Subject: Re: [PATCH v5 0/8] per-cgroup tcp buffer pressure settings
Posted by KAMEZAWA Hiroyuki on Fri, 07 Oct 2011 08:55:00 GMT
View Forum Message <> Reply to Message

On Fri, 7 Oct 2011 12:20:04 +0400
Glauber Costa <glommer@parallels.com> wrote:

> >
> >> So what I really mean here with "will integrate later", is that I think
> >> that we'd be better off tracking the allocations themselves at the slab
> >> level.
> >>
> >>> Can't tcp-limit-code borrows some amount of charges in batch from kmem_limit
> >>> and use it ?
> >> Sorry, I don't know what exactly do you mean. Can you clarify?
> >>
> > Now, tcp-usage is independent from kmem-usage.
> >
> > My idea is
> >
> > 1. when you account tcp usage, charge kmem, too.
>
> Absolutely.
> > Now, your work is
> > a) tcp use new xxxx bytes.
> > b) account it to tcp.uage and check tcp limit
> >
> > To ingegrate kmem,
> > a) tcp use new xxxx bytes.
> > b) account it to tcp.usage and check tcp limit
> > c) account it to kmem.usage
> >
> > ? 2 counters may be slow ?
>
> Well, the way I see it, 1 counter is slow already =)
> I honestly think we need some optimizations here. But
> that is a side issue.
>
> To begin with: The new patchset that I intend to spin
> today or Monday, depending on my progress, uses res_counters,
> as you and Kirill requested.
>
> So what makes res_counters slow IMHO, is two things:
>
> 1) interrupts are always disabled.
> 2) All is done under a lock.
>
> Now, we are starting to have resources that are billed to multiple

> counters. One simple way to work around it, is to have child counters
> that has to be accounted for as well everytime a resource is counted.
>
> Like this:
>
> 1) tcp has kmem as child. When we bill to tcp, we bill to kmem as well.
> For protocols that do memory pressure, we then don't bill kmem from
> the slab.
> 2) When kmem_independent_account is set to 0, kmem has mem as child.
>

Seems reasonable.

> >
> >
> >>> - Don't you need a stat file to indicate "tcp memory pressure works!" ?
> >>> It can be obtained already ?
> >>
> >> Not 100 % clear as well. We can query the amount of buffer used, and the
> >> amount of buffer allowed. What else do we need?
> >>
> >
> > IIUC, we can see the fact tcp.usage is near to tcp.limit but never can see it
> > got memory pressure and how many numbers of failure happens.
> > I'm sorry if I don't read codes correctly.
>
> IIUC, With res_counters being used, we get at least failcnt for free, right?
>

Right. you can get failcnt and max_usage and can have soft_limit base
implemenation at the same time.

Thank you.
-Kame