acmqueue

Port Squatting

Don't irk your local sysadmin

Dear KV,

A few years ago you upbraided some developers for not following the correct process when requesting a reserved network port from IETF (Internet Engineering Task Force). While I get that squatting a used port is poor practice, I wonder if you, yourself, have ever tried to get IETF to allocate a port. We recently went through this with a new protocol on an open-source project, and it was a nontrivial and frustrating exercise. While I wouldn't encourage your readers to squat ports, I can see why they might just look for unallocated ports on their own and simply start using those, with the expectation that if their protocols proved popular, they would be granted the allocations later.

Frankly Frustrated

Dear Frankly,

Funny you should ask this question at this point. This summer I, too, requested not one, but two ports for a service I'd been working on (Conductor: <https://github.com/gvnn3/conductor>). I've always been annoyed that there isn't a simple, distributed, automation system for orchestrating network tests, so I sat down and wrote one. The easiest way to implement the system was to have two reserved ports—one for the conductor and one for the players—so that each could contact the others independently without have to pass ephemeral qports around after they were allocated by the operating system during process startup.

Simple enough, you might think. It's not actually IETF to which one applies—it's IANA (Internet Assigned Numbers Authority). It has a form you fill out on its Web site, detailing your request (<https://www.iana.org/form/ports-services>), which asks fairly reasonable questions about who you are, which transport protocol your protocol uses (UDP, TCP, SCTP, etc.), and how the protocol is used. Because there are only 16 bits in the port field for UDP, TCP, and SCTP, space is limited, so you can see why IANA would want to be careful in its port allocations. Looking over the current assignments, we can see that nearly 10 percent of the space has already been allocated for TCP, with more than 6,100 assigned ports for TCP.

I submitted my request for a pair of ports over TCP and SCTP on July 7. I applied for both because it made sense to address both of the currently available, reliable, transport protocols. I was assured that by September 8 I would have a single port assigned, and finally got it on the 18th. Let's look at the process.

Once you submit your port request, it goes into a ticketing system, RT (request tracker), which is looked after by someone whom I'll call a secretary. The secretary seems to do some form of triage on the ticket and then passes it along to someone else. For the past two months, the secretary asked clarifying questions about the use of the two port numbers. It was plain from the interaction that the secretary did not have any significant networking knowledge but acted as a pass-through for the experts reviewing the case. As might be expected with any sort of overly bureaucratic process and with this form of telephone game, information was often lost or duplicated, requiring me to explain

at length how I was going to use the ports. In the end, I was contacted by an expert—someone actually knowledgeable about networking technology—and we agreed that the service could be built with one port number. I say “agreed,” but mostly I relented, because I was going to do this right, even if I put my fist through a whiteboard—and let me tell you, I came very close to just that.

This brings me to a few statistics about the assigned numbers. Many of the assignments for TCP aren’t for a single port, but for multiple ports, meaning the number of services is fewer than the 6,100-plus assigned ports. Not only are there many services with more than one port, but it would also seem that dead assignments are not garbage collected, which means that although only 10 percent of the space is used, there is no way to reclaim ports when protocols or the companies that created them die. Looking through the list of assigned ports is a walk down the memory lane of failed companies.

All of which is not to encourage people to squat numbers, but it is pretty clear that IANA could do some work to streamline the process, as well as reclaim some of the used space. The biggest problem actually exists in the first 1,024 ports, which most operating systems consider to be “system” ports. A system port is usable only by a service running as root, and this is considered privileged. The domain name system, for example, runs on port 53. It’s in this low space that IANA needs to get its collective act together and kill off a few services. Although I’m sure all of you are using port 222—Berkeley rshd with SPX auth—each and every day.

KV

Dear KV,

I’ve been revising the logging output for a large project, and it seems every time I propose a change, our systems admins start screaming at me to revert what I’ve done. They seem to think that the format of our log output was set in stone at version 1 of the product and that I shouldn’t actually touch anything, even though the product—now at version 3—does quite a bit more than it did in version 1. I understand that changing the output means they’ll have to change some scripts, but I can’t help it if there are new features that need to log new information.

Log Rolled

Dear Logged,

I don’t know if you know this, but systems administrators are simply lazy, drunken layabouts who spend all their days slacking off work, putting their feet up the desk, and sipping single malts while the boss isn’t looking. Actually—in point of fact—systems administrators are often the busiest and most harried people at any IT site, and they are the people responsible for knowing if all the systems are UP or DOWN. If you capriciously change the logging output on their systems, the tools they have lovingly crafted to track the performance of your system will indicate things are DOWN when they’re probably not, and this will result in a lot of screaming. I like coding in a quiet environment; I do not like screaming, so do not make the sysadmins scream.

There are good ways and bad ways to update log output. Inserting a new column at the beginning of each line, thereby throwing off all the following columns, is an incredibly bad way of updating a log file. In fact, the first columns of any log output should always be the date and time—with seconds. Using the date for the first column makes writing analysis scripts far easier. Just like extending a programming API, unless you have a very good reason, you should always add new

information at the end of the line. Extra columns are the easiest to ignore and the least likely to cause the sysadmin tools to go nutty. That therefore reduces the amount and volume of screaming in the office (see above about offices and quiet).

Another less offensive way of updating log output is to add whole new lines of information, so that scripts can look at the old lines correctly and, for as long as possible, ignore the new information. Allowing the script authors some time to update their scripts is a kindness that is repaid in free liquor at conferences, and is just the kind of thing you would want to encourage.

Finally, you might simply add an option to the program to output the old log format so that the people running your software have time, again, to update their scripts, or, perhaps, they really don't need the new information and would like the chance to use your system without touching their pristine and beautiful scripts. Think first before forcing new information on the user.

KV

LOVE IT, HATE IT? LET US KNOW

feedback@queue.acm.org

KODE VICIOUS, known to mere mortals as George V. Neville-Neil, works on networking and operating system code for fun and profit. He also teaches courses on various subjects related to programming. His areas of interest are code spelunking, operating systems, and rewriting your bad code (OK, maybe not that last one). He earned his bachelor's degree in computer science at Northeastern University in Boston, Massachusetts, and is a member of ACM, the Usenix Association, and IEEE. He is an avid bicyclist and traveler who currently lives in New York City.

© 2014 ACM 1542-7730/14/0900 \$10.00