

---

Subject: Re: [RFC][PATCH 0/3] Kernel memory accounting container (v2)

Posted by [Pavel Emelianov](#) on Thu, 13 Sep 2007 11:33:07 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

KAMEZAWA Hiroyuki wrote:

> On Thu, 13 Sep 2007 13:11:35 +0400

> Pavel Emelyanov <xemul@openvz.org> wrote:

>

>> First of all - why do we need this kind of control. The major  
>> "pros" is that kernel memory control protects the system  
>> from DoS attacks by processes that live in container. As our  
>> experience shows many exploits simply do not work in the  
>> container with limited kernel memory.

>>

>> I can split the kernel memory container into 4 parts:

>>

>> 1. kmalloc-ed objects control

>> 2. vmalloc-ed objects control

>> 3. buddy allocated pages control

>> 4. kmem\_cache\_alloc-ed objects control

>>

> <snip>

>> To play with it, one need to mount the container file system

>> with -o kmem and then mark some caches as accountable via

>> /sys/slab/<cache\_name>/cache\_account.

>>

> Hmm, how can we know "How many kmem will we need ?" in precise per-object

> style ? Is this useful ?

You can start with unlimited container and check how many kernel memory your applications use normally and set the limit to 120% of this.

You may also set this to some reasonable value like 50% of normal zone to protect your system from a fork bomb or similar.

This is the same question as "how many user pages will my container consume". The answer is - find it out experimentally or ask for someone who has already done so.

> Following kind of limitation of user friendly params is bad ?

>

> - # of file handles

> - # of tasks

> - # of sockets/ connections / packets

> - # of posix IPC related things

> - and other sources of DoS.

These are not enough and none of them are reasonable. E.g. the struct vm\_area\_struct objects are allocated for many mmap() calls, but how to find it out how many of them you will require.

However some controllers will be done as well.

> Thanks,  
> -Kame  
>  
>  
>

---

Containers mailing list  
Containers@lists.linux-foundation.org  
<https://lists.linux-foundation.org/mailman/listinfo/containers>

---