Subject: Re: [PATCH 2/4] The character devices layer changes
Posted by Pavel Emelianov on Tue, 15 Jan 2008 08:05:22 GMT
View Forum Message <> Reply to Message

Serge E. Hallyn wrote:
> Quoting Pavel Emelyanov (xemul@openvz.org):
>> These changes include the API for the control group
>> to map/remap/unmap the devices with their permissions
>> and one important thing.
>>
>> The fact is that the struct cdev is cached in the inode
>> for faster access, so once we looked one up we go through
>> the fast path and omit the kobj_lookup() call. This is no
>> longer good when we restrict the access to cdevs.
>>
>> To address this issue, I store the last_perm and last(_map)
>> fields on the struct cdev (and protect them with the cdev_lock)
>> and force the re-lookup in the kobj mappings if needed.
>>
>> I know, this might be slow, but I have two points for it:
>> 1. The re-lookup happens on open() only which is not
>>   a fast-path. Besides, this is so for block layer and
>>   nobody complains;
>> 2. On a well-isolated setup, when each container has its
>>   own filesystem this is no longer a problem - each
>>   cgroup will cache the cdev on its inode and work good.
>
> What about simply returning -EPERM when open()ing a cdev
> with ->map!=task_cdev_map(current)?

In this case it will HAVE to setup isolated filesystem for
each cgroup. I thought that this flexibility doesn't hurt.

> Shouldn't be a problem for ttys, since the container init
> already has the tty open, right?

Yup, but this is not the case for /dev/null or /dev/zero.

> Otherwise, the patchset looks good to me. Want to look
> through this one a little more (i think that'd be easier
> with the -EPERM approach) and scrutinize patch 4, but
> overall it makes sense.

OK, thanks.

> If I understand right, we're taking 14k per cgroup for
> kobjmaps? Do we consider that a problem?

14k? I allocate the struct kobj_map which is only 256 pointers
(i.e. - 2K) and the struct probe that is 32 bytes. I.e. 4k
or a single page. I think this is OK.

> thanks,
> -serge
>

[snip]

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