Subject: Re: [PATCH] cgroup: prefer [kv]zalloc over [kv]malloc+memset in memory controller code.
Posted by Jesper Juhl on Mon, 01 Nov 2010 19:59:13 GMT
View Forum Message <> Reply to Message

On Mon, 1 Nov 2010, Johannes Weiner wrote:

> On Mon, Nov 01, 2010 at 08:40:56PM +0100, Jesper Juhl wrote:
> > Hi (please CC me on replies),
> >
> >
> > Apologies to those who receive this multiple times. I screwed up the To:
> > field in my original mail :-(
> >
> >
> > In mem_cgroup_alloc() we currently do either kmalloc() or vmalloc() then
> > followed by memset() to zero the memory. This can be more efficiently
> > achieved by using kzalloc() and vzalloc().
> >
> >
> > Signed-off-by: Jesper Juhl <jj@chaosbits.net>
>
> Looks good to me, but there is also the memset after kmalloc in
> alloc_mem_cgroup_per_zone_info().

Dang, I missed that one. Thanks for pointing it out.

Hmm, I'm wondering if we should perhaps add kzalloc_node()/vzalloc_node()
just like kzalloc() and vzalloc()..

> Can you switch that over as well in
> this patch? You can pass __GFP_ZERO to kmalloc_node() for zeroing.
>

Sure thing.

Signed-off-by: Jesper Juhl <jj@chaosbits.net>
---
 memcontrol.c | 9 +++------
 1 file changed, 3 insertions(+), 6 deletions(-)

diff --git a/mm/memcontrol.c b/mm/memcontrol.c
index 9a99cfa..bc32ffe 100644
--- a/mm/memcontrol.c
+++ b/mm/memcontrol.c
@@ -4169,13 +4169,11 @@ static int alloc_mem_cgroup_per_zone_info(struct mem_cgroup

```
*mem, int node)
   */
  if (!node_state(node, N_NORMAL_MEMORY))
   tmp = -1;
- pn = kmalloc_node(sizeof(*pn), GFP_KERNEL, tmp);
+ pn = kmalloc_node(sizeof(*pn), GFP_KERNEL|__GFP_ZERO, tmp);
  if (!pn)
   return 1;

  mem->info.nodeinfo[node] = pn;
- memset(pn, 0, sizeof(*pn));
-
  for (zone = 0; zone < MAX_NR_ZONES; zone++) {
   mz = &pn->zoneinfo[zone];
   for_each_lru(l)
@@ -4199,14 +4197,13 @@ static struct mem_cgroup *mem_cgroup_alloc(void)

  /* Can be very big if MAX_NUMNODES is very big */
  if (size < PAGE_SIZE)
-  mem = kmalloc(size, GFP_KERNEL);
+  mem = kzalloc(size, GFP_KERNEL);
  else
-  mem = vmalloc(size);
+  mem = vzalloc(size);

  if (!mem)
   return NULL;

- memset(mem, 0, size);
  mem->stat = alloc_percpu(struct mem_cgroup_stat_cpu);
  if (!mem->stat) {
   if (size < PAGE_SIZE)
```

--
Jesper Juhl <jj@chaosbits.net> http://www.chaosbits.net/
Plain text mails only, please http://www.expita.com/nomime.html
Don't top-post http://www.catb.org/~esr/jargon/html/T/top-post.html

_____
Containers mailing list
Containers@lists.linux-foundation.org
https://lists.linux-foundation.org/mailman/listinfo/containe rs

---

Subject: Re: [PATCH] cgroup: prefer [kv]zalloc over [kv]malloc+memset in memory controller code.

On Mon, Nov 01, 2010 at 08:40:56PM +0100, Jesper Juhl wrote:
> Hi (please CC me on replies),
>
>
> Apologies to those who receive this multiple times. I screwed up the To:
> field in my original mail :-(
>
>
> In mem_cgroup_alloc() we currently do either kmalloc() or vmalloc() then
> followed by memset() to zero the memory. This can be more efficiently
> achieved by using kzalloc() and vzalloc().
>
>
> Signed-off-by: Jesper Juhl <jj@chaosbits.net>

Looks good to me, but there is also the memset after kmalloc in
alloc_mem_cgroup_per_zone_info(). Can you switch that over as well in
this patch? You can pass __GFP_ZERO to kmalloc_node() for zeroing.

Thanks!

_____
Containers mailing list
Containers@lists.linux-foundation.org
https://lists.linux-foundation.org/mailman/listinfo/containe rs