Subject: Re: [ckrm-tech] [PATCH 0/2] resource control file system - aka containers on top of nsproxy!
Posted by Paul Jackson on Fri, 09 Mar 2007 22:06:03 GMT
View Forum Message <> Reply to Message

> the emphasis here is on 'from inside' which basically
> boils down to the following:
>
> if you create a 'resource container' to limit the
> usage of a set of resources for the processes
> belonging to this container, it would be kind of
> defeating the purpose, if you'd allow the processes
> to manipulate their limits, no?

Wrong - this is not the only way.

For instance in cpusets, -any- task in the system, regardless of what cpuset it is currently assigned to, might be able to manipulate -any- cpuset in the system.

Yes -- some sufficient mechanism is required to keep tasks from escalating their resources or capabilities beyond an allowed point.

But that mechanism might not be strictly based on position in some hierarchy.

In the case of cpusets, it is based on the permissions on files in the cpuset file system (normally mounted at /dev/cpuset), versus the current priviledges and capabilities of the task.

A root priviledged task in the smallest leaf node cpuset can manipulate every cpuset in the system. This is an ordinary and common occurrence.

I say again, as you seem to be skipping over this detail, one advantage of basing an API on a file system is the usefulness of the file system permission model (the -rwxrwxrwx permissions and the uid/gid owners on each file and directory node).

--

I won't rest till it's the best ...
Programmer, Linux Scalability
Paul Jackson <pj@sgi.com> 1.925.600.0401

Containers mailing list
Containers@lists.osdl.org
https://lists.osdl.org/mailman/listinfo/containers