Subject: Re: [PATCH 5/5] Account for the slub objects
Posted by Pavel Emelianov on Mon, 01 Oct 2007 14:10:11 GMT
View Forum Message <> Reply to Message

[snip]

>> +int slub_alloc_notify(struct kmem_cache *s, void *obj, gfp_t gfp)
>> +{
>> + struct page *pg;
>> + struct kmem_container *cnt;
>> + struct kmem_container **obj_container;
>> +
>> + pg = virt_to_head_page(obj);
>> + obj_container = pg->cgroups;
>> + if (unlikely(obj_container == NULL)) {
>> + /*
>> + * turned on after some objects were allocated
>> + */
>> + if (slub_newpage_notify(s, pg, GFP_ATOMIC) < 0)
>> + goto err;
>> +
>> + obj_container = pg->cgroups;
>> + }
>> +
>> + rcu_read_lock();
>> + cnt = task_kmem_container(current);
>> + if (res_counter_charge(&cnt->res, s->size))
>
> Is s->size measure in pages or bytes? I suspect it is bytes.

In bytes, of course :) the struct anon_vma (for example) is difficult
to account in pages :P

>> + goto err_locked;
>> +
>> + css_get(&cnt->css);
>> + rcu_read_unlock();
>> + obj_container[slab_index(obj, s, page_address(pg))] = cnt;
>> + return 0;
>> +
>> +err_locked:
>> + rcu_read_unlock();
>> +err:
>> + return -ENOMEM;
>> +}
>> +
>> +void slub_free_notify(struct kmem_cache *s, void *obj)
>> +{

```
>> + struct page *pg;
>> + struct kmem_container *cnt;
>> + struct kmem_container **obj_container;
>> +
>> + pg = virt_to_head_page(obj);
>> + obj_container = pg->cgroups;
>> + if (obj_container == NULL)
>> +  return;
>> +
>> + obj_container += slab_index(obj, s, page_address(pg));
>> + cnt = *obj_container;
>> + if (cnt == NULL)
>> +  return;
>> +
>> + res_counter_uncharge(&cnt->res, s->size);
>> + *obj_container = NULL;
>> + css_put(&cnt->css);
>> +}
>> +
>
> Quick check, slub_free_notify() and slab_alloc_notify() are called
> from serialized contexts, right?
```

Yup.

```
>> +int slub_on_notify(struct kmem_cache *cachep)
>> +{
>> + return (is_kmalloc_cache(cachep) ? -EINVAL : 0);
>> +}
>> +
>
> I know you've mentioned several times in the comments that
> kmalloc slab's cannot be accounted for, could you please
> add a big comment here as well.
```

This is useless comment - I will implement the kmalloc accounting
soon, so it will just go away.

```
> Also, looks like if I turn on notification for kmalloc
```

You cannot, but let's go on with this assumption :)

```
> slab's, the listener fails, which will cause all
> allocations to fail?
```

Nope, just this cache will not generate events and that's it.

_____
Containers mailing list