
Subject: Services running on CT0 and VEs?

Posted by [jantman](#) on Sat, 20 Mar 2010 01:01:29 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hello,

I'm in the process of evaluating OpenVZ and Xen for a production virtualization project. My major goals are consolidation and fault-tolerance (the latter of which I still have to look into a bit more). I really like OpenVZ's lower overhead, as we run a standardized environment (all Linux servers run CentOS 5, same kernel, all built from the same basic installation).

We use Puppet to manage all of our servers. I happened to be looking around online, and came by <http://forum.openvz.org/index.php?t=msg&goto=14818&>, a support forum post about stopping the puppet service on CT0 stops is on CTx's.

The root cause appeared to be that the init scripts use killproc, which on CT0 kill all processes with a given name, even those inside containers.

Does anyone have any experience with this? What are the workarounds? Replacing killproc with a script that makes sure it only kills CT0 processes? Altering all of the init scripts for every service?

This is a major concern to me, as our standardized install includes a number of services that *must* run on every host, regardless of whether it's physical or virtual - SSHd, rsyslog, denyhosts, iptables, puppet, etc.

Thanks for any pointers.

-Jason

Subject: Re: Services running on CT0 and VEs?

Posted by [kir](#) on Sat, 20 Mar 2010 20:50:23 GMT

[View Forum Message](#) <> [Reply to Message](#)

It's pretty easy -- one should not kill the processes using the process name, because it relies on a false assumption that process name is unique to the software instance -- in facts it is not, and you'll clearly see that if you'd run two instances of apache or mysqld (which is not uncommon).

Therefore, most daemons and their initscripts work the following way -- then a daemon is started it saves its pid into a file, and then the initscript wants to stop the daemon (or make it reload its config, or do anything else like that), it uses the PID from the file. This is true for most services and definitely sshd and rsyslog from your list work this way (iptables is not a daemon and I do not know anything about denyhosts).

This is what puppet and puppet initscript should do -- save PID at start and use it to shutdown.

If this is absolutely impossible, then killproc should be modified to filter out processes not belonging to host system. This is pretty easy since each /proc/\$PID/status have a field named

envID: -- if its value is zero this means host system, otherwise it's a process from some container. My guess is such filtering can be implemented in one or at most 34 lines of shell, so it's not a big deal.
