

Internet migration to Linux

100%
LINUX

In 1999, Mailbox Internet was the first UK Internet Service Provider to successfully migrate 100% to a Linux platform.

Joel Rowbottom tells the story...

Although relatively unknown at the time, Mailbox Internet was one of the first independent Internet providers in the UK. The company was established in 1992 to provide Internet services to public relations agencies as an offshoot of parent company Mailbox, which supplies services to the PR and fulfilment industries. To this day the company is still a wholly owned network, with no outsourcing of software or hardware requirements so common to the virtual Internet providers of today.

The story starts in 1992. In the beginning, we were primarily using Solaris machines (a batch of Sun SparcStation 20 servers provided the core services such as web, mail and DNS), and a sprinkling of Apple kit. The reasons for this were that the hardware was readily available at the time and the equipment was familiar to the staff contracted to set up the ISP from the outset.

In the fullness of time however, the scalability and security of such machines and the cost of running them began to cause problems: a replacement

part for a SparcStation at that point was quite expensive, not to mention the cost of licences for the only operating system available at the time, Solaris. The in-house administration system exported details every so often to a Perl script called "Sys-Setup", whose task it was to compile the relevant bits of DNS zone file, email alias file, authentication etc. and throw it out to the relevant servers. It was referred to quite frequently in the office as "that bit of voodoo which makes things work!"

I must admit I feel somewhat responsible for Mailbox Internet's migration to Linux. In 1998 I joined the company and found a network which was creaking at the seams. Personally I'd been using Linux since around 1993: a fan of the Slackware distribution introduced to me by a friend. Additionally I'd replaced the core systems at three separate companies with Linux servers by the time I got to Mailbox, and it proved to be quite a speedy way of fixing problems for good.

A question which was asked of me by the company Chairman several times throughout the re-implementation was "why Linux?": why indeed? Well, mostly it was the reliability of the operating system itself. We'd been using Solaris on most of the servers and the frequent need for patches has been well-documented - it was a mix of versions 2.5.1 and 2.6 with OpenWindows and CDE, which didn't seem all that efficient on the limited hardware we had available.

A second issue was cost: for the price of a second-hand SparcStation 20 we could have purchased a brand new multi-processor x86 server, without any software licencing charges or ongoing support costs (99% of problems we could fix ourselves in any case).

A final issue - and certainly one which bore thinking about - was security. There are all sorts of horror stories floating around the Net about most operating systems (including Linux) but it all comes down to how you configure your server. Indeed, a properly configured Solaris can be locked down, but it takes all sorts of security patches and non-stan-

Technician Shish Batal installs a new server at Telehouse-1



ard tweaks. Linux can be tightened up quickly and easily with the aid of “ipchains” and taking a long hard look at what you really need to run on your machine (do you really need to run “inetd” if you’re running a mail server?).

Keeping the Net alive

The main issue of course was that we had to do all the replacement work without impacting the thousands of customers whom we already serviced, so it was going to be a lot of testing and a lot of late nights! Next we put together a schedule of what we need to provide to our users.

First on the list was the news server, which for a typical full newfeed of the time (May 1998) would eat up to 10Gb of disk space a day. We’d recently found a manufacturer who was willing to part with a stack of 18Gb SCSI disks and installed these into the dual Intel PII-400 motherboard. SMP was still in its infancy so we only put one CPU on the board to start with and added the other as soon as kernel 2.2 was released.

A brief installation of our slimmed-down version of Linux, use software RAID to “glue” the partitions together into one big mount, install INN2.2 and we were ready to go. The machine performed admirably (first time!) and there was much rejoicing: we’d put the first Linux box onto our network and the users had a reliable Usenet feed. No problem!

After that, it was quite easy: we replaced the DNS servers very quickly utilising a copy of ISC’s “bind” package, and our dialup authentication system was provided by a copy of Cistron RADIUS. Intel Celeron boxes proved to be great servers. It’s worth pointing out by the way that where space is a consideration there are some fantastic 1U-high cases out there if you look around, which will usually work quite happily with standard all-onboard motherboards.

Mail system

Email proved to be a problem: we knew we’d need failsafe machines, but there was the problem of legacy UUCP and all sorts of problems rebuilding sendmail configuration files from SysSetup. Sendmail at that point didn’t have any of the advanced spam filters, ETRN or monitoring hooks which we needed, and after a series of discussions with peers at other ISPs we decided to plump for Phillip Hazel’s “exim” package instead.

Exim is a drop-in replacement for sendmail that doesn’t have a configuration file which looks like an explosion in a punctuation factory. It has got its drawbacks, but it’s reliable and simple to fault-find in the event of a glitch. We had a steep learning curve to climb but there’s an experienced Exim support network which proved very useful.

The email system itself finally consisted of four machines, with a clustering method designed to make addition of extra machines simple: one



machine would run a POP3 daemon and two machines would act as redirections for the virtual hosts. The POP3 daemon machine would access a filestore via a networked filesystem. We patched a version of CuciPOP to allow us to authenticate from an alternative password file in order to create virtual POP3 users and enable locking for a networked filesystem rather than the scoreboard-style approach which it already used.

Core Web server

The system flew, first time as well – it’s very important that it’s controlled quite closely when you migrate an email system, as one single bounced email is enough to let the cat out of the bag and make your users annoyed.

