NetGuide, a consulting, educational, and Internet services company, has used Linux as its major operating system right from the start. The authors describe how the company, which has grown from four to 30 employees in only four years, choose the Linux system and how it has evolved.

Linux in the Workplace

Jacob Hallén, Consultant
Anders Hammarqvist, Chalmers University of Technology
Fredrik Juhlin, RJ Dataknosult
Anders Chrigström, NetGuide

FROM THE TRENCHES: Wolfgang B. Strigel, editor • wstrigel@spc.ca

In November 1994, VMS, Novell NetWare, and Unix were the only reasonable options for setting up large dial-in services. At the time, we were setting out to form the company NetGuide to provide dial-up services and third-party Internet support services to the public. We started with four partners/employees and inadequate funding, a fortunate accident that led us to using Linux.

Our expertise was in Unix, and we were not particularly fond of the two other systems. The remaining systems on the market, Windows NT, Windows 3.*, MacOS, and AmigaOS, were impossible for various reasons—Microsoft did not support a properly working TCP/IP stack for Windows at the time, the Macintosh has never been good at multitasking, and the Amiga system was already on its way out.

With Unix as the only option, we had to select the most useful dialect. AIX/RS6000, OSF 1/Alpha, and Irix/R3000 were ruled out quickly with funds being scarce. The only economically viable options were something running on a PC or something running on a lower-end SparcStation. Our main experiences were with the Sparc architecture, having administrated and used both SunOS 4 and 5 extensively. However, Sun Microsystems was withdrawing support for SunOS 4 and the problems with SunOS 5 (version 5.3) were very difficult.

0740-7459/99/$10.00 © 1999

Bill Gates may be right after all to be concerned that Microsoft could lose its market dominance overnight. As the following articles testify, Linux continues to gain recognition in academia as well as in the business world. Free software is an intriguing concept that is starting to gain momentum. About one year ago, Netscape announced its intention to make the source code for its Communicator client freely available. Since then, other companies, including Corel, Oracle, and Informix, have released versions of their software that run on Linux, a free operating system.

As the paper by Bokhari et al. shows, it is very appealing to universities, especially those in developing countries, to use free software. It is quite another thing to entrust mission critical software applications to freeware. Our second paper by Hallen et al. shows that the pain to do so is not much worse that the pain you expect from using 'bleeding edge" technology. I vividly remember the trials and tribulations of using object-oriented technology in the early days of C++. Although the compilers and support software had to be bought for real money on the commercial market, it seems we actually fought more bugs and support problems than Hallen and Bokhari who use today's freeware.

—Wolfgang Strigel, From the Trenches Editor

This left us with the Intel platform and a choice between Linux, FreeBSD, NetBSD, and a few commercial Unix releases. We decided to evaluate the free software products since we were a poor startup company and since we wanted to have the source code available. It quickly turned out that there was some turbulence in the Berkeley Software Distribution groups and that releases of the BSD-based packages were hard to find. Linux, on the other hand, was available in several neat releases from a site we could easily connect to (ftp.funet.fi). One of the partners had started teaching a programming course on the Slackware distribution a few months prior and another one had installed the same release on his PC.

In both cases, we were impressed with the ease of installation, the available documentation, the completeness of the distribution, and the stability of the kernel. We especially found the HOWTO documents, which were unique to Linux, a useful feature.

Since Linux was a fairly young operating system, we considered selecting it a risk. However, it was a calculated risk and, compared to the other risks you take when starting a new enterprise, not a very significant one. Some people may have considered the lack of vendor support and backup services important factors. We considered these negligible since there was a Swedish company selling support at the time. The availability of several Linux distributions and the fact that we had the source code and the know-how to maintain the systems ourselves kept us comfortable as well.

Why We Still Run Linux

NetGuide has grown into a thriving business with approximately 30 employees. We can easily afford the licensing costs for any mainstream operating system, but we choose to stick with Linux. We left the Slackware distributions in late Spring 1995 due to the pains we had with upgrades. The Debian distribution has supplied us with all the functionality we need. After changing to Debian, we haven't done any serious evaluation of any of the other releases.

We have had some trouble with Linux, as described in the Difficulties and Solutions section, but have reason to believe that no other operating system would have performed better. There will always be problems no matter what system you select.

When we have had problems, the Linux community has, in every instance, provided help analyzing and fixing reported problems, generally within one to three days. The turnaround time has been better than what we have come to expect from commercial software houses.

The main reason for continuing to use Linux is that it keeps our servers for vital services like the WWW, mail, domain name server, network file server, remote Windows file systems, printing, and faxing, running with little maintenance and virtually no downtime due to bugs or service time. You can do the same with a commercial Unix system and you might be able to do it with Windows NT or one of the free BSD variants, but the most cost-effective way appears to be Linux.

