Subject: Re: [PATCH v2 03/11] memcg: change defines to an enum
Posted by Michal Hocko on Fri, 10 Aug 2012 15:43:53 GMT
View Forum Message <> Reply to Message

On Thu 09-08-12 17:01:11, Glauber Costa wrote:
> This is just a cleanup patch for clarity of expression. In earlier
> submissions, people asked it to be in a separate patch, so here it is.
>
> [v2: use named enum as type throughout the file as well]
>
> Signed-off-by: Glauber Costa <glommer@parallels.com>
> CC: Michal Hocko <mhocko@suse.cz>
> CC: Johannes Weiner <hannes@cmpxchg.org>
> Acked-by: Kamezawa Hiroyuki <kamezawa.hiroyu@jp.fujitsu.com>

Acked-by: Michal Hocko <mhocko@suse.cz>

> ---
> mm/memcontrol.c | 26 ++++++++++++++++++----------
> 1 file changed, 16 insertions(+), 10 deletions(-)
>
> diff --git a/mm/memcontrol.c b/mm/memcontrol.c
> index 2cef99a..b0e29f4 100644
> --- a/mm/memcontrol.c
> +++ b/mm/memcontrol.c
> @@ -393,9 +393,12 @@ enum charge_type {
> };
>
> /* for encoding cft->private value on file */
> -#define _MEM   (0)
> -#define _MEMSWAP (1)
> -#define _OOM_TYPE (2)
> +enum res_type {
> + _MEM,
> + _MEMSWAP,
> + _OOM_TYPE,
> +};
> +
> #define MEMFILE_PRIVATE(x, val) ((x) << 16 | (val))
> #define MEMFILE_TYPE(val) ((val) >> 16 & 0xffff)
> #define MEMFILE_ATTR(val) ((val) & 0xffff)
> @@ -3983,7 +3986,8 @@ static ssize_t mem_cgroup_read(struct cgroup *cont, struct cftype
*cft,
>   struct mem_cgroup *memcg = mem_cgroup_from_cont(cont);
>   char str[64];
>   u64 val;
> - int type, name, len;
> + int name, len;

> + enum res_type type;
>
>   type = MEMFILE_TYPE(cft->private);
>   name = MEMFILE_ATTR(cft->private);
> @@ -4019,7 +4023,8 @@ static int mem_cgroup_write(struct cgroup *cont, struct cftype *cft,
>   const char *buffer)
> {
>   struct mem_cgroup *memcg = mem_cgroup_from_cont(cont);
> - int type, name;
> + enum res_type type;
> + int name;
>   unsigned long long val;
>   int ret;
>
> @@ -4095,7 +4100,8 @@ out:
> static int mem_cgroup_reset(struct cgroup *cont, unsigned int event)
> {
>   struct mem_cgroup *memcg = mem_cgroup_from_cont(cont);
> - int type, name;
> + int name;
> + enum res_type type;
>
>   type = MEMFILE_TYPE(event);
>   name = MEMFILE_ATTR(event);
> @@ -4423,7 +4429,7 @@ static int mem_cgroup_usage_register_event(struct cgroup *cgrp,
>   struct mem_cgroup *memcg = mem_cgroup_from_cont(cgrp);
>   struct mem_cgroup_thresholds *thresholds;
>   struct mem_cgroup_threshold_ary *new;
> - int type = MEMFILE_TYPE(cft->private);
> + enum res_type type = MEMFILE_TYPE(cft->private);
>   u64 threshold, usage;
>   int i, size, ret;
>
> @@ -4506,7 +4512,7 @@ static void mem_cgroup_usage_unregister_event(struct cgroup
*cgrp,
>   struct mem_cgroup *memcg = mem_cgroup_from_cont(cgrp);
>   struct mem_cgroup_thresholds *thresholds;
>   struct mem_cgroup_threshold_ary *new;
> - int type = MEMFILE_TYPE(cft->private);
> + enum res_type type = MEMFILE_TYPE(cft->private);
>   u64 usage;
>   int i, j, size;
>
> @@ -4584,7 +4590,7 @@ static int mem_cgroup_oom_register_event(struct cgroup *cgrp,
> {
>   struct mem_cgroup *memcg = mem_cgroup_from_cont(cgrp);
>   struct mem_cgroup_eventfd_list *event;
> - int type = MEMFILE_TYPE(cft->private);

> + enum res_type type = MEMFILE_TYPE(cft->private);
>
>   BUG_ON(type != _OOM_TYPE);
>   event = kmalloc(sizeof(*event), GFP_KERNEL);
> @@ -4609,7 +4615,7 @@ static void mem_cgroup_oom_unregister_event(struct cgroup *cgrp,
> {
>   struct mem_cgroup *memcg = mem_cgroup_from_cont(cgrp);
>   struct mem_cgroup_eventfd_list *ev, *tmp;
> - int type = MEMFILE_TYPE(cft->private);
> + enum res_type type = MEMFILE_TYPE(cft->private);
>
>   BUG_ON(type != _OOM_TYPE);
>
> --
> 1.7.11.2
>
> --
> To unsubscribe from this list: send the line "unsubscribe cgroups" in
> the body of a message to majordomo@vger.kernel.org
> More majordomo info at http://vger.kernel.org/majordomo-info.html

--
Michal Hocko
SUSE Labs