Subject: Re: [PATCH 02/11] memcg: Reclaim when more than one page needed.
Posted by Glauber Costa on Tue, 26 Jun 2012 07:12:22 GMT
View Forum Message <> Reply to Message

>
>> + * retries
>> + */
>> +#define NR_PAGES_TO_RETRY 2
>> +
>
> Should be 1 << PAGE_ALLOC_COSTLY_ORDER? Where does this number come from?
> The changelog doesn't specify.

Hocko complained about that, and I changed. Where the number comes from,
is stated in the comments: it is a number small enough to have high
changes of had been freed by the previous reclaim, and yet around the
number of pages of a kernel allocation.

Of course there are allocations for nr_pages > 2. But 2 will already
service the stack most of the time, and most of the slab caches.

>> static int mem_cgroup_do_charge(struct mem_cgroup *memcg, gfp_t gfp_mask,
>> - unsigned int nr_pages, bool oom_check)
>> + unsigned int nr_pages, unsigned int min_pages,
>> + bool oom_check)
>> {
>> unsigned long csize = nr_pages * PAGE_SIZE;
>> struct mem_cgroup *mem_over_limit;
>> @@ -2182,18 +2190,18 @@ static int mem_cgroup_do_charge(struct mem_cgroup *memcg,
gfp_t gfp_mask,
>> } else
>> mem_over_limit = mem_cgroup_from_res_counter(fail_res, res);
>> /*
>> - * nr_pages can be either a huge page (HPAGE_PMD_NR), a batch
>> - * of regular pages (CHARGE_BATCH), or a single regular page (1).
>> - *
>> * Never reclaim on behalf of optional batching, retry with a
>> * single page instead.
>> */
>> - if (nr_pages == CHARGE_BATCH)
>> + if (nr_pages > min_pages)
>> return CHARGE_RETRY;
>>
>> if (!(gfp_mask & __GFP_WAIT))
>> return CHARGE_WOULDBLOCK;
>>
>> + if (gfp_mask & __GFP_NORETRY)
>> + return CHARGE_NOMEM;

```
>> +
>>    ret = mem_cgroup_reclaim(mem_over_limit, gfp_mask, flags);
>>    if (mem_cgroup_margin(mem_over_limit) >= nr_pages)
>>     return CHARGE_RETRY;
>> @@ -2206,7 +2214,7 @@ static int mem_cgroup_do_charge(struct mem_cgroup *memcg,
gfp_t gfp_mask,
>>     * unlikely to succeed so close to the limit, and we fall back
>>     * to regular pages anyway in case of failure.
>>     */
>> - if (nr_pages == 1 && ret)
>> + if (nr_pages <= NR_PAGES_TO_RETRY && ret)
>>     return CHARGE_RETRY;
>>
>>    /*
>> @@ -2341,7 +2349,8 @@ again:
>>      nr_oom_retries = MEM_CGROUP_RECLAIM_RETRIES;
>>     }
>>
>> -  ret = mem_cgroup_do_charge(memcg, gfp_mask, batch, oom_check);
>> +  ret = mem_cgroup_do_charge(memcg, gfp_mask, batch, nr_pages,
>> +      oom_check);
>>     switch (ret) {
>>     case CHARGE_OK:
>>      break;
```