The System Setup

When we originally set up the system for our current location, it consisted of two Linux servers and a group of workstations designed to run Windows 95 as well as Linux. The only task of one server was to act as a terminal server for the internet service provider customers, which meant hooking it up with a lot of modems. The other server had a lot more to

Web Sites and References

The Debian Project: http://www.debian.org/

Samba: http://samba.anu.edu.au/samba/

Hylafax: http://www.hylafax.org/

Windows Hylafax Client: http://www.transcom.de/whfc/

LPRng: http://www.astart.com/LPRng.html

Sendmail: http://www.sendmail.org/

The Cathedral and the Bazaar (1997), by Eric Raymond: http://www.tuxedo.org/~esr/writings/cathedral-bazaar/

RFC (Request for Comment) is a series of standards defining the Internet. The most authoritative source is http://www.ietf.org/.

Radius (Remote Authentication Dial In User Service): RFC 2138.

NTP (Network Time Protocol): RFC 1305.

NNTP (Network News Transfer Protocol): RFC 977.

do, since it acted as a mail, Web, and news server for the office and for the ISP customers. It also provided Network Information System, Samba, Atalkd, and interactive logins for the office. While this might seem a heavy workload, it actually performed rather well.

Over time, the usage of Internet resources, such as news and mail, grew along with the company. Consequently, we decided to split the work on several servers. Today, the company has six servers running Linux, dividing the tasks as follows:

- Endevour: news server, handling 45 Gbytes of incoming news flow per day
- Vega: NIS server, Web server and Radius server
- Valdez: application server, providing interactive login, Samba, mail, and Web services
- Ejdern: DNS server, handling name services for approximately 300 domains
- Nautilus: mail server, handling mail for the ISP customers and forwarding staff mail to Valdez
- Frontpage-server: server configured with Apache and Frontpage to help customers develop their own Web sites

In addition to these tasks, some servers also act as network time protocol servers and secondary NIS and DNS servers. We also have two Windows NT servers. One acts as a Lotus Domino server and runs our WWW-hotel service, which hosts external Web sites providing network access, adequate bandwidth, security, and administrative services. The other is currently not in use, but will host our intranet.

Three Architectures

The Intel architecture is not the only one that supports Linux. By chance we had Sun SparcStation and Digital Alpha machines at our disposal, and our faith in Linux prompted us to install it on these machines as well. The SparcStation ended up as our DNS server and the Alpha became our main Linux application server.

Getting Linux to function well on the SparcStation proved to be a challenge. We wanted to use Debian, since we were using it on our Intel machines. However, the Sparc distribution of Debian was not a smooth installation at the time, since it was not very far along when we started looking at it. The porting effort had recently been plagued by a broken compiler, and there were bugs in the C library that proved fatal to many programs. After helping to fix these bugs, we compiled the packages containing the utilities we needed. The Sparc distribution has progressed quite far since then, although it is not yet as stable as the Intel version.

After the initial setup, the SparcStation has run very smoothly. It is the master DNS server for all the domains we manage and doubles as an NTP server for our internal machines.

The Alpha installation was a dream in comparison to the Sparc installation. This time we already knew what we were getting in to, as one of us had bought himself a Alpha to play with a few months prior. Again we went with Debian, and, though officially unstable, it turned out as solid as the Intel distribution. Debian for the Alpha was much further along than the Sparc port when we starting looking at it. There were no major bugs and most Alpha specific trouble stemmed from developers not anticipating a 64-bit architecture. There were a few programs not yet available for the Alpha, but we were able to compile all of the missing Debian packages. The major complaint with the switch turned out to be that Netscape was not available for AlphaLinux.

Some Special Uses of Linux

Though some of us wish it otherwise, we needed to support non-Linux platforms, primarily Microsoft Windows. These machines, which were used primarily for office automation, needed reliable centralized file storage and printing facilities. We found Samba able to provide both. Using Linux for file storage proved useful, as the files can then be backed up with the rest of the system, rather than requiring a separate backup system for MSWindows.

Running Samba requires very little extra administration. The server was essentially set up and forgotten, as it takes all its dynamic data, such as users, from the Linux facilities. Since it uses normal

MSWindows networking, the client setup is also quite simple, requiring only that we set up TCP/IP networking and instruct the user how to mount a Samba file system. To simplify the client setup further, we are looking into using Samba to provide the user profiles, which would enable us to set up and standardize the mounts for users.

