
Subject: [PATCH v5 09/10] Display current tcp memory allocation in kmem cgroup
Posted by [Glauber Costa](#) on Mon, 07 Nov 2011 15:26:34 GMT
[View Forum Message](#) <> [Reply to Message](#)

This patch introduces kmem.tcp.max_usage_in_bytes file, living in the kmem_cgroup filesystem. The root cgroup will display a value equal to RESOURCE_MAX. This is to avoid introducing any locking schemes in the network paths when cgroups are not being actively used.

All others, will see the maximum memory ever used by this cgroup.

Signed-off-by: Glauber Costa <glommer@parallels.com>
CC: David S. Miller <davem@davemloft.net>
CC: Hiroyouki Kamezawa <kamezawa.hiroyu@jp.fujitsu.com>
CC: Eric W. Biederman <ebiederm@xmission.com>

```
---
mm/memcontrol.c | 10 ++++++++
1 files changed, 10 insertions(+), 0 deletions(-)

diff --git a/mm/memcontrol.c b/mm/memcontrol.c
index 9394224..b532f91 100644
--- a/mm/memcontrol.c
+++ b/mm/memcontrol.c
@@ -540,6 +540,12 @@ static struct cftype tcp_files[] = {
    .trigger = mem_cgroup_reset,
    .read_u64 = mem_cgroup_read,
},
+ {
+ .name = "kmem.tcp.max_usage_in_bytes",
+ .private = MEMFILE_PRIVATE(_KMEM_TCP, RES_MAX_USAGE),
+ .trigger = mem_cgroup_reset,
+ .read_u64 = mem_cgroup_read,
+ },
};

static void tcp_create_cgroup(struct mem_cgroup *cg, struct cgroup_subsys *ss)
@@ -4254,6 +4260,10 @@ static int mem_cgroup_reset(struct cgroup *cont, unsigned int event)
    case RES_MAX_USAGE:
        if (type == _MEM)
            res_counter_reset_max(&mem->res);
+ #if defined(CONFIG_CGROUP_MEM_RES_CTLR_KMEM) && defined(CONFIG_INET)
+ else if (type == _KMEM_TCP)
+     res_counter_reset_max(&mem->tcp.tcp_memory_allocated);
+ #endif
    else
        res_counter_reset_max(&mem->memsw);
    break;
--
```

1.7.6.4
