

Sam Vilain <sam@vilain.net> writes:

> On Wed, 2006-03-29 at 07:47 -0600, Serge E. Hallyn wrote:
>> Alas, the spacing on the picture didn't quite work out :) I think that
>> by nested containers, you mean overlapping nested containers. In your
>> example, how are you suggesting that cont1 refers to items in
>> container1.1.2's shmem? I assume, given your previous posts on openvz,
>> that you want every shmem id in all namespaces "nested" under cont1 to
>> be unique, and for cont1 to refer to any item in container1.1.2's
>> namespace just as it would any of cont1's own shmem?
>>
>> In that case I am not sure of the actual usefulness. Someone with
>> different use for containers (you? :) will need to justify it. For me,
>> pure isolation works just fine. Clearly it will be most useful if we
>> want fine-grained administration, from parent namespaces, of the items
>> in a child namespace.
>
> The overlapping is important if you want to pretend that the
> namespace-able resources are allowed to be specified per-process, when
> really they are specified per-family.
>
> In this way, a process family is merely a grouping of processes with
> like namespaces, and depending on which way they overlap you get the
> same behaviour as when processes only have one resource different, and
> therefore remove the overhead on fork().

I missed this subthread originally.

I think it is important that we can have containers in containers
if at all possible. This means large software collections can count
on them being present.

As for having some items inside a namespace show up in both
a parent and a child namespace I think the case is less clearly
defined. If possible that is something we want to avoid as it
complicates the implementation.

For pids I will be surprised if we can avoid it.

For most other namespaces I think we can, and it is a good thing
to avoid.

Eric
