LinuxFocus article number 283
http://linuxfocus.org

Interview with Jeff Dike

by Floris Lambrechts
<floris-(at)-linuxfocus-(dot)-org>

About the author:

In a couple of months I will officially be an industrial engineer in electronics/ict. I like free software, open standards and nice user interfaces.

Abstract:

User Mode Linux, UML in short, is one of the many interesting features that the Latest Linux kernel (2.6.x) brings to your desktop, server or mobile device. This interview is an edited transcript of an email conversation I had with the creator of UML, Jeff Dike, in January 2003.

At the end, David Coulson steps in to say a few words about tools to manage UML systems.

Introduction

As always, the Linux development kernel being worked on by hackers around the world (note: at the time of writing this was kernel 2.5) contains a whole load of new features. While UML is only one of them, it's quite a special one - it has the possibility to change the way you administer your Linux systems.
Because we feel it deserves some more attention than it has received, we highlight UML in a special introductory interview with no one less than the project leader himself, Jeff Dike.

Jeff himself

LF: *So, you are Jeff Dike, 'The guy in charge' of the UML project. Can you give a short introduction of yourself?*

JD: Yes, you have found me...

OK, here is a short blurb from Bill Stearns, who is my official biographer:

> Jeff finished his tour of service to the State of New Hampshire when his parole was granted in 1992. Since then he has served as a freedom fighter in Idaho for the Potato Liberation Front, an extra in "Death to Smoochy" (dead body #7), and as lead starcher in Deering NH’s largest Chinese laundry. He is currently rewriting the linux kernel as a DOS batch file.

LF: *Are you working on UML (semi)professionally, or is it just a hobby?*

JD: It’s a hobby more or less. I have received a good amount of support from Dartmouth ISTS (www.ists.dartmouth.edu) for honeypot-related UML work. I’ve also had occasional support from companies which wanted features added or bugs fixed.

LF: *First, let’s get the basics right. After reading the website user-mode-linux.sourceforge.net, my basic understainding of the concept is that UML is a Linux kernel that runs on another ’hosting’ kernel. Therfore, the UML kernel lives in userspace, and provides ’virtual’ hardware to whatever that is running under it. Did I ’get’ it right?*

JD: Yes. UML treats a Linux system as a platform that Linux may possibly be ported to, exactly analogous to looking at a bare x86 or ppc machine and considering porting Linux to it. Of course, Linux as a platform to run an OS on is very different from a hardware platform, and one of the things that UML proved by its existence is that the Linux system call interface is rich enough that an OS can be implemented in terms of it.

UML’s hardware is virtual, in the sense that it accesses all outside resources through the host kernel. Normally this means that the devices seen by a UML instance are really virtual, i.e. a "disk" which is really a file on the host. However, you can assign a physical disk, partition, CD-ROM, or floppy to a UML block device by way of the device file, and the UML will be able to access the data on it. However, it will not be able to issue low-level commands to the device in the same way that the driver in the host kernel does.

However, there has been some work in this area, with a patch recently posted which claims (I haven’t looked at it yet) to allow stock PCI device drivers to be built into UML and drive the appropriate physical devices.

LF: *So it *does* run in user space... Can you run it as non-root? Can you even chroot it?*

JD: Yes, and yes. In fact, I highly discourage running it as root, not because there’s any great danger to it, but because it is totally pointless. Some users have some problem getting used to the idea that they can run UML as a normal user and then become root inside the UML with all the privileges that come with it. They think they need to run UML as root, so [I am] very emphatically telling them that’s not necessary, and having them find out that it’s true is something of a learning experience for them. I’m also gratified that this attitude has taken hold in the UML user community - I hardly ever have to do this re-education any more; other users now do it for me.

It can be chrooted, and that is recommended procedure for people who are using UML to contain possibly hostile software or users, as an extra layer of protection to the host in the event that someone finds a way of breaking out.

LF: *There is a 2.4 and the newer 2.5 series of UML. Can you run the 2.5 UML on a 2.4 host kernel, and* *vice versa?*

JD: There are basically no dependencies between the host kernel version and the UML kernel version. As long as the host can run UML at all, it can run any version of UML.

LF: *Cool... Is it dependant on the x86 platforms then? Does it run on, say, MIPS or PPC?*

JD: No, it doesn’t. There was a ppc port that was nearly fully functional at one point, but it stopped being maintained, and has bitrotted. Ports are certainly possible, and shouldn’t be that hard. I’d like to see some, especially if they will be maintained.

