

Serge E. Hallyn wrote:

> Quoting Cedric Le Goater (clg@fr.ibm.com):

>> with all the infos you've gathered in /proc, why don't you just kill the  
>> process ?

>>

>> The patch we have to restore pending signals in 2.6.21-mm2-lxc3 does :

>>

>> +static int pid\_set\_siginfo(mcrk\_session\_t \* s, void \*ptarg)

>> +{

>> + mcrk\_pid\_setsignal\_t arg;

>> + siginfo\_t si;

>> + int ret;

>> +

>> + if (!ptarg) {

>> + return -EINVAL;

>> + }

>> +

>> + if (copy\_from\_user(&arg, ptarg, sizeof(arg)))

>> + return -EFAULT;

>> + if (copy\_from\_user(&si, U64\_2\_PTR(arg.siginfo), sizeof(si)))

>> + return -EFAULT;

>

> Hmm, one problem with especially this second copy\_from\_user() is that  
> you are making the checkpoint image more kernel dependant.

right. we need an opaque structure to hold the siginfo data.

> Whatever approach we take both high-level and low-level, we do want to

> avoid having checkpoint images directly reflect in-kernel structures,

> right?

yes.

> That's one area where the /proc approach has an inherent advantage over

> using netlink to dump information, it avoids the temptation to just dump

> and restore straight from the kernel pointer, which would threaten to

> make restoring a checkpoint from another kernel much more dangerous.

I agree. You have to be self disciplined and define nice structures for  
all the data you want to exchange between kernel and user.

>> + if (arg.shared) {

>> + ret = kill\_proc\_info(si.si\_signo, &si, current->pid);

>> + } else {

```
>> + ret = send_sig_info(si.si_signo, &si, current);
>> + }
>> + return ret;
>> +}
>
> This part is fine with me, but assumes we take the more kernel-guided
> approach, right.
>
> And that's what I'm trying to get people to discuss :) Do we want a
> more kernel-guided approach, or do we want to provide pieces of
> functionality that userspace exploits?
>
> Oh, or are you saying this would just replace the biggest chunk of my
> set_sigpending() function below?
```

I think so :)

cheers,

C.

---

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

---