Subject: Re: [RFC][PATCH 5/7] UBC: kernel memory accounting (core)
Posted by dev on Thu, 17 Aug 2006 13:33:44 GMT
View Forum Message <> Reply to Message

```
> On Wed, 2006-08-16 at 11:47 -0700, Dave Hansen wrote:
>
>>On Wed, 2006-08-16 at 19:40 +0400, Kirill Korotaev wrote:
>>
>>>--- ./include/linux/mm.h.kmemcore	2006-08-16 19:10:38.000000000
>>>+0400
>>>+++ ./include/linux/mm.h	2006-08-16 19:10:51.000000000 +0400
>>>@@ -274,8 +274,14 @@ struct page {
>>>	unsigned int gfp_mask;
>>>	unsigned long trace[8];
>>> #endif
>>>+#ifdef CONFIG_USER_RESOURCE
>>>+	union {
>>>+		struct user_beancounter *page_ub;
>>>+	} bc;
>>>+#endif
>>> };
>>
>>Is everybody OK with adding this accounting to the 'struct page'?
>
>
> My preference would be to have container (I keep on saying container,
> but resource beancounter) pointer embeded in task, mm(not sure),
> address_space and anon_vma structures.  This should allow us to track
> user land pages optimally.  But for tracking kernel usage on behalf of
> user, we will have to use an additional field (unless we can re-use
> mapping).  Please correct me if I'm wrong, though all the kernel
> resources will be allocated/freed in context of a user process.  And at
> that time we know if a allocation should succeed or not.  So we may
> actually not need to track kernel pages that closely.  We are not going
> to run reclaim on any of them anyways.
```

objects are really allocated in process context
(except for TCP/IP and other softirqs which are done in arbitrary
process context!)
And objects are not always freed in correct context (!).

Note, page_ub is not for _user_ pages. user pages accounting will be added
in next patch set. page_ub is added to track kernel allocations.

Kirill