Subject: Re: Re: [PATCH v5 00/10] per-cgroup tcp memory pressure
Posted by davem on Fri, 18 Nov 2011 19:51:07 GMT
View Forum Message <> Reply to Message

From: Glauber Costa <glommer@parallels.com>
Date: Fri, 18 Nov 2011 17:39:03 -0200

> On 11/17/2011 07:35 PM, David Miller wrote:
>> From: James Bottomley<jbottomley@parallels.com>
>> Date: Tue, 15 Nov 2011 18:27:12 +0000
>>
>>> Ping on this, please. We're blocked on this patch set until we can
>>> get
>>> an ack that the approach is acceptable to network people.
>>
>> __sk_mem_schedule is now more expensive, because instead of
>> short-circuiting
>> the majority of the function's logic when "allocated<=
>> prot->sysctl_mem[0]"
>> and immediately returning 1, the whole rest of the function is run.
>
> Not the whole rest of the function. Rather, just the other two
> tests. But that's the behavior we need since if your parent is on
> pressure, you should be as well. How do you feel if we'd also provide
> two versions for this:
> 1) non-cgroup, try to return 1 as fast as we can
> 2) cgroup, also check your parents.

Fair enough.

> How about we make the jump_label only used for sockets (which is basic
> what we have now, just need a clear name to indicate that), and then
> enable it not when the first non-root cgroup is created, but when the
> first one sets the limit to something different than unlimited?
>
> Of course to that point, we'd be accounting only to the root
> structures,
> but I guess this is not a big deal.

This sounds good for now.

>> TCP specific stuff in mm/memcontrol.c, at best that's not nice at all.
>
> How crucial is that?

It's a big deal. We've been working for years to yank protocol specific
things even out of net/core/*.c, it simply doesn't belong there.

I'd even be happier if you had to create a net/ipv4/tcp_memcg.c and include/net/tcp_memcg.h

> Thing is that as far as I am concerned, all the
> memcg people
...

What the memcg people want is entirely their problem, especially if it involves crapping up non-networking files with protocol specific junk.
