

# ZACK'S KERNEL NEWS

## GUIDE TO WHITESPACE IN KERNEL SOURCE

Michael S. Tsirkin, as part of indoctrinating some coworkers into the arcana of Linux kernel development practices, found himself struggling to explain to them the intricacies of whitespace utilization in the kernel sources. While many coding conventions are documented in the Documentation/CodingStyle file in the kernel source tree, many whitespace rules are typically expected to be learned by example and have not been documented anywhere, until now.

By painstakingly sifting through the code, Michael came up with a set of clear rules describing the ways in which whitespace should be utilized in the kernel: whether to put spaces around an operator, how braces should be spaced, how to handle commas and semicolons, and so on. When he posted the result, various developers helped refine and correct his initial impressions, so that

the final result could be considered quite accurate.

One of the most interesting features of a list like this is how it is enforced. Up until now, very little documentation for whitespace had existed, and developers would more or less encourage each other to use various conventions. In some cases no conventions existed, and they had to evolve gradually, as more popular unspoken preferences gained dominance over others.

Over the years, some developers have rejected the normal conventions, both the conventions regarding whitespace and other coding standards. These programmers have simply submitted patches in the form they personally thought best. It will be interesting to see how the community (and the source tree) are affected by Michael's formalization of what had been a loose and constantly evolving set of ideas.

## LINUX-KERNEL FINDS A NEW HOME (AGAIN)

As with virtually all free software projects, most aspects of Linux kernel development are dependent on contributions of hardware and networking bandwidth, in addition to the contributions of the developers themselves. Commercial vendors sometimes help out by contributing hardware and support resources to the kernel development community.

Recently, Dell donated a new computer to host the vger mailing lists, which include the linux-kernel development list, among others. Red Hat moved the machine into one of their collocation facilities and gave it a gigabit network connection. Although users didn't directly notice the switchover from the old system to the new, they sure noticed the speedup in performance. Users posting to linux-kernel have begun seeing their own posts appear just minutes after posting, while before the wait could be measured in hours.

## REPORTING USER ACTIVITY

During the recent Kernel Summit, the point was raised that it was difficult to know how much testing each kernel pre-release or release candidate actually received before the final release. Presumably, if only a few users actually test the kernels, the final release may be much buggier than otherwise. To address this issue, Andrea Arcangeli set up a server at [klive.cpushare.com](http://klive.cpushare.com) to gather statistics about exactly how much use a kernel release receives.

To maximize the data gathered on this server, all future Linux kernel releases will be configured to send data to Andrea's server on a periodic

basis. No, I'm joking! In fact, the data is submitted entirely on a volunteer basis. Interested users may run the *klive* script available at the URL given above to periodically gather and submit their usage data. Currently over 100 users are participating in the experiment. Any participants wishing to stop submitting their statistics may easily remove the software.

Andrea did consider getting rid of the client software and making the whole thing into a kernel configuration option (disabled by default), and this may still happen, but there has been some concern among kernel developers that this effort might be misconstrued by users as an attempt to surreptitiously spy on them. It's likely that this concern will keep any such patch out of the kernel proper and relegated to an optional user-space client only.

The Linux kernel mailing list comprises the core of Linux development activities. Traffic volumes are immense, often reaching ten thousand messages in a given week, and keeping up to date with the entire scope of development is a virtually impossible task for one person. One of the few brave souls to take on this task is Zack Brown.



Our regular monthly column keeps you abreast of the latest discussions and decisions, selected and summarized by Zack. Zack has been publishing a weekly online digest, the Kernel Traffic newsletter for over five years now. Even reading Kernel Traffic alone can be a time consuming task. Linux Magazine now provides you with the quintessence of Linux Kernel activities, straight from the horse's mouth.

# MAINTAINERSHIP

There have been a lot of maintainership issues in the recent past. For example, the SMBFS maintainer, Urban Widmark, has apparently not been reading email sent to his address as listed in the MAINTAINERS file. Adrian Bunk noticed this in August, but Tvrko A. Ursulin recently confirmed he'd had a similar experience months ago. As Urban could not be reached, Adrian put out the call that SMBFS needed a new maintainer.

