Subject: Re: [RFC][PATCH] Devices visibility container
Posted by Pavel Emelianov on Mon, 24 Sep 2007 11:47:26 GMT
View Forum Message <> Reply to Message

Cedric Le Goater wrote:
> Pavel Emelyanov wrote:
>> Hi.
>>
>> At KS we have pointed out the need in some container, that allows
>> to limit the visibility of some devices to task within it. I.e.
>> allow for /dev/null, /dev/zero etc, but disable (by default) some
>> IDE devices or SCSI discs and so on.
>>
>> Here's the beta of the container. Currently this only allows to
>> hide the _character_ devices only from the living tasks. To play
>> with it you just create the container like this
>>
>> # mount -t container none /cont/devs -o devices
>> # mkdir /cont/devs/0
>>
>> it will have two specific files
>>
>> # ls /cont/devs
>> devices.block devices.char notify_on_release releasable release_agent tasks
>>
>> then move a task into it
>>
>> # /bin/echo -n $$ > /cont/devs/0/tasks
>>
>> after this you won't be able to read from even /dev/zero
>>
>> # hexdump /dev/zero
>> hexdump: /dev/zero: No such device or address
>> hexdump: /dev/zero: Bad file descriptor
>>
>> meanwhile from another ssh session you will. You may allow access
>> to /dev/zero like this
>>
>> # /bin/echo -n '+1:5' > /cont/devs/0/devices.char
>>
>> More generally, the '+<major>:<minor>' string grants access to
>> some device, and '-<major>:<minor>' disables one.
>>
>> The TODO list now looks like this:
>> * add the block devices support :) don't know how to make it yet;
>
> I think the mapping is done trough a pseudo-fs for the block devices.
> It probably means that we will have to mount it multiple times to

> handle the isolation.

Maybe. I looked over the block layer and found that character one
was simpler to start with.

>> * make /proc/devices show relevant info depending on who is
>>   reading it. currently even if major 1 is disabled for task,
>>   it will be listed in this file;
>> * make it possible to enable/disable not just individual major:minor
>>   pair, but something more flexible, e.g. major:* for all minors
>>   for given major or major:m1-m2 for minor range, etc;
>
> yep.

:)

>> * add the ability to restrict the read/write permissions for a
>>   container. currently one may just control the visible-invisible
>>   state for a device in a container, but maybe just readable or
>>   just writable would be better.
>>
>> This patch is minimally tested, because I just want to know your
>> opinion on whether it worths developing the container in such a way or not.
>
> it looks simple enough to me.

Well, OK. Then I will go on developing this one.

> I'm wondering how many control groups subsystems we will need
> to make The *Container* and if it's not worth just merging
> them in a big unified one.

Ha ha, so am I :)

> Thanks !
>
> C.

Thanks,
Pavel