Subject: Re: [PATCH v6 01/10] Basic kernel memory functionality for the Memory Controller
Posted by Glauber Costa on Mon, 28 Nov 2011 11:03:09 GMT
View Forum Message <> Reply to Message

On 11/28/2011 12:24 AM, KAMEZAWA Hiroyuki wrote:
> On Fri, 25 Nov 2011 15:38:07 -0200
> Glauber Costa<glommer@parallels.com> wrote:
>
>> This patch lays down the foundation for the kernel memory component
>> of the Memory Controller.
>>
>> As of today, I am only laying down the following files:
>>
>> * memory.independent_kmem_limit
>> * memory.kmem.limit_in_bytes (currently ignored)
>> * memory.kmem.usage_in_bytes (always zero)
>>
>> Signed-off-by: Glauber Costa<glommer@parallels.com>
>> Reviewed-by: Kirill A. Shutemov<kirill@shutemov.name>
>> CC: Paul Menage<paul@paulmenage.org>
>> CC: Greg Thelen<gthelen@google.com>
>> ---
>> Documentation/cgroups/memory.txt | 36 ++++++++++++-
>> init/Kconfig | 14 +++++
>> mm/memcontrol.c | 107 ++--
>> 3 files changed, 150 insertions(+), 7 deletions(-)
>>
>> diff --git a/Documentation/cgroups/memory.txt b/Documentation/cgroups/memory.txt
>> index 06eb6d9..bf00cd2 100644
>> --- a/Documentation/cgroups/memory.txt
>> +++ b/Documentation/cgroups/memory.txt
>> @@ -44,8 +44,9 @@ Features:
>>   - oom-killer disable knob and oom-notifier
>>   - Root cgroup has no limit controls.
>>
>> - Kernel memory and Hugepages are not under control yet. We just manage
>> - pages on LRU. To add more controls, we have to take care of performance.
>> + Hugepages is not under control yet. We just manage pages on LRU. To add more
>> + controls, we have to take care of performance. Kernel memory support is work
>> + in progress, and the current version provides basically functionality.
>>
>> Brief summary of control files.
>>
>> @@ -56,8 +57,11 @@ Brief summary of control files.
>>   (See 5.5 for details)
>>   memory.memsw.usage_in_bytes # show current res_counter usage for memory+Swap
>>   (See 5.5 for details)

```
>> + memory.kmem.usage_in_bytes  # show current res_counter usage for kmem only.
>> +     (See 2.7 for details)
>>    memory.limit_in_bytes   # set/show limit of memory usage
>>    memory.memsw.limit_in_bytes  # set/show limit of memory+Swap usage
>> + memory.kmem.limit_in_bytes  # if allowed, set/show limit of kernel memory
>>    memory.failcnt    # show the number of memory usage hits limits
>>    memory.memsw.failcnt   # show the number of memory+Swap hits limits
>>    memory.max_usage_in_bytes  # show max memory usage recorded
>> @@ -72,6 +76,9 @@ Brief summary of control files.
>>    memory.oom_control   # set/show oom controls.
>>    memory.numa_stat   # show the number of memory usage per numa node
>>
>> + memory.independent_kmem_limit  # select whether or not kernel memory limits are
>> +       independent of user limits
>> +
>>   1. History
>>
>>   The memory controller has a long history. A request for comments for the memory
>> @@ -255,6 +262,31 @@ When oom event notifier is registered, event will be delivered.
>>     per-zone-per-cgroup LRU (cgroup's private LRU) is just guarded by
>>     zone->lru_lock, it has no lock of its own.
>>
>> +2.7 Kernel Memory Extension (CONFIG_CGROUP_MEM_RES_CTLR_KMEM)
>> +
>> + With the Kernel memory extension, the Memory Controller is able to limit
>> +the amount of kernel memory used by the system. Kernel memory is fundamentally
>> +different than user memory, since it can't be swapped out, which makes it
>> +possible to DoS the system by consuming too much of this precious resource.
>> +Kernel memory limits are not imposed for the root cgroup.
>> +
>> +Memory limits as specified by the standard Memory Controller may or may not
>> +take kernel memory into consideration. This is achieved through the file
>> +memory.independent_kmem_limit. A Value different than 0 will allow for kernel
>> +memory to be controlled separately.
>> +
>> +When kernel memory limits are not independent, the limit values set in
>> +memory.kmem files are ignored.
>> +
>> +Currently no soft limit is implemented for kernel memory. It is future work
>> +to trigger slab reclaim when those limits are reached.
>> +
>> +CAUTION: As of this writing, the kmem extention may prevent tasks from moving
>> +among cgroups. If a task has kmem accounting in a cgroup, the task cannot be
>> +moved until the kmem resource is released. Also, until the resource is fully
>> +released, the cgroup cannot be destroyed. So, please consider your use cases
>> +and set kmem extention config option carefully.
>> +
>
```

> This seems that memcg 'has' kernel memory limiting feature for all kinds of kmem..
> Could you add a list of "currently controled kmems" section ?
> And update the list in later patch ?
>
> Thanks,
> -Kame
>
>
Hi Kame,

Thanks for your review.
Since none of your comments are blockers, I'd prefer to send follow up patches if you don't mind - assuming Dave won't have any restrictions himself that would prevent him from picking this series. If I have to resend it anyway, I'll be more than happy to address them all in my next submission

---