Subject: Re: [RFC][PATCH 2/7] RSS controller core
Posted by Dave Hansen on Mon, 12 Mar 2007 23:02:08 GMT
View Forum Message <> Reply to Message

On Mon, 2007-03-12 at 23:41 +0100, Herbert Poetzl wrote:
> On Mon, Mar 12, 2007 at 11:42:59AM -0700, Dave Hansen wrote:
> > How about we drill down on these a bit more.
> >
> > On Mon, 2007-03-12 at 02:00 +0100, Herbert Poetzl wrote:
> > > - shared mappings of 'shared' files (binaries
> > >   and libraries) to allow for reduced memory
> > >   footprint when N identical guests are running
> >
> > So, it sounds like this can be phrased as a requirement like:
> >
> > "Guests must be able to share pages."
> >
> > Can you give us an idea why this is so?
>
> sure, one reason for this is that guests tend to
> be similar (or almost identical) which results
> in quite a lot of 'shared' libraries and executables
> which would otherwise get cached for each guest and
> would also be mapped for each guest separately
>
> > On a typical vserver system,
>
> there is nothing like a typical Linux-VServer system :)
>
> > how much memory would be lost if guests were not permitted
> > to share pages like this?
>
> let me give a real world example here:
>
> - typical guest with 600MB disk space
> - about 100MB guest specific data (not shared)
> - assumed that 80% of the libs/tools are used

I get the general idea here, but I just don't think those numbers are
very accurate. My laptop has a bunch of gunk open (xterm, evolution,
firefox, xchat, etc...). I ran this command:

lsof | egrep '/(usr/|lib.*\.so)' | awk '{print $9}' | sort | uniq | xargs du -Dcs

and got:

113840 total

On a web/database server that I have (ps aux | wc -l == 128), I just ran the same:

39168 total

That's assuming that all of the libraries are fully read in and populated, just by their on-disk sizes. Is that not a reasonable measure of the kinds of things that we can expect to be shared in a vserver? If so, it's a long way from 400MB.

Could you try a similar measurement on some of your machines? Perhaps mine are just weird.

> > > - virtual 'physical' limit should not cause
> > > swap out when there are still pages left on
> > > the host system (but pages of over limit guests
> > > can be preferred for swapping)
> >
> > Is this a really hard requirement?
>
> no, not hard, but a reasonable optimization ...
>
> let me note once again, that for full isolation
> you better go with Xen or some other Hypervisor
> because if you make it work like Xen, it will
> become as slow and resource hungry as any other
> paravirtualization solution ...

Believe me, _I_ don't want Xen. :)

> > It seems a bit fluffy to me.
>
> most optimizations might look strange at first
> glance, but when you check what the limitting
> factors for OS-Level virtualizations are, you
> will find that it looks like this:
>
> (in order of decreasing relevance)
>
> - I/O subsystem
> - available memory
> - network performance
> - CPU performance
>
> note: this is for 'typical' guests, not for
> number crunching or special database, or pure
> network bound applications/guests ...

I don't doubt this, but doing this two-level page-out thing for
containers/vservers over their limits is surely something that we should
consider farther down the road, right?

It's important to you, but you're obviously not doing any of the
mainline coding, right?

> > What are the consequences if this isn't done? Doesn't
> > a loaded system eventually have all of its pages used
> > anyway, so won't this always be a temporary situation?
>
> let's consider a quite limited guest (or several
> of them) which have a 'RAM' limit of 64MB and
> additional 64MB of 'virtual swap' assigned ...
>
> if they use roughly 96MB (memory footprint) then
> having this 'fluffy' optimization will keep them
> running without any effect on the host side, but
> without, they will continously swap in and out
> which will affect not only the host, but also the
> other guests ...

All workloads that use $limit+1 pages of memory will always pay the
price, right? :)

-- Dave

_____
Containers mailing list
Containers@lists.osdl.org
https://lists.osdl.org/mailman/listinfo/containers