
Subject: Re: [RFC][PATCH 1/7] Resource counters
Posted by [Herbert Poetzl](#) on Tue, 13 Mar 2007 15:21:50 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Tue, Mar 13, 2007 at 03:09:06AM -0600, Eric W. Biederman wrote:

> Herbert Poetzl <herbert@13thfloor.at> writes:

>

>> On Sun, Mar 11, 2007 at 01:00:15PM -0600, Eric W. Biederman wrote:

>>> Herbert Poetzl <herbert@13thfloor.at> writes:

>>>

>>>>

>>>> Linux-VServer does the accounting with atomic counters,

>>>> so that works quite fine, just do the checks at the

>>>> beginning of whatever resource allocation and the

>>>> accounting once the resource is acquired ...

>>>

>>> Atomic operations versus locks is only a granularity thing.

>>> You still need the cache line which is the cost on SMP.

>>>

>>> Are you using `atomic_add_return` or `atomic_add_unless` or

>>> are you performing you actions in two separate steps

>>> which is racy? What I have seen indicates you are using

>>> a racy two separate operation form.

>>

>> yes, this is the current implementation which

>> is more than sufficient, but I'm aware of the

>> potential issues here, and I have an experimental

>> patch sitting here which removes this race with

>> the following change:

>>

>> - doesn't store the accounted value but

>> limit - accounted (i.e. the free resource)

>> - uses `atomic_add_return()`

>> - when negative, an error is returned and

>> the resource amount is added back

>>

>> changes to the limit have to adjust the 'current'

>> value too, but that is again simple and atomic

>>

>> best,

>> Herbert

>>

>> PS: `atomic_add_unless()` didn't exist back then

>> (at least I think so) but that might be an option

>> too ...

>

> I think as far as having this discussion if you can remove that race

> people will be more willing to talk about what vserver does.

well, shouldn't be a big deal to brush that patch up
(if somebody actually `_is_` interested)

> That said anything that uses locks or atomic operations (finer grained
> locks) because of the cache line ping pong is going to have scaling
> issues on large boxes.

right, but atomic ops have much less impact on most
architectures than locks :)

> So in that sense anything short of per cpu variables sucks at scale.
> That said I would much rather get a simple correct version without the
> complexity of per cpu counters, before we optimize the counters that
> much.

actually I thought about per cpu counters quite a lot, and
we (Linux-VServer) use them for accounting, but please
tell me how you use per cpu structures for implementing
limits

TIA,
Herbert

> Eric

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