
Subject: [PATCH v5 03/14] memcg: change defines to an enum
Posted by [Glauber Costa](#) on Tue, 16 Oct 2012 10:16:40 GMT
[View Forum Message](#) <> [Reply to Message](#)

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>
Acked-by: Kamezawa Hiroyuki <kamezawa.hiroyu@jp.fujitsu.com>
Acked-by: Michal Hocko <mhocko@suse.cz>
Acked-by: Johannes Weiner <hannes@cmpxchg.org>
CC: Tejun Heo <tj@kernel.org>

```
---
mm/memcontrol.c | 26 ++++++-----
1 file changed, 16 insertions(+), 10 deletions(-)

diff --git a/mm/memcontrol.c b/mm/memcontrol.c
index 7a9652a..71d259e 100644
--- a/mm/memcontrol.c
+++ b/mm/memcontrol.c
@@ -386,9 +386,12 @@ 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)
@@ -3915,7 +3918,8 @@ 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);
```

```

@@ -3951,7 +3955,8 @@ 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;

@@ -4027,7 +4032,8 @@ 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);
@@ -4363,7 +4369,7 @@ 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;

@@ -4446,7 +4452,7 @@ 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;

@@ -4524,7 +4530,7 @@ 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);
@@ -4549,7 +4555,7 @@ 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);
```

```
--
```

```
1.7.11.7
```
