Subject: Re: [PATCH 2/3] memcgroup: fix typo in VM_BUG_ON()
Posted by Balbir Singh on Mon, 18 Feb 2008 07:01:27 GMT
View Forum Message <> Reply to Message

Li Zefan wrote:
> Balbir Singh wrote:
>> YAMAMOTO Takashi wrote:
>>>> Li Zefan wrote:
>>>>> No need for VM_BUG_ON(pc), since 'pc' is the list entry. This should
>>>>> be VM_BUG_ON(page).
>>>>>
>>>>> Signed-off-by: Li Zefan <lizf@cn.fujitsu.com>
>>>>> Acked-by: KAMEZAWA Hiroyuki <kamezawa.hiroyu@jp.fujitsu.com>
>>>> pc is of type page_cgroup and we use list_for_each_entry_safe_reverse. Not sure
>>>> why we can't bug on pc.
>>> pc is dereferenced before this VM_BUG_ON.
>>>
>>> YAMAMOTO Takashi
>>>
>> OK, so the VM_BUG_ON needs to move to an earlier location. Agreed.
>>
>
> No, 'pc' has been dereferenced in list_for_each_entry_safe_reverse().
>
>
> #define list_for_each_entry_safe_reverse(pos, n, head, member) \
> for (pos = list_entry((head)->prev, typeof(*pos), member), \
>   n = list_entry(pos->member.prev, typeof(*pos), member); \
>   ^^^^^^^^^^^^
>   &pos->member != (head);   \
>   ^^^^^^^^^^^
>   pos = n, n = list_entry(n->member.prev, typeof(*n), member))
>

Hmm.. We used to have a for loop with !list_empty() as a termination condition
and VM_BUG_ON(!pc) is a spill over. With the new loop, VM_BUG_ON(!pc) does not
make sense.

--
 Warm Regards,
 Balbir Singh
 Linux Technology Center
 IBM, ISTL

_____

Subject: Re: [PATCH 2/3] memcgroup: fix typo in VM_BUG_ON()
Posted by Li Zefan on Mon, 18 Feb 2008 07:14:25 GMT
View Forum Message <> Reply to Message

Balbir Singh wrote:
> Li Zefan wrote:
>> Balbir Singh wrote:
>>> YAMAMOTO Takashi wrote:
>>>>> Li Zefan wrote:
>>>>>> No need for VM_BUG_ON(pc), since 'pc' is the list entry. This should
>>>>>> be VM_BUG_ON(page).
>>>>>>
>>>>>> Signed-off-by: Li Zefan <lizf@cn.fujitsu.com>
>>>>>> Acked-by: KAMEZAWA Hiroyuki <kamezawa.hiroyu@jp.fujitsu.com>
>>>>> pc is of type page_cgroup and we use list_for_each_entry_safe_reverse. Not sure
>>>>> why we can't bug on pc.
>>>> pc is dereferenced before this VM_BUG_ON.
>>>>
>>>> YAMAMOTO Takashi
>>>>
>>> OK, so the VM_BUG_ON needs to move to an earlier location. Agreed.
>>>
>> No, 'pc' has been dereferenced in list_for_each_entry_safe_reverse().
>>
>>
>> #define list_for_each_entry_safe_reverse(pos, n, head, member) \
>> for (pos = list_entry((head)->prev, typeof(*pos), member), \
>>   n = list_entry(pos->member.prev, typeof(*pos), member); \
>>   ^^^^^^^^^^^
>>   &pos->member != (head);   \
>>   ^^^^^^^^^^^
>>   pos = n, n = list_entry(n->member.prev, typeof(*n), member))
>>
>
> Hmm.. We used to have a for loop with !list_empty() as a termination condition
> and VM_BUG_ON(!pc) is a spill over. With the new loop, VM_BUG_ON(!pc) does not
> make sense.
>
>

I see, and I'll post a new patch to just remove it.
_____

Subject: [PATCH] memcgroup: remove a useless VM_BUG_ON()

Posted by Li Zefan on Mon, 18 Feb 2008 07:34:48 GMT

Remove this VM_BUG_ON(), as Balbir stated:

We used to have a for loop with !list_empty() as a termination condition
and VM_BUG_ON(!pc) is a spill over. With the new loop, VM_BUG_ON(!pc) does not
make sense.

Signed-off-by: Li Zefan <lizf@cn.fujitsu.com>
---
 mm/memcontrol.c | 1 -
 1 files changed, 0 insertions(+), 1 deletions(-)

```
diff --git a/mm/memcontrol.c b/mm/memcontrol.c
index 718acf8..631002d 100644
--- a/mm/memcontrol.c
+++ b/mm/memcontrol.c
@@ -534,7 +534,6 @@ unsigned long mem_cgroup_isolate_pages(unsigned long nr_to_scan,
  if (scan >= nr_to_scan)
   break;
  page = pc->page;
- VM_BUG_ON(!pc);

  if (unlikely(!PageLRU(page)))
   continue;
```
--
1.5.4.rc3

_____