

---

Subject: [PATCH 17/28] [FLAT 2/6] Helpers to obtain pid numbers

Posted by [Pavel Emelianov](#) on Fri, 15 Jun 2007 16:17:36 GMT

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

---

The is the implementation of [PREP 2/14] patch for the flat model

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

---

```
pid.h | 27 ++++++
sched.h | 102
+++++
2 files changed, 129 insertions(+)
```

```
--- ./include/linux/pid.h.flatnrs 2007-06-15 15:22:18.000000000 +0400
+++ ./include/linux/pid.h 2007-06-15 15:23:00.000000000 +0400
@@ -171,6 +171,33 @@ static inline int pid_ns_accessible(stru
    return 1;
}
#else
#ifdef CONFIG_PID_NS_FLAT
+static inline pid_t pid_nr(struct pid *pid)
+{
+ pid_t nr = 0;
+ if (pid)
+ nr = pid->nr;
+ return nr;
+}
+
+static inline pid_t pid_vnr(struct pid *pid)
+{
+ pid_t nr = 0;
+ if (pid)
+ nr = pid->vnr;
+ return nr;
+}
+
+static inline pid_t pid_nr_ns(struct pid *pid, struct pid_namespace *ns)
+{
+ return ns == &init_pid_ns ? pid_nr(pid) : pid_vnr(pid);
+}
+
+static inline int pid_ns_accessible(struct pid_namespace *ns, struct pid *pid)
+{
+ return pid->ns == &init_pid_ns || pid->ns == ns;
+}
+#endif
```

```

#endif

#define do_each_pid_task(pid, type, task) \
--- ./include/linux/sched.h.flatnrs 2007-06-15 15:19:14.000000000 +0400
+++ ./include/linux/sched.h 2007-06-15 15:23:00.000000000 +0400
@@ -1302,6 +1302,108 @@ static inline pid_t task_ppid_nr_ns(stru
    return rcu_dereference(tsk->real_parent)->tgid;
}
#else
+#ifdef CONFIG_PID_NS_FLAT
+static inline pid_t task_pid_nr(struct task_struct *tsk)
+{
+ return tsk->pid;
+}
+
+static inline pid_t task_pid_vnr(struct task_struct *tsk)
+{
+ return tsk->vpid;
+}
+
+static inline pid_t task_pid_nr_ns(struct task_struct *tsk,
+ struct pid_namespace *ns)
+{
+ return (ns == &init_pid_ns ?
+ task_pid_nr(tsk) : task_pid_vnr(tsk));
+}
+
+static inline void set_task_vpid(struct task_struct *tsk, pid_t nr)
+{
+ tsk->vpid = nr;
+}
+
+
+static inline pid_t task_tgid_nr(struct task_struct *tsk)
+{
+ return tsk->tgid;
+}
+
+static inline pid_t task_tgid_vnr(struct task_struct *tsk)
+{
+ return tsk->vtgid;
+}
+
+static inline pid_t task_tgid_nr_ns(struct task_struct *tsk,
+ struct pid_namespace *ns)
+{
+ return (ns == &init_pid_ns ?
+ task_tgid_nr(tsk) : task_tgid_vnr(tsk));

```

```

+}
+
+static inline void set_task_vtgid(struct task_struct *tsk, pid_t nr)
+{
+ tsk->vtgid = nr;
+}
+
+
+static inline pid_t task_pgrp_nr(struct task_struct *tsk)
+{
+ return tsk->signal->pgrp;
+}
+
+static inline pid_t task_pgrp_vnr(struct task_struct *tsk)
+{
+ return tsk->signal->vpgrp;
+}
+
+static inline pid_t task_pgrp_nr_ns(struct task_struct *tsk,
+ struct pid_namespace *ns)
+{
+ return (ns == &init_pid_ns ?
+ task_pgrp_nr(tsk) : task_pgrp_vnr(tsk));
+}
+
+static inline void set_task_vpgrp(struct task_struct *tsk, pid_t nr)
+{
+ tsk->signal->vpgrp = nr;
+}
+
+
+static inline pid_t task_session_nr(struct task_struct *tsk)
+{
+ return tsk->signal->__session;
+}
+
+static inline pid_t task_session_vnr(struct task_struct *tsk)
+{
+ return tsk->signal->vsession;
+}
+
+static inline pid_t task_session_nr_ns(struct task_struct *tsk,
+ struct pid_namespace *ns)
+{
+ return (ns == &init_pid_ns ?
+ task_session_nr(tsk) : task_session_vnr(tsk));
+}
+

```

```
+static inline void set_task_vsession(struct task_struct *tsk, pid_t nr)
+{
+ tsk->signal->vsession = nr;
+}
+
+
+static inline pid_t task_ppid_nr_ns(struct task_struct *tsk,
+ struct pid_namespace *ns)
+{
+ if (tsk->vpid == 1)
+ return 0;
+
+ return rcu_dereference(tsk->real_parent)->vtgid;
+}
+#endif
#endif

/**
```

---

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

---