
Subject: [PATCH v7 09/10] Display maximum tcp memory allocation in kmem cgroup

Posted by [Glauber Costa](#) on Tue, 29 Nov 2011 23:57:00 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>

```
net/ipv4/tcp_memcontrol.c | 12 ++++++++
1 files changed, 11 insertions(+), 1 deletions(-)
```

```
diff --git a/net/ipv4/tcp_memcontrol.c b/net/ipv4/tcp_memcontrol.c
```

```
index f7350fa1..6ec90de 100644
```

```
--- a/net/ipv4/tcp_memcontrol.c
```

```
+++ b/net/ipv4/tcp_memcontrol.c
```

```
@@ -29,6 +29,12 @@ static struct cftype tcp_files[] = {
```

```
    .trigger = tcp_cgroup_reset,
```

```
    .read_u64 = tcp_cgroup_read,
```

```
    },
```

```
+ {
```

```
+ .name = "kmem.tcp.max_usage_in_bytes",
```

```
+ .private = RES_MAX_USAGE,
```

```
+ .trigger = tcp_cgroup_reset,
```

```
+ .read_u64 = tcp_cgroup_read,
```

```
+ },
```

```
};
```

```
static inline struct tcp_memcontrol *tcp_from_cgproto(struct cg_proto *cg_proto)
```

```
@@ -201,7 +207,8 @@ static u64 tcp_cgroup_read(struct cgroup *cont, struct cftype *cft)
```

```
    val = tcp_read_usage(memcg);
```

```
    break;
```

```
    case RES_FAILCNT:
```

```
- val = tcp_read_stat(memcg, RES_FAILCNT, 0);
```

```
+ case RES_MAX_USAGE:
```

```
+ val = tcp_read_stat(memcg, cft->private, 0);
```

```
    break;
```

```
    default:
```

```
        BUG();
```

```
@@ -222,6 +229,9 @@ static int tcp_cgroup_reset(struct cgroup *cont, unsigned int event)
```

```
tcp = tcp_from_cgproto(cg_proto);
```

```
    switch (event) {  
+ case RES_MAX_USAGE:  
+   res_counter_reset_max(&tcp->tcp_memory_allocated);  
+   break;  
    case RES_FAILCNT:  
      res_counter_reset_failcnt(&tcp->tcp_memory_allocated);  
      break;  
--
```

```
1.7.6.4
```
