

---

Subject: [PATCH 13/28] [PREP 13/14] Miscellaneous preparations in pid namespaces

Posted by [Pavel Emelianov](#) on Fri, 15 Jun 2007 16:11:28 GMT

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

---

The most important one is moving `exit_task_namespaces` behind `exit_notify` in `do_exit()` to make it possible to see the task's pid namespace to properly notify the parent.

The other important change is redefining `child_reaper()`. It turned out that all the places that use it pass current task and thus it is safe to dereference `task->nsproxy` pointer.

Signed-off-by: Pavel Emelianov <[xemul@openvz.org](mailto:xemul@openvz.org)>

---

```
include/linux/pid_namespace.h | 7 ++++++-
kernel/exit.c                 | 2 +-
kernel/pid.c                  | 2 ++
3 files changed, 9 insertions(+), 2 deletions(-)
```

--- ./include/linux/pid\_namespace.h.nsmiscprep 2007-06-15 15:12:25.000000000 +0400

+++ ./include/linux/pid\_namespace.h 2007-06-15 15:13:07.000000000 +0400

@@ -29,7 +29,9 @@ extern struct pid\_namespace init\_pid\_ns;

```
static inline void get_pid_ns(struct pid_namespace *ns)
{
+ #ifdef CONFIG_PID_NS
    kref_get(&ns->kref);
+ #endif
}
```

extern struct pid\_namespace \*copy\_pid\_ns(int flags, struct pid\_namespace \*ns);  
@@ -37,12 +39,15 @@ extern void free\_pid\_ns(struct kref \*kre

```
static inline void put_pid_ns(struct pid_namespace *ns)
{
+ #ifdef CONFIG_PID_NS
    kref_put(&ns->kref, free_pid_ns);
+ #endif
}
```

```
static inline struct task_struct *child_reaper(struct task_struct *tsk)
{
- return init_pid_ns.child_reaper;
+ BUG_ON(tsk != current);
+ return tsk->nsproxy->pid_ns->child_reaper;
```

```

}

#endif /* _LINUX_PID_NS_H */
--- ./kernel/exit.c.nsmiscprep 2007-06-15 15:03:10.000000000 +0400
+++ ./kernel/exit.c 2007-06-15 15:13:07.000000000 +0400
@@ -955,8 +955,8 @@ fastcall NORET_TYPE void do_exit(long co

    tsk->exit_code = code;
    proc_exit_connector(tsk);
- exit_task_namespaces(tsk);
    exit_notify(tsk);
+ exit_task_namespaces(tsk);
#ifdef CONFIG_NUMA
    mpol_free(tsk->mempolicy);
    tsk->mempolicy = NULL;
--- ./kernel/pid.c.nsmiscprep 2007-06-15 15:11:14.000000000 +0400
+++ ./kernel/pid.c 2007-06-15 15:13:07.000000000 +0400
@@ -82,6 +82,8 @@ struct pid_namespace init_pid_ns = {
    .child_reaper = &init_task
};

+EXPORT_SYMBOL_GPL(init_pid_ns);
+
/*
 * Note: disable interrupts while the pidmap_lock is held as an
 * interrupt might come in and do read_lock(&tasklist_lock).

```

---

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

---