LF: *So until someone picks up that work, it is actually a part of the kernel that is not yet ported?*

JD: No. UML **is** a port. You can’t say that it hasn’t been ported. Porting UML to a new architecture is akin to porting a hardware architecture to a new chip in the same architecture.

LF: *I get the picture. So what about ’real’ ports: how much work would one have to do to make UML run on Windows?*

JD: This has already been (mostly) done twice. The most recent effort apparently required about a week of hacking. The older one (http://umlwin32.sf.net) has been ongoing in peoples’ spare time over a period of months, so it’s hard to say how much continuous effort it represents.

And I would just like to say that it would be nice for one of these ports to be finished and debugged. There is a great deal of demand and interest in UML running on Windows.

LF: *How did you feel about UML being included in the official Linux 2.5 tree?*

JD: Technically, it’s not a big deal. I had been maintaining UML out of the tree for more than three years (and I am continuing to maintain the 2.4 UML out of tree), and it would not have been a problem to continue doing so.

The recognition that it brought was nice. When Linus merged UML into 2.5, the world saw that UML was worthwhile (and a good part of the world became aware of UML for the first time). This "stamp of approval" is probably the most significant aspect of having UML in the Linus tree, as far as I’m concerned. It now can’t be dismissed as a loony kernel patch on the fringes of respectable kernel development.

LF: *UML is a new concept for sysadmins, an unexplored area. Are the UML management tools (like the UMLd) alrady mature - are they effective at handling the new challenges UML creates for sysadmins?*

JD: This would be a question to ask Bill Stearns or David Coulson. I try to deal strictly with the kernel and leave management tools to other people.

LF: *David? What about UMLd?*

DC: UMLd is a daemon which manages UMLs for both users and administrators. For users, it allows them to reboot a crashed UML, or update the kernel, without having shell access to the machine. For administrators, it allows UMLs to be controled and modified, even created, without having to mess around with files on the server.

Ideally, UMLd would be able to control every aspect of the UML, which on the most part it does. However, as UML is constantly being developed, UMLd has to be kept up to date with anything which is added to the UML kernel.

(...)
I’ve been running UMLd on a two host, 30+ UML network for a number of months, and I’ve had very few problems with it. I use it for building all of my UMLs, rebooting them, upgrading the kernels, and so forth. To be useful for end-users, UMLd really needs a nice GUI front-end writing for it, which is something which I will likely start writing once I’m happy with the features of UMLd and the protocol which UMLd talks with the clients has been stabalized.

LF: You guys seem to have a lot of fun. One example is the , "The Sysadmin Disaster of the Month" on UML boxes contest, another one is this quote: "UML for fun and profit / Well, forget about the profit part" (http://uml.openconsultancy.com/). Is UML something between a toy and a tool, or are you guys just of the humurous type ;-) ?

JD: I think it is the latter. UML has brought together a bunch of people who happen to have somewhat whacked senses of humor, so, of course, we get along fine :-)

It may also be that this virtualized world of universes inside universes attracts people with a certain mentality, and that could explain some of the humor floating around.

No matter how serious what we’re doing is, or how serious the implications and uses of it are, you gotta have fun. It would be a very dreary world if you’re not allowed to let what you’re doing amuse you somehow.

LF: *Sixteen months after the first questions, do you have any additional remarks?*

JD: I would also mention that ther is a working x86_64 port of UML which I hope to release soon.

Also, I have done some more writing about what’s new in UML, and what’s coming up in my diary.

Conclusion

User Mode Linux is a nice addition the Linux kernel that is ideal for testing and demoing applications, kernels and distributions. It can also be used to make ’virtual hosting’ more flexible by giving people their ’own’ virtual Linux image with root access without endangering the host system.

References

- The complete email conversation for this interview (with some extra info)
- Jeff Dike’s UML diary to see what’s new and what’s coming
- user-mode-linux.sourceforge.net : homepage of the UML project
- usermodelinux.org : a community site for and by UML users, with lots of other useful links

Webpages maintained by the LinuxFocus Editor team

Translation information:
en --> -- : Floris Lambrechts <floris-(at)-linuxfocus-(dot)-org>

2005-01-14, generated by lfparser_pdf version 2.51