Subject: Re: [PATCH] Show slab memory usage on OOM and SysRq-M
Posted by Eric Dumazet on Tue, 17 Apr 2007 15:12:13 GMT
View Forum Message <> Reply to Message

On Tue, 17 Apr 2007 16:22:48 +0300
"Pekka Enberg" <penberg@cs.helsinki.fi> wrote:

```
> Hi,
>
> On 4/17/07, Pavel Emelianov <xemul@sw.ru> wrote:
> > +static unsigned long get_cache_size(struct kmem_cache *cachep)
> > +{
> > +	unsigned long slabs;
> > +	struct kmem_list3 *l3;
> > +	struct list_head *lh;
> > +	int node;
> > +
> > +	slabs = 0;
> > +
> > +	for_each_online_node (node) {
> > +		l3 = cachep->nodelists[node];
> > +		if (l3 == NULL)
> > +			continue;
> > +
> > +		spin_lock(&l3->list_lock);
> > +		list_for_each (lh, &l3->slabs_full)
> > +			slabs++;
> > +		list_for_each (lh, &l3->slabs_partial)
> > +			slabs++;
> > +		list_for_each (lh, &l3->slabs_free)
> > +			slabs++;
> > +		spin_unlock(&l3->list_lock);
> > +	}
> > +
> > +	return slabs * ((PAGE_SIZE << cachep->gfporder) +
> > +		(OFF_SLAB(cachep) ? cachep->slabp_cache->buffer_size : 0));
> > +}
>
> Considering you're doing this at out_of_memory() time, wouldn't it
> make more sense to add a ->nr_pages to struct kmem_cache and do the
> tracking in kmem_getpages/kmem_freepages?
>
```

To avoid a deadlock ? yes...

This nr_pages should be in struct kmem_list3, not in struct kmem_cache, or else you defeat NUMA optimizations if touching a field in kmem_cache at kmem_getpages()/kmem_freepages() time.

```
for_each_online_node (node) {
        l3 = cachep->nodelists[node];
        if (l3)
            slabs += l3->nr_pages; /* dont lock l3->list_lock */
}
```