

---

Subject: [PATCH 27/28] [MULTI 6/6] Moving the pid into the namespace  
Posted by [Pavel Emelianov](#) on Fri, 15 Jun 2007 16:29:44 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

This is the implementation of move\_pid\_to\_ns for multilevel model  
(see [PATCH 15/18] and [FLAT 6/6])

Signed-off-by: Pavel Emelianov <xemul@openvz.org>

---

pid.c | 42 +++  
1 files changed, 42 insertions(+)

```
--- ./kernel/pid.c.multiclone 2007-06-15 15:41:30.000000000 +0400
+++ ./kernel/pid.c 2007-06-15 15:42:36.000000000 +0400
@@ -531,6 +531,48 @@ struct pid * fastcall find_pid_ns(int nr
    pnr = find_nr_by_nr(nr, ns);
    return pnr != NULL ? pnr->pid : NULL;
}
+
+static inline int move_pid_to_ns(struct pid *pid, struct pid_namespace *ns)
+{
+ int err;
+
+ /*
+  * the pid is in this ns already. e.g. this may happen if
+  * the task has equal pid and pgid
+  */
+ if (find_nr_by_pid(pid, ns))
+ return 0;
+
+ err = alloc_pid_nr(pid, ns);
+ if (err < 0)
+ return err;
+
+ spin_lock_irq(&pidmap_lock);
+ hash_pid_nr(pid->pid_nrs);
+ spin_unlock_irq(&pidmap_lock);
+ return 0;
+}
+
+static inline void del_pid_from_ns(struct pid *pid, struct pid_namespace *ns)
+{
+ struct pid_number *pnr, *prev;
+
+ for_each_pid_nr(prev, pid) {
+ pnr = prev->next;
```

```
+ if (pnr != NULL && pnr->ns == ns)
+   goto found;
+ }
+ return;
+
+found:
+ prev->next = pnr->next;
+
+ spin_lock_irq(&pidmap_lock);
+ unhash_pid_nr(pnr);
+ spin_unlock_irq(&pidmap_lock);
+
+ free_pid_nr(pnr);
+}
#endif
#endif
```

---

Containers mailing list

Containers@lists.linux-foundation.org

<https://lists.linux-foundation.org/mailman/listinfo/containers>

---