The final chief service to migrate to Linux was the core web server. This had the potential to be the iceberg under the Titanic: the previous system administrator allowed incoming telnet and FTP access to the entire filesystem so the possibility existed of hidden architecture-dependent code within customer user areas. In any event, we decided to stop telnet access since almost all users were accessing the system via FTP.

The solution we eventually went for involved moving blocks of sites to one of three separate servers and running a mount to a network filesystem. Users were locked into their own directories thanks to the security features in the ProFTPD daemon and a copy of Apache appropriately configured gave us the capability for serving webpages.

Admittedly, that one didn’t work completely first time round. We had a lot of older customers coming out of the woodwork within a few weeks: it turns out they’d been accessing through absolute paths, or had been uploading compiled Solaris programs! A brief chat about security solved that one.

Exhibiting at
Linux Expo 2000

After that final server we'd completed the process of migrating all core services from Solaris. It wasn't that difficult. Certainly the biggest pitfalls encountered were converting the Solaris-standard paths to the Linux-standard paths and making sure that custom-written software was recompiled. Because of the SysSetup script (now rewritten into our in-house Intranet admin system using mod_perl and Apache) it's a quick job to rebuild the entire ISP from scratch: nobody can manage that much subscription data manually!

Administration Systems

I mentioned earlier the "SysSetup" script that we had written. In the early days the administration machine was an old Altos running via a UUCP link to a Sun SparcStation clone which fed the text-based report to a Perl script! This might sound all very convoluted but it was certainly an imaginative way of setting things up automatically. The time came though when the Altos was ready to go and we had to replace it.

There were two alternatives: either we found a shrink-wrap system or we built one. There were no shrink-wrap systems for Linux which were documented or complete at the time so it was up to us to write one.

We tried quite a few approaches to creating the user interfaces: an ncurses interface, using GTK widgets, Java applications, and so forth. The final approach we chose was the obvious one - a web-based system with client-side verification using Javascript. Server-side operations were written in Perl, powered in the first instance via a MySQL database. Apache with SSL meant that we could write several different interfaces for customers and resellers while still retaining a "master console" system for ourselves.

Nowadays it all links together with the support ticket database, billing system, reporting system, and event logging. It even talks to our Panasonic DBS phone system, automatically logging when a customer calls, who they were passed to and how long they were on the call.

More recently, we've been taking a look at running SuSE on the desktops of all the adminis-

trative staff at Mailbox. It saves us a lot of hassle on maintenance since one of the technical staff can always telnet in and fix it, kill an errant process, or whatever it takes. Additionally it's giving us that extra bit of security which you can't lock down if you're using Windows, although invariably we get the "where's Microsoft Office?" type questions! The receptionists have been happy with it for the past eight months and we've started running it for the marketing staff. The techies, of course, have been running Linux for years...

During 1999 we launched our £35 per month colocation deal and a sudden boom meant that we had to expand our network. We also ran into a problem with logging and network monitoring. Of course, when a server or a router dies you need to get straight on to fixing it (even if it's the middle of the night) so we had to sort out some sort of monitoring software.

On that note I can thoroughly recommend "NetSaint" as a monitoring application: combined with a copy of "sms_client" it has the ability to page you out of hours if things go wrong. Net-Saint will try to relay an email, carry out a DNS query, ping a router or check a webpage. If anything is amiss it'll try again, only alerting you when something is really broken. We liked Net-Saint so much in the end that we began using it to monitor our entire network: while it can be hard getting out of bed, it's better than having a reputation for being a broken ISP!

Bandwidth

The other issue with providing colo is that of bandwidth. We use managed switches at Mailbox which have SNMP capability: handy for querying with MRTG - the Multi-Router Traffic Grapher. We also occasionally use the network monitor Etheral to pinpoint troublesome servers though we've only ever had to do that twice.

Out of bizarre mix of servers which we collocated at our Fulham facility, I can safely say that a substantial portion of them run Linux: there's an Apple Network Server (fondly referred to as "Erik's Fridge"), an Acorn Archimedes, a whole raft of x86 machines, various ex-University Sparc boxes; the list goes on. It's good for those customers to be able to email us and say "I don't suppose you know how to tweak Exim to do this, do you...?"

The Mailbox network is one to be proud of: it's expandable, reliable, and received a lot of favourable comments at the recent Linux Expo show at Olympia. All our core servers are sited at Telehouse-1, a lights-out facility in Docklands: we rarely need to visit it to do upgrades. Now we're working in preparation for the roll-out of business and domestic ADSL, upgrading the core routing kit to deal with the demand of our colocation services. It rarely stops round here! ■

Info

Mailbox Internet:
<http://www.mailbox.net.uk>
 Joel: joel@mailbox.net.uk



The Fulham low-cost colocation facility: a bazaar of server hardware

THE AUTHOR
 JOEL ROWBOTTOM BSC (HONS)
 IS MANAGING DIRECTOR OF
 MAILBOX INTERNET LTD, UK

Linux on a Sparc

So what happened to the SparcStations? We tried originally to build our own distribution: it was a struggle and eventually we gave up. A year later support for Linux on Sparc was a lot more visible and after a brief brush with the Red Hat Sparc distribution we settled on Debian. The servers are still quite happily ticking over as backup machines for mail and DNS.