Subject: Re: Checkpoint/restart (was Re: [PATCH 0/4] - v2 - Object creation with a specified id)
Posted by Kirill Korotaev on Wed, 23 Apr 2008 06:40:21 GMT
View Forum Message <> Reply to Message

> If the current interface is insufficient, we should first expand it in
> such a way that it can be used for checkpoint. That certainly won't
> work in all cases. fork(), for instance, doesn't take any arguments and
> is going to be awfully hard to expand. :)
>
> I'd love to hear some of your insights about how things like the current
> iptables interfaces are insufficient for checkpoint/restart.

iptables is a bad example. Luckily for checkpointing - it always had an interface
"load full state", "dump full state".

But even iptables are not working in current form for checkpointing - they can't
save/restore state of conntracks. Do you think netdev@/netfilter@ guys will be happy
to have APIs allowing to set conntracks and have all the pain related
to API stability - cause conntrack state changed a couple of times during last 2 years.

Consider more intimate kernel states like:
a. task statistics
b. task start time
c. load average
d. skb state and it's data.
e. mount tree.

If you think over, e.g. (b) is a bad thing. It was used to be accounted in jiffies, then in timespec.
(a) is another example of dataset which we can't predict. task statistics change over a time.
Why bother with such intimate data in user-space at all?
Why the hell user-space should know about it and be ABLE to modify it?

Why do we need to export ability to set IDs for some objects which none of the
operating systems do and then to have a burden to support it for application compatibility
the rest of our lifes? Do you really believe none of the applications except for checkpointing
will be using it?

My personal vision is that:
1. user space must initialize checkpointing/restore state via some system call,
   supply file descriptor from where data can be read/written to.
2. must call the syscall asking kernel to restore/save different subsytems one by one.
3. finalize cpt/restore state via the syscall
But user-space MUST NOT bother about data content. At least not about the data supplied by the kernel.
It can add additional sections if needed, e.g. about iptables state.

Having all this functionality in a signle syscall we specifically CLAIM a black box, and that no one can use this interfaces for something different from checkpoint/restore.

So I think we have to know what other maintainers think before we can go.

>> These next ids are suitable, well, only for ids which is very, very small
>> part of kernel state needed to restore group of processes.
>
> I couldn't agree more. This id setting mechanism would only be useful
> for a small subset of the things we need during a restart.
>
> -- Dave
>
> ____
> Containers mailing list
> Containers@lists.linux-foundation.org
> https://lists.linux-foundation.org/mailman/listinfo/containers
>

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