Subject: Re: [PATCH 1/4] signal: Introduce kill_pid_ns_info
Posted by Pavel Emelianov on Wed, 12 Dec 2007 16:09:03 GMT
View Forum Message <> Reply to Message

Eric W. Biederman wrote:
> Implement the basic helper function that walks all of the processes in
> a pid namespace and sends them all a signal.
>
> Both locations that could use this functions are also updated to use
> this function.
>
> I use find_ge_pid instead of for_each_process because it has a chance
> of not touching every process in the system.
>
> Signed-off-by: Eric W. Biederman <ebiederm@xmission.com>

Looks sane.

Acked-by: Pavel Emelyanov <xemul@openvz.org>

but one comment inside.

[snip]

> diff --git a/kernel/signal.c b/kernel/signal.c
> index 074905f..1eb0661 100644
> --- a/kernel/signal.c
> +++ b/kernel/signal.c
> @@ -1083,6 +1083,45 @@ int group_send_sig_info(int sig, struct siginfo *info, struct
task_struct *p)
> return ret;
> }
>
> +int __kill_pid_ns_info(int sig, struct siginfo *info, struct pid_namespace *ns)

Shouldn't it be static?

> +{
> + int retval = 0, count = 0;
> + struct task_struct *p;
> + struct pid *pid;
> + int nr;
> +
> + /* Since there isn't a pid namespace list of tasks use the closet
> + * approximation we have: find_ge_pid.
> + */
> + nr = 0;
> + while ((pid = find_ge_pid(nr + 1, ns))) {

```
> +  int err;
> +
> +  nr = pid_nr_ns(pid, ns);
> +  p = pid_task(pid, PIDTYPE_PID);
> +  if (!p || (nr <= 1) || !thread_group_leader(p) ||
> +      same_thread_group(p, current))
> +   continue;
> +
> +  err = group_send_sig_info(sig, info, p);
> +  ++count;
> +  if (err != -EPERM)
> +   retval = err;
> + }
> + return count ? retval : -ESRCH;
> +}
> +
```

[snip]

Thanks,
Pavel

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