Zack's Kernel News

Bug hunting

The saga of the bug database seems unlikely to resolve itself any time soon. In addition to Martin J. Bligh's and others' work on a Bugzilla system for kernel development, and John Bradford's ongoing investigation into a possible replacement, a third project has popped up to solve roughly the same set of problems.

In late January, Dan Carpenter announced Smatch, a bug hunting and tracking system based on the Stanford Checker. Unlike Bugzilla or John's database, Smatch attempts to identify bugs on its own by modifying the kernel sources, and then running scripts to identify problems in those sources. Any problems it finds are stored and organized in its own custom bug tracking database.

Smatch is still in the early stages of development, but already starting to be functional. Meanwhile, John's work on his own bug tracking system is moving right along, with a number of enhancements. As of early January users may create their own account on the web page itself, making it much easier to start using the system for serious work. If a bug report is accompanied by a .config file, it is possible for developers to download a corresponding .config file with the same options set.

Patches can also be submitted against particular bugs, and automatically checked to see if they apply against any number of kernel trees. He's also made significant advancements on a command-line interface to the system, that could operate via email. This would, among other things, allow developers to automate and refine their database searches, to prune away all reports but the most relevant to the task at hand.

Perl futures

It looks as though Perl will be required for kernel compilation in the near future. Many developers have fought against this contingency for a long time, on the grounds that only software that is virtually guaranteed to be installed on a given system, should be required for kernel compilation. Now it seems that Perl has been elevated from the status of 'just another optional tool' to that of 'oh come on, everyone has it'.

When the subject came up on the mailing list in January, Jeff Garzik revealed that the klibc code, intended to be a small subset of the C library, would soon be merged into the main kernel tree, and it has Perl dependencies. So all architectures would soon require Perl in order to build the kernel.

Folks who follow such things will remember Eric S. Raymond's bitter experience with CML2, a kernel configuration system based on Python. CML2 was flamed into oblivion, partly because of that dependency, and Eric retreated entirely from the kernel mailing list. Even now there are many developers who feel that any code currently using Perl should be rewritten in C. The argument against this is that Perl is the best tool for playing with text, and one thing any kernel configuration system will do a lot of is play with text.

Regardless of what actually happens, there will undoubtedly be developers who will forever oppose the presence of Perl, and others who will insist it should be included.

INFO

The Kernel Mailing List comprises the core of Linux development activities. Traffic volumes are immense 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 that take on this impossible task is Zack Brown.

Our regular monthly column keeps you up to date on the latest discussions and decisions, selected and summarized by Zack. Zack has been publishing a weekly digest, the Kernel Traffic Mailing List for several years now, reading just the digest is a time consuming task.

Linux Magazine now provides you with the quintessence of Linux Kernel activities straight from the horse's mouth.

The extra mile

Mel Gorman decided to go where no one had gone before, and in mid January announced actual honest-to-goodness documentation for the virtual memory subsystem. Aside from some slides presented by Andrea Arcangeli at a conference long long ago, this appears to be a first. He chose to cover the 2.4.20 kernel for his initial work, as the 2.5 tree was still in too much flux.

He actually wrote two separate documents. One was an overview of the VM subsystem, giving an organized presentation of its various pieces and how they worked. The second document dug right down into the code, giving commentary on the various functions. Something like this happens very rarely, and when it does it is accompanied by shouts of joy and happiness throughout the world.

Someone even suggested starting a SourceForge project for these docs, but Mel said he already planned eventually to merge his work into the Linux Kernel Documentation Project on GNU's Savannah site. Over the next few weeks Mel continued his work, announcing several revisions on the kernel mailing list.

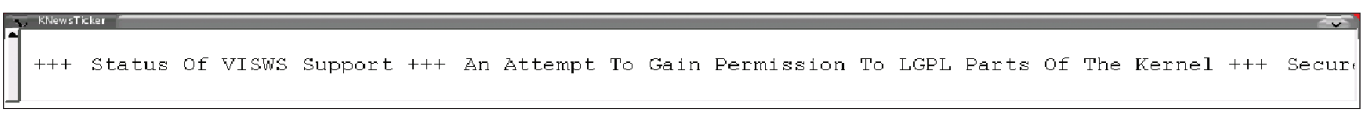

Figure 1: KNewsTicker showing that you can keep up to date with kernel traffic news