
Subject: Re: [PATCH] Show slab memory usage on OOM and SysRq-M
Posted by [Pekka Enberg](#) on Tue, 17 Apr 2007 14:02:46 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Pavel,

At some point in time, I wrote:

> > So, now we have two locks protecting cache_chain? Please explain why
> > you can't use the mutex.

On Tue, 17 Apr 2007, Pavel Emelianov wrote:

> Because OOM can actually happen with this mutex locked. For example
> kmem_cache_create() locks it and calls kmalloc(), or write to
> /proc/slabinfo also locks it and calls do_tune_cpu_caches(). This is
> very rare case and the deadlock is VERY unlikely to happen, but it
> will be very disappointing if it happens.
>
> Moreover, I put the call to show_slabs() into sysrq handler, so it may
> be called from atomic context.
>
> Making mutex_trylock() is possible, but we risk of loosing this info
> in case OOM happens while the mutex is locked for cache shrinking (see
> cache_reap() for example)...
>
> So we have a choice - either we have an additional lock on a slow and
> rare paths and show this info for sure, or we do not have a lock, but
> have a risk of loosing this info.

I don't worry about performance as much I do about maintenance. Do you know if mutex_trylock() is a problem in practice? Could we perhaps fix the worst offenders who are holding cache_chain_mutex for a long time?

In any case, if we do end up adding the lock, please add a BIG FAT COMMENT explaining why we have it.

At some point in time, I wrote:

> > I would also drop the OFF_SLAB bits because it really doesn't matter
> > that much for your purposes. Besides, you're already per-node and
> > per-CPU caches here which attribute to much more memory on NUMA setups
> > for example.

On Tue, 17 Apr 2007, Pavel Emelianov wrote:

> This gives us a more precise information :) The precision is less than 1%
> so if nobody likes/needs it, this may be dropped.

My point is that the "precision" is useless here. We probably waste more memory in the caches which are not accounted here. So I'd just drop it.
