Subject: Re: [Vserver] Re: Container Test Campaign
Posted by Clement Calmels on Wed, 05 Jul 2006 07:40:28 GMT
View Forum Message <> Reply to Message

Hi,

> In general - please don't get the impression I try to be fastidious. I'm
> just trying to help you create a system in which results can be
> reproducible and trusted. There are a lot of factors that influence the
> performance; some of those are far from being obvious.

Don't get me wrong I'm looking for such remarks :)

> IMO you should add a reboot here, in between _different_ tests, just
> because previous tests should not influence the following ones.
> Certainly you do not need a reboot before iterations of the same test.

I don't do this first because I didn't want to get test nodes wasting
their time rebooting instead of running test. What do you think of
something like this:
o reboot
o run dbench (or wathever) X times
o reboot

> > For test inside a 'guest' I just do something like:
> > o build the appropriate kernel (2.6.16-026test014-x86_64-smp for
> > example)
> > o reboot
> >
> Here you do not have to reboot. OpenVZ tools does not require OpenVZ
> kernel to be built.

You got me... I was still believing the VZKERNEL_HEADERS variable was
needed. Things have changed since vzctl 3.0.0-4...

> > o build the utilities (vztcl+vzquota for example)
> > o reboot
> > o launch a guest
> >
> Even this part is tricky! You haven't specified whether you create the
> guest before or after the reboot. Let me explain.
>
> If you create a guest before the reboot, the performance (at least at
> the first iteration) could be a bit higher than if you create a guest
> after the reboot. The reason is in the second case the buffer cache will
> be filled with OS template data (which is several hundred megs).
can
I can split the "launch a guest" part into 2 parts:

o guest creation
o reboot
o guest start-up
Do you feel comfortable with that?

> > -The results are the average value of several iterations of each set of
> > these kind of tests.
> Hope you do not recompile the kernels before the iterations (just to
> speed things up).
> > I will try to update the site with the numbers of
> > iterations behind each values.
> >
> Would be great to have that data (as well as the results of the
> individual iterations, and probably graphs for the individual iterations
> -- to see the "warming" progress, discrepancy between iterations,
> degradation over iterations (if that takes place) etc).

I will try to get/show those datas.

> The same will happen with most of the other tests involving I/O. Thus,
> test results will be non-accurate. To achieve more accuracy and exclude
> the impact of the disk and filesystem layout to the results, you should
> reformat the partition you use for testing each time before the test.
> Note that you don't have to reinstall everything from scratch -- just
> use a separate partition (mounted to say /mnt/temptest) and make sure
> most of the I/O during the test happens on that partition.

It would be possible for 'host' node... inside the 'guest' node, I don't
know if it makes sense. Just adding an 'external' partition to the
'guest' for I/O test purpose? For example in an OpenVZ guest, creating a
new and empty simfs partition in order to run test on it?

> > - For the settings of the guest I tried to use the default settings (I
> > had to change some openvz guest settings) just following the HOWTO on
> > vserver or openvz site.
> > For the kernel parameters, did you mean kernel config file tweaking?
> >
> No I mean those params from /proc/sys (== /etc/sysctl.conf). For
> example, if you want networking for canOpenVZ guests, you have to turn on
> ip_forwarding. There are some params affecting network performance, such
> as various gc_thresholds. For the big number of guests, you have to tune
> some system-wide parameters as well.

For the moment, I just follow the available documentation:
 http://wiki.openvz.org/Quick_installation#Configuring_sysctl _settings
Do you think these paramenters can hardly affect network performance?
>From what I understand lot of them are needed.

> > - All binaries are always build in the test node.
> >
> I assuming you are doing your tests on the same system (i.e. same
> compiler/libs/whatever else), and you do not change that system over
> time (i.e. you do not upgrade gcc on it in between the tests).

I hope! :)
--
Clément Calmels <clement.calmels@fr.ibm.com>
