

---

Subject: Re: [RFC][PATCH] Restore fd flags in restarted process

Posted by [serue](#) on Tue, 17 Jun 2008 22:31:11 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

Quoting [sukadev@us.ibm.com](mailto:sukadev@us.ibm.com) ([sukadev@us.ibm.com](mailto:sukadev@us.ibm.com)):

```
>
> >From e33c0c11cc612896cb12ddad1925037e52e76eb3 Mon Sep 17 00:00:00 2001
> From: Sukadev Bhattiprolu <sukadev@linux.vnet.ibm.com>
> Date: Tue, 17 Jun 2008 12:32:30 -0700
> Subject: [PATCH] Restore fd flags in restarted process.
>
> We currently get these flags using fcntl(F_GETFL) and save them while
> checkpointing but we do not restore them when restarting the process.
>
> Signed-off-by: Sukadev Bhattiprolu <sukadev@us.ibm.com>
```

Thanks, will apply and test.

-serge

```
> ---
> cr.c | 10 ++++++++
> sci.h | 7 +++++
> 2 files changed, 13 insertions(+), 4 deletions(-)
>
> diff --git a/cr.c b/cr.c
> index c52dd70..5163a3d 100644
> --- a/cr.c
> +++ b/cr.c
> @@ -251,7 +251,7 @@ int getfdinfo(pinfo_t *pi)
>   if (len >= 0) pi->fi[n].name[len] = 0;
>   stat(dname, &st);
>   pi->fi[n].mode = st.st_mode;
> - pi->fi[n].flag = PT_FCNTL(syscallpid, pi->fi[n].fdnum, F_GETFL);
> + pi->fi[n].flag = PT_FCNTL(syscallpid, pi->fi[n].fdnum, F_GETFL, 0);
>   if (S_ISREG(st.st_mode))
>     pi->fi[n].offset = (off_t)PT_LSEEK(syscallpid, pi->fi[n].fdnum, 0, SEEK_CUR);
>   else if (S_ISFIFO(st.st_mode))
> @@ -841,6 +841,14 @@ int restore_fd(int fd, pid_t pid)
>   }
>   }
>
> + /*
> +  * Restore any special flags this fd had
> +  */
> + ret = PT_FCNTL(pid, fdinfo->fdnum, F_SETFL, fdinfo->flag);
> + DEBUG("---- restore_fd() fd %d setfl flag 0x%x, ret %d\n",
> +   fdinfo->fdnum, fdinfo->flag, ret);
```

```
> +
> +
> free(fdinfo);
> }
> if (1) {
> diff --git a/sci.h b/sci.h
> index b0cac3c..0b32ae4 100644
> --- a/sci.h
> +++ b/sci.h
> @@ -138,10 +138,11 @@ int call_func(pid_t pid, int scratch, int flag, int funcaddr, int argc, ...);
>    0, 0, off, \
>    0, 0, w)
>
> -#define PT_FCNTL(p, fd, cmd) \
> - ptrace_syscall(p, 0, 0, SYS_fcntl, 2, \
> +#define PT_FCNTL(p, fd, cmd, arg) \
> + ptrace_syscall(p, 0, 0, SYS_fcntl, 3, \
>    0, 0, fd, \
> - 0, 0, cmd)
> + 0, 0, cmd, \
> + 0, 0, arg)
>
> #define PT_CLOSE(p, fd) \
> ptrace_syscall(p, 0, 0, SYS_close, 1, \
> --
> 1.5.2.5
```

---

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

---