Printing tends to be a problem whenever you use computers, and printing in a network environment causes even more problems. The Lexmark networked laser printer that was the sole printing device of the company up to a few months ago was supposed to be able to cope with printing jobs from different types of sources. In practice, it turned out that two flavors of Unix, two flavors of MSWindows, and two versions of MacOS left the printer confused in a number of different ways, depending on the order in which they tried to print. We found that using LPRng, one of the later versions of the line printer daemon, on Linux to talk to the printer allowed reliable printing via the network. Now we had Linux printing reliably, and again Samba rescued the MSWindows clients, allowing them to print via the Linux queuing system. Atalkd has done the same for the Macintoshes. Printing works smoothly from all environments and there is no need for operator intervention. An additional advantage is that when setting up printing on a new platform, you can monitor exactly what happens on the Linux print server. This was not possible when sending print jobs directly to the printer.

Another useful Linux program is Hylafax, which is possibly the most capable computer fax solution available. Inquiries with system administrators at the Chalmers University of Technology led us to select it over other fax servers. Our current usage is limited to reception and printing of faxes, except for the automatic distribution of faxes from known senders. The fax system has been running for two years with virtually no attendance. Recent improvements to the MSWindows client software for Hylafax prompted us to look into the possibility of faxing from our MSWindows machines.

Linux for an ISP

Of all the things we use Linux for, the one hardest to replace would be Sendmail. As an ISP we must be able to handle the e-mail requirements of our customers, which is not as simple as it may seem. We handle e-mail for over 250 domains. To get the mail to our customers we use personal post office protocol (POP) and Internet message access protocol (IMAP) mailboxes, in addition to Unix-to-Unix copy protocol (UUCP); simple mail transfer protocol (SMTP) on demand, which is used to queue mail and send it to a server when the customer connects; and multidrop POP, where the messages will be distributed to the individual recipients by the client's mail software. Customers often want a combination of services where some addresses end up in a personal mailbox on our server while others are forwarded to their mail server. Sometimes they even want an address to end up both in a personal mailbox and have it forwarded. Thanks to Sendmail, we are able to do this on a single machine with a single configuration.

As part of being an ISP, we use a modem server. At the time the company was started, standalone modems were the only reasonable choice. Experience with standalone terminal servers, mainly the Annex line, prompted us to look elsewhere for a reliable solution for providing serial ports to the modems. We ended up choosing Rocketport multiserial boards for use with Linux, based on recommendations and the availability of a Linux driver for them. We had no information about the quality of the driver when choosing the board, but trusted the availability of source code to make sure we could use it. After some customization of the login facilities on the machine, the system worked reasonably well for the three years it was in use. An increasing number of modems and the need to support a pool of ISDN lines prompted us to upgrade to Livingston Portmasters. A Linux machine still acts as Radius server for the Portmasters. This solution is extremely stable and has cut the amount of time spent on the dial-in lines to almost nothing.

Not every aspect of using Linux has been easy.

Difficulties and Solutions

Certainly not every aspect of using Linux has been easy. Some problems required more work than others to overcome, and for completeness of the story some of them should be included.

Many of our initial problems were hardware compatibility issues. Perhaps most notable was the Adaptec 2940 SCSI card we equipped our all-round server with. We were about to move our offices from a tiny closet to a new and spacious location, and getting the new server to work was essential to make

the move. We solved the problem by making big alterations to the SCSI driver in some long debugging sessions. The source of the problem was that we were eager to use the latest technology to get the most out of our machines and were on the bleeding edge of Linux kernel development. We finally realized that it is not prudent for a commercial setup to use software that is at the forefront of development for production, so today the problem with hardware compatibility is almost a fading memory.

We also had some difficulties with Linux point-to-point protocol support. Initially, PPP worked reasonably well, but lacked some essential features. This version was also more of a quick hack than a stable software packet. A rewrite of the PPP server in late 1995 added much of the needed functionality, but contained a number of bugs. In addition, we lost the support for routing a subnet over the PPP interface in the rewrite. After some investigation and discussion, we decided to use the source code to track down, fix, and report bugs rather than wait for someone else to do it. The reason for this was not that we expected a long wait for the patch, but rather that it felt satisfying to contribute to the development of Linux, even in this small way.

You will need access to advanced support services, from either your own personnel or a third party.

The resemblance of Linux to Windows 95 was striking in a small but rather irritating problem. The clients would frequently lose contact with the server. The problem was easily fixed by restarting the NIS software on the client. For those members of the staff who did not have access to the root password, this meant having to reboot the workstation. An upgrade to the NIS software fixed this problem.

