
Subject: [PATCH 2/3] Prepare pid_nr() etc functions to work with not-NULL pids
Posted by [Pavel Emelianov](#) on Wed, 03 Oct 2007 14:20:43 GMT
[View Forum Message](#) <> [Reply to Message](#)

Just make the __pid_nr() etc functions that expect the argument to always be not NULL.

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

```
diff --git a/include/linux/pid.h b/include/linux/pid.h
index e29a900..50b6899 100644
--- a/include/linux/pid.h
+++ b/include/linux/pid.h
@@ -135,21 +135,32 @@ extern void zap_pid_ns_processes(struct
 * see also task_xid_nr() etc in include/linux/sched.h
 */
```

```
+static inline pid_t __pid_nr(struct pid *pid)
+{
+ return pid->numbers[0].nr;
+}
+
static inline pid_t pid_nr(struct pid *pid)
{
 pid_t nr = 0;
 if (pid)
- nr = pid->numbers[0].nr;
+ nr = __pid_nr(pid);
 return nr;
}

+pid_t __pid_nr_ns(struct pid *pid, struct pid_namespace *ns);
pid_t pid_nr_ns(struct pid *pid, struct pid_namespace *ns);

+static inline pid_t __pid_vnr(struct pid *pid)
+{
+ return pid->numbers[pid->level].nr;
+}
+
static inline pid_t pid_vnr(struct pid *pid)
{
 pid_t nr = 0;
 if (pid)
- nr = pid->numbers[pid->level].nr;
+ nr = __pid_vnr(pid);
 return nr;
}
```

```
}
```

```
diff --git a/kernel/pid.c b/kernel/pid.c
```

```
index d7388d7..b7a11cf 100644
```

```
--- a/kernel/pid.c
```

```
+++ b/kernel/pid.c
```

```
@@ -430,12 +441,12 @@ struct pid *find_get_pid(pid_t nr)
```

```
    return pid;
```

```
}
```

```
-pid_t pid_nr_ns(struct pid *pid, struct pid_namespace *ns)
```

```
+pid_t __pid_nr_ns(struct pid *pid, struct pid_namespace *ns)
```

```
{
```

```
    struct upid *upid;
```

```
    pid_t nr = 0;
```

```
- if (pid && ns->level <= pid->level) {
```

```
+ if (ns->level <= pid->level) {
```

```
    upid = &pid->numbers[ns->level];
```

```
    if (upid->ns == ns)
```

```
        nr = upid->nr;
```

```
@@ -443,6 +454,14 @@ pid_t pid_nr_ns(struct pid *pid, struct
```

```
    return nr;
```

```
}
```

```
+pid_t pid_nr_ns(struct pid *pid, struct pid_namespace *ns)
```

```
+{
```

```
+ pid_t nr = 0;
```

```
+ if (pid)
```

```
+ nr = __pid_nr_ns(pid, ns);
```

```
+ return nr;
```

```
+}
```

```
+
```

```
pid_t task_pid_nr_ns(struct task_struct *tsk, struct pid_namespace *ns)
```

```
{
```

```
    return pid_nr_ns(task_pid(tsk), ns);
```