Andrew Morton has been following the SMBFS situation with concern, and Andrew has expressed the idea that CIFS (the Common Internet File System) might be able to simply replace SMBFS entirely, at which point SMBFS could be deprecated and eventually removed from the Linux kernel. But apparently Red Hat tried this with Fedora and discovered a wide array of Windows variants that were not covered properly by CIFS, so they had to switch SMBFS back on in

their distribution. Meanwhile, the CIFS developers have been working to increase support for the various Windows variants; and at the other end of things, various folks have expressed interest in becoming the SMBFS maintainer, though so far no one has actually taken it over.

The Distributed Lock Manager patch, developed originally by Red Hat, has been receiving updates, most recently by David Teigland. David's most recent contributions updated the patch to use ConfigFS to configure lockspace members and node addresses. This had previously been done via a combination of SysFS and ioctls. But these patches had to go directly to Andrew Morton because DLM had no official maintainer. Nish Aravamudan ended up asking David directly if he wanted (or would be willing) to be the official DLM maintainer, with an entry in the MAINTAINERS file and all

the other benefits. David agreed and submitted a patch to list himself as DLM maintainer.

It seems the Advansys SCSI driver and the Rocketport driver have both been orphaned recently. Jiri Slaby noticed this and was the first to post patches to remove non-responsive emails from those driver entries in the MAINTAINERS file. In fact, as Adrian Bunk pointed out, the MAINTAINERS file doesn't tend to list every single unmaintained driver, so the better patch has been to remove the Advansys entry altogether. The Rocketport driver is perhaps a bit more robust or more generally useful than the Advansys driver – its entry was not deleted, though the bad email address was still removed. According to Adrian, there is still quite a bit of cleanup required in the MAINTAINERS file before it will reflect the true state of affairs for many drivers and subsystems.



# SUPPORTING OLDER GCC VERSIONS

Adrian Bunk has been trying to simplify the kernel sources to some extent, by removing support for older C compilers. His argument is that the newer versions are supported and available, and so the older ones just aren't needed. But if he did think so, he soon had to rethink.

It seems that GCC version 2.95 is still the fastest thing on wheels, which makes a huge difference to kernel developers compiling many kernels in a single day. As Kurt Wall put it on the developer mailing list, GCC 2.95 makes the difference between compiling overnight to find the result waiting in the morning, and that same compile continuing into the next afternoon. And many other developers spoke out against removing support for older compiler versions.

To be fair, there are good reasons for wanting to end support for older compilers. Adrian's patches would get rid of a lot of conditional code in the kernel, making the source more readable and maintainable, not to mention smaller. And he's right that modern hardware can make up for the speed sacrifice. As he points out, it would also be possible to use more modern machines to compile kernels intended to run on older, slower boxes.

Still, GCC 2.95 also runs yet faster on modern hardware, and this speedup is also valuable. With such an outcry against his patches, it does seem unlikely that GCC 2.95 support will vanish any time soon, unless future compiler versions can start to regain some of their lost speed.



# NVIDIA SECRETIVE WITH SPECS

We've known this for a long time, but it keeps coming up. Michael Thonke asked recently if there were any plans to implement NCQ support for Nvidia NForce4 (CK804) SATAII based chipsets. But as Jeff Garzik explained to him, "They are the only company that gives me zero information on their SATA controllers. As such, there are zero plans for NCQ on NVIDIA controllers at this time."

According to some kernel developers, Nvidia is entirely a Microsoft-oriented company, and diverting that focus to anything else, including Linux, simply doesn't interest them.

Some Nvidia folks, such as Allen Martin, do read the linux-kernel mailing list, and Allen has said openly that the most likely scenario for NCQ

support would be in a binary-only driver; and that Nvidia felt it just wasn't worth putting in the work of coming up with that driver, just to support that one feature. Chris Wedgwood has called for a boycott of Nvidia products, saying that constant pressure is needed in order to convince them that Linux users are an important market not to be ignored.

The strategy of constant pressure may be the best course of action when confronted by companies like Nvidia, who consistently refuse to release the documentation needed to write open source drivers for their hardware. If users don't buy that hardware, maybe as the number of Linux users grows, Nvidia will start to see reasons to release their developer guides.