Subject: Re: [patch 1/2] [RFC] Simple tamper-proof device filesystem.
Posted by Oren Laadan on Fri, 21 Dec 2007 01:46:38 GMT
View Forum Message <> Reply to Message

Pavel Emelyanov wrote:
> Oren Laadan wrote:
>> Serge E. Hallyn wrote:
>>> Quoting Pavel Emelyanov (xemul@openvz.org):
>>>> Oren Laadan wrote:
>>>>> Serge E. Hallyn wrote:
>>>>>> Quoting Oren Laadan (orenl@cs.columbia.edu):
>>>>>>> I hate to bring this again, but what if the admin in the container
>>>>>>> mounts an external file system (eg. nfs, usb, loop mount from a file,
>>>>>>> or via fuse), and that file system already has a device that we would
>>>>>>> like to ban inside that container ?
>>>>>> Miklos' user mount patches enforced that if !capable(CAP_MKNOD),
>>>>>> then mnt->mnt_flags |= MNT_NODEV. So that's no problem.
>>>>> Yes, that works to disallow all device files from a mounted file system.
>>>>>
>>>>> But it's a black and white thing: either they are all banned or allowed;
>>>>> you can't have some devices allowed and others not, depending on type
>>>>> A scenario where this may be useful is, for instance, if we some apps in
>>>>> the container to execute withing a pre-made chroot (sub)tree within that
>>>>> container.
>>>>>
>>>> But that's been pulled out of -mm! ? Crap.
>>>>
>>>>> Since anyway we will have to keep a white- (or black-) list of devices
>>>>> that are permitted in a container, and that list may change even change
>>>>> per container -- why not enforce the access control at the VFS layer ?
>>>>> It's safer in the long run.
>>>> By that you mean more along the lines of Pavel's patch than my whitelist
>>>> LSM, or you actually mean Tetsuo's filesystem (i assume you don't mean that
>>>> by 'vfs layer' :), or something different entirely?
>>> :)
>>>
>>> By 'vfs' I mean at open() time, and not at mount(), or mknod() time.
>>> Either yours or Pavel's; I tend to prefer not to use LSM as it may
>>> collide with future security modules.
>> Oren, AFAIS you've seen my patches for device access controller, right?
> If you mean this one:
> http://openvz.org/pipermail/devel/2007-September/007647.html
> then ack :)
>
> Great! Thanks.
>
>> Maybe we can revisit the issue then and try to come to agreement on what
>> kind of model and implementation we all want?

>>> That would be great, Pavel. I do prefer your solution over my LSM, so
>>> if we can get an elegant block device control right in the vfs code that
>>> would be my preference.
>> I concur.
>>
>> So it seems to me that we are all in favor of the model where open()
>> of a device will consult a black/white-list. Also, we are all in favor
>> of a non-LSM implementation, Pavel's code being a good example.
>
> Thank you, Oren and Serge! I will revisit this issue then, but
> I have a vacation the next week and, after this, we have a New
> Year and Christmas holidays in Russia. So I will be able to go
> on with it only after the 7th January :(Hope this is OK for you.
>
> Besides, Andrew told that he would pay little attention to new
> features till the 2.6.24 release, so I'm afraid we won't have this
> even in -mm in the nearest months :(

Sounds great ! (as for the delay, it wasn't the highest priority issue
to begin with, so no worries).

Ah.. coincidentally they are celebrated here, too, on the same time :D
Merry Christmas and Happy New Year !

Oren.

>
> Thanks,
> Pavel
>
>> Oren.
>>
>>> The only thing that makes me keep wanting to go back to an LSM is the
>>> fact that the code defining the whitelist seems out of place in the vfs.
>>> But I guess that's actually separated into a modular cgroup, with the
>>> actual enforcement built in at the vfs. So that's really the best
>>> solution.
>>>
>>> -serge
>

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