Subject: Re: [PATCH v5 6/8] tcp buffer limitation: per-cgroup limit
Posted by Eric Dumazet on Tue, 04 Oct 2011 12:48:55 GMT
View Forum Message <> Reply to Message

Le mardi 04 octobre 2011 à 16:17 +0400, Glauber Costa a écrit :
> This patch uses the "tcp_max_mem" field of the kmem_cgroup to
> effectively control the amount of kernel memory pinned by a cgroup.
>
> We have to make sure that none of the memory pressure thresholds
> specified in the namespace are bigger than the current 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>
> ---

> --- a/include/net/tcp.h
> +++ b/include/net/tcp.h
> @@ -256,6 +256,7 @@ extern int sysctl_tcp_thin_dupack;
>  struct mem_cgroup;
>  struct tcp_memcontrol {
>   /* per-cgroup tcp memory pressure knobs */
> + int tcp_max_memory;
>   atomic_long_t tcp_memory_allocated;
>   struct percpu_counter tcp_sockets_allocated;
>   /* those two are read-mostly, leave them at the end */
> diff --git a/mm/memcontrol.c b/mm/memcontrol.c

So tcp_max_memory is an "int".

> +static u64 tcp_read_limit(struct cgroup *cgrp, struct cftype *cft)
> +{
> + struct mem_cgroup *memcg = mem_cgroup_from_cont(cgrp);
> + return memcg->tcp.tcp_max_memory << PAGE_SHIFT;
> +}

1) Typical integer overflow here.

You need :

return ((u64)memcg->tcp.tcp_max_memory) << PAGE_SHIFT;

2) Could you add const qualifiers when possible to your pointers ?