Subject: [PATCH 1/2 (resend)] Clean up the kill_someting_info
Posted by Pavel Emelianov on Mon, 04 Feb 2008 13:02:02 GMT
View Forum Message <> Reply to Message

This is the first step (of two) in removing the kill_pgrp_info.

All the users of this function are in kernel/signal.c, but all
they need is to call __kill_pgrp_info() with the tasklist_lock
read-locked.

Fortunately, one of its users is the kill_something_info(),
which already needs this lock in one of its branches, so clean
these branches up and call the __kill_pgrp_info() directly.

Based on Oleg's view of how this function should look.

Changes from v1:
* Fixed double rcu_read_lock() in the if (pid > 0) branch.
  This caused udev die from segfault on FC3, as noticed by Andrew.

Signed-off-by: Oleg Nesterov <oleg@tv-sign.ru>
Signed-off-by: Pavel Emelyanov <xemul@openvz.org>

---
 kernel/signal.c | 26 ++++++++++++++++++----------
 1 files changed, 15 insertions(+), 11 deletions(-)

diff --git a/kernel/signal.c b/kernel/signal.c
index cc45a6b..a805c74 100644
--- a/kernel/signal.c
+++ b/kernel/signal.c
@@ -1134,14 +1134,22 @@ EXPORT_SYMBOL_GPL(kill_pid_info_as_uid);
 static int kill_something_info(int sig, struct siginfo *info, int pid)
 {
  int ret;
- rcu_read_lock();
- if (!pid) {
- ret = kill_pgrp_info(sig, info, task_pgrp(current));
- