance in this matter. People should identify groups and go out and talk about AUUG. Greg Lehey has presentation material if anybody wants it.

Observation by Sarah Kelly: She had known of SAGE for two years, but had never heard of AUUG.
Reply: We have obviously failed in our attempts to become better known in the universities. We need assistance from within.

Observation: The AUUG should take more active political standpoints.
Reply: Yes, we are planning to do more of that.

Observation from Mark White: After people find out about AUUG, the first thing they will do is to go to the web site, which is in dire need of improvement.

Reply: We have recognised this problem and have a subcommittee dedicated to fixing it.
Due to time constraints, further discussion was postponed. The management committee welcomes further input.

Motion to accept: David Newall. Seconded: Peter Chubb. Carried.

7. Treasurer's report

The Treasurer's report will be printed separately.

Question (David Newall): How do our finances look compared to two years ago?
Reply: We're no better or worse off than then.

Question: What about conference attendance?
Reply: It did not meet our expectations. This year we had 136 attendees, last year it was 150.

Question: Has the rate of decline of membership accelerated?
Reply: Not in the last four years. Before that, the membership figures were not in our control.

Motion to accept: Alan Cowie. Seconded: Mark White. Carried.

8. Other business.

Michael Paddon observed that a number of issues beyond our control have played a role in the decline in membership: UNIX is no longer a hot item, and the economic downturn has had an effect across the board. We should maintain a longer perspective: AUUG has see continual ups and downs in membership over the last 26 years.

Alan Cowie stressed the need to get people to plug the Security Symposium in Brisbane. Warren Toomey agreed to do so. It was observed that the symposia generate significant revenue for AUUG.

Peter Gray suggested to hold a BoF on the future of AUUG. Catherine Allen agreed to run a BoF on 27 September.

Andrew McRae asked how SAGE membership is doing. Reply (from audience): It is increasing at about 20% per year.

Andrew McRae observed that SAGE and ISOC were originally offshoots of AUUG, and asked if it is time to recombine. Frank Crawford replied: We tried to start discussions in the past, but we didn't get much in the way of a reply. It was noted that SAGE, ISOC and AUUG each target different groups.

Con Zymaris asked for ideas on content for AUUGN. Response: book reviews, tech tips. Frank Crawford's "home network" page was quoted as a good ex ample. Catherine Allen asked for more articles on Open Source. Con reminded that we need more article submissions from members.

Meeting closed, 6:05 pm.

Advertisement: American Bookstore

AUUG Chapter Meetings and Contact Details

CITY	LOCATION	OTHER
ADELAIDE	We meet at IBM in 180 Greenhill Road, Parkside, at 7 pm on the second Wednesday of each month.	Contact sa-exec@auug.org.au for further details.
BRISBANE	Inn on the Park 507 Coronation Drive Toowong	For further information, contact the QAUUG Executive Committee via email (qauug-exec@auug.org.au). The techno-logically deprived can contact Rick Stevenson on (07) 5578-8933. To subscribe to the QAUUG announcements mailing list, please send an e-mail message to: <majordomo@auug.org.au> containing the message "subscribe qauug <e-mail address>" in the e-mail body.
CANBERRA	Australian National University	
HOBART	University of Tasmania	
MELBOURNE	Various. For updated information See: http://www.vic.auug.org.au/ auugvic/av_meetings.html	The meetings alternate between Technical presentations in the odd numbered months and purely social occasions in the even numbered months. Some attempt is made to fit other AUUG activities into the schedule with minimum disruption.
PERTH	The Victoria League 276 Onslow Road Shenton Park	
SYDNEY	TBA	

FOR UP-TO-DATE DETAILS ON CHAPTERS AND MEETINGS, INCLUDING THOSE IN ALL OTHER AUSTRALIAN CITIES, PLEASE CHECK THE AUUG WEBSITE AT HTTP://WWW.AUUG.ORG.AU OR CALL THE AUUG OFFICE ON 1-800-625655.

Membership Application

FRONT

Membership Application

BACK

Volume 23 ● Number 1
March 2002

Features:

Implementing a Bridging Firewall 17
Paranoid II: The Revenge of Tinfoil Hat 19
Linux vs Windows: Security Alert Comparison 22
1001 Things to do with Liquid Nitrogen 23
The Great Giveaway 25
Interview: Jordan K. Hubbard of the FreeBSD Project 28
Symposium Report −3rd AUUG Security Symposium 34
Quantum Computing: Interview with Bruno Marchal 34
TeraServer: Build a large, cheap Linux file server 35
Why Gnutella Can’t Scale. No, Really. 39
The Gelato Federation Team at UNSW 45
Writing Gnome Applications with Glade and Python 46
Wind River: Transfer of FreeBSD Sponsorship 51
Quick Toots: CERES 52
Chrooting All Services in Linux 54
QCAD Technical Drawing 58
Linux.conf.au: The Photos 66
Hands−on Review: Sharp Zaurus 69
New UCITA Revisions 74
The Open Cluster Framework Project 78
Oracle 9i EE Installation on Red Hat Linux 80

News:

Public Notices 13
AUUG2002: Call for Papers 5
AUUG2002: Call for Nominations 11
AUUG2002: Nomination Form 12
AUUG Inc. AGM Meeting Minutes 84
AUUG: Chapter Meetings and Contact Details 89

Regulars:

President’s Column 3
/var/spool/mail/auugn 4
My Home Network 14