We can't use Linux for every computer-based task. Office programs for Linux, such as Applixware and StarOffice, are now entering the market. However, we continue to use Windows 95 on our workstations, because our staff is familiar with Microsoft products and the Windows environment. Since the printer and file server run smoothly with support from Linux, there is low incentive to change. We haven't been able to keep a pure Linux profile on the server side either. As mentioned earlier, we have a Windows NT server with Lotus Domino, running Sweden's first Lotus Notes WWW-hotel.

Running Linux

The preconditions for running Linux in an enterprise can be summarized in one word: expertise.

One of the most important factors in using computers for business purposes is reliability. The machines must work when your organization and your customers expect them to. There must be no surprises when there is a software upgrade or when you are undertaking a new project. There must be no undue delays because the developer of a software package decides to quit. You must be fairly sure that there are no traps or unexpected side effects in the installed modules.

The person responsible for machine installation and support must have a good grasp of the system. We have in-house expertise, but it is reasonable to buy support services from a third party. We know of at least three companies in Sweden that advertise such services.

NetGuide's support staff has consisted of one person with fairly limited knowledge of Unix system administration. His job has been to take care of the day-to-day running of the systems and to deal with tasks within his area of competence. For advanced and custom tasks beyond his skills, he has had access to several seasoned systems administrators and Unix experts. They have also jumped in and investigated major problems and helped during crises. This setup has been highly successful. It allows the systems administrator to gain experience and build skills, while permitting us to have our most highly skilled staff members spend their time on more profitable work.

Consequently, the skills needed for day-to-day systems administration are fairly low and dropping as the various Linux distributions become more user-friendly and easy to install. They are still a bit higher than those needed for the Microsoft packages in a standard environment, but the difference is minimal. The skill demands for advanced support services are at the same level as for systems and applications experts at the major software houses. You need access to such services, either from your own personnel or from a trusted third party.

NetGuide has run a successful enterprise using Linux as its major operating system for four years and has no current plans to change the company strategy. Both of our non-Intel machines running Linux have proven solid.

The high quality of software available for Linux is mainly due to the nature of software development in the Linux community. Some major software packages are not supported on the Linux platform, with equivalent replacements not readily available. This situation is rapidly changing, and just about everything will be available within a year, either as a port by the product vendor or as a comparable replacement from another source.

As with all operating systems, it is necessary to ensure proper in-house and external support and to become familiar with a platform before putting it into production. The stability, support level, performance, and reliability of Linux are equal to, or better than, its commercial counterparts. The important factor is dependability. ❖

About the Authors

Jacob Hallén is an independent consultant, spending most of his time assisting the Royal Library in modernizing national computer-based library resources. He is also the former CEO of NetGuide. Prior to these positions, he has taught computer engineering and computer science courses at Chalmers University of Technology and has built systems for the control of irrigation and animal fodder production.

Anders Hammarqvist, a former consultant at NetGuide, is currently pursuing a MS in Engineering and Physics at Chalmers University of Technology. He has also worked as a Unix systems administrator for the Department of Physics at Chalmers.

Fredrik Juhlin is a consultant at his own company, RJ Dataknosult, and is a former systems administrator and consultant at NetGuide.

Anders Chrigström is a consultant at NetGuide. He has a background as a Unix system administrator at the Physics Department of Chalmers University of Technology.

The authors may be contacted through Hallén at Gibraltargatan 90-753, 412 79 Göteborg, Sweden; e-mail jacob@cd.chalmers.se.

Call for Participation: Standard Dictionary of Measures of the Software Aspects of Dependability

The IEEE Software Engineering Standards Committee has a project underway to develop a Standard Dictionary of Measures of the Software Aspects of Dependability, chaired by J. Dennis Lawrence of the Lawrence Livermore National Laboratory. The scope of this project is to specify and classify measures of the software aspects of dependability and to provide conventions for the application and use of these measures. The standard will include the following aspects of dependability: reliability, availability, and maintainability of software. An important purpose of this project is to determine criteria for including measures in the dictionary. Once we have identified appropriate criteria, there will be a sound basis for deciding whether to include proposed measures in the standard.

The Working Group desires to develop a consensus within the software engineering community regarding appropriate criteria. Therefore, we invite your recommendations on the criteria that should be employed. The criteria should be independent of specific measures (that is, they should apply across the board to all measures). For example, suppose one person claims that failure rate is a good measure of software reliability and another person claims that mean time to failure (MTTF) is a good measure. What criteria could we apply to both of these measures to determine whether they should be included in the standard (for example, have the application of these measures helped software developers and users achieve their reliability goals?)?

Your recommendations are important to us. Please send them by email to me at nschneid@nps.navy.mil. Please provide your name, affiliation, and email address in your message. In addition, if you are not a member of this Working Group, please indicate whether you would like to become a member. Thank you in advance for assisting in this important project.

—Norman F. Schneidewind