Sending data in a signal

The well-known kill system call has been around for decades and is used to send a signal to another process. The most common use is to terminate or kill another process by sending the KILL or TERM signal but it can be used for a form of IPC, usually around giving the other process a "kick" to do something.

One thing that isn't as well known is besides sending a signal to a process, you can send some data to it. This can either be an integer or a pointer and uses similar semantics to the known kill and signal handler. I came across this when there was a merge request for procps. The main changes are using sigqueue instead of kill in the sender and using a signal action not a signal handler in the receiver.

To illustrate this feature, I have a small set of programs called sender and receiver that will pass an integer between them.

Spring Issue of 2600 Released - Important News

Unfortunately, you won't be able to find this issue in stores. With most of the country and a good part of the world in quarantine, bookstores haven't been all that popular. In fact, we were told after printing the issues that our distributors ordered to not ship to them after all. We're now stuck with the entire cost of printing while distributors and stores pay nothing.

It gets worse. Our previous issue (the one still on stands) can't be sold to Barnes and Noble "curbside pickup" customers even though most everything else in their stores can be. Why? It's their "policy" that magazines can't be sold this way and that policy can't be changed despite the current circumstances. It makes no sense at all to us. Our issues are right there in the store yet they can't be sold to customers.
Rather than working out options where we might have a chance at survival, we're being told that we have to figure out what to do with all these issues or pay a penalty for not shipping them. We find ourselves in the middle of a Kafka novel where everything is stacked against the publisher because that's just how it is.

We've seen injustices before where distributors have gone out of business without paying us, sometimes simply changing their name and continuing to make millions while we don't get a dime. But this time it's different. This time what's happening affects all of us, and what we were hoping we'd see was a sense of community where we all supported one another and helped everyone get through this terrible crisis. That most certainly hasn't been the case in the publishing world.

The problem with COBOL

A lot of the blame has fallen on COBOL, a 1950s-era programming language that a lot of the systems still run on. But this isn't like old hardware breaking down, and COBOL isn't exactly broken. The problems with running a decades-old programming language are more subtle—so subtle that you can run for years without any obvious problem.

The problems only show up when you suddenly need to handle an unprecedented surge in traffic and you find out your state unemployment system can't scale up the way a service like Netflix or Zoom can. But to understand why that is, you need to understand the way network management has changed over the past 20 years (that is, the shift from pets to cattle) and how technical debt can lock you into the old way of doing things.

And most importantly... you have to look at the Big Picture.

Will Kahn-Greene: Experimenting with Symbolic

One of the things I work on is Tecken which runs Mozilla Symbols Server. It's a server that handles Breakpad symbols files upload, download, and stack symbolication.

Bug #1614928 covers adding line numbers to the symbolicated stack results for the symbolication API. The current code doesn't parse line records in Breakpad symbols files, so it doesn't know anything about line numbers. I spent some time looking at how much effort it'd take to improve the hand-written Breakpad symbol file parsing code to parse line records which requires us to carry those changes through to the caching layer and some related parts—it seemed really tricky.

That's the point where I decided to go look at Symbolic which I had been meaning to look at since Jan wrote the Native Crash Reporting: Symbol Servers, PDBs, and SDK for C and c++ blog post a year ago.
Major R language update brings big changes [6]

Version 4.0.0 of the R language for statistical computing has been released, with changes to the syntax of the language as well as features pertaining to error-checking and long vectors.

The upgrade was published on April 24. Source code for R 4.0.0 is accessible at cran.r-project.org. A GNU project, R has gathered steam with the rise of data science and machine learning, currently ranking 10th in the Tiobe Index of language popularity and seventh in the PyPL Popularity of Programming Language index.

IBM announces Elyra AI Toolkit, a set of AI-centric extensions to Jupyter Notebooks [7]

Jupyter Notebooks are now the open standard for data science and artificial intelligence (AI) model development. In keeping with our commitment to open source and the Jupyter community, in particular, IBM is proud to announce Elyra, a set of open source AI-centric extensions to Jupyter Notebooks, and, more specifically, the new JupyterLab user interface.

How to use count() method in python [8]

The built-in count() method of Python is very helpful when we need to find out how many times a particular string appears in a text or how many times an item appears in a tuple or list. By default, this method will search a particular sub-string in the whole content of a particular string but this method can also be used to search the sub-string in the particular portion of the main string. The uses of count() method in Python are explained in this article using multiple examples.

How to use a break and continue statement within a loop in Python [9]

Break and continue statements are used inside the loop of any programming language for different purposes. These two statements are considered as jump statements because both statements move the control from one part to another part of the script. The break statement is used within any loop to terminate the loop based on any specific condition before the termination condition appears. The continue statement is used within any loop to omit one or more statements of the loop based on any specific condition but it is not used to terminate the loop. How these statements are used inside the python loop are shown in this tutorial.

Dealing with an all-CAPS/first-CAP jumble [10]
I sometimes need to tally lists of single words in which the same word might appear capitalised or all in capital letters.

**Development**

**Source URL:** [http://www.tuxmachines.org/node/136983](http://www.tuxmachines.org/node/136983)

**Links:**