Subject: Re: [RFC][PATCH 1/7] Resource counters
Posted by xemul on Tue, 13 Mar 2007 15:41:05 GMT
View Forum Message <> Reply to Message

>>> 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 :)

Right. But atomic_add_unless() is slower as it is
essentially a loop. See my previous letter in this sub-thread.

>> 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 (Llinux-VServer) use them for accounting, but please
> tell me how you use per cpu structures for implementing
> limits

Did you ever look at how get_empty_filp() works?
I agree, that this is not a "strict" limit, but it
limits the usage wit some "precision".

/* off-the-topic */ Herbert, you've lost Balbir again:
In this sub-thread some letters up Eric wrote a letter with
Balbir in Cc:. The next reply from you doesn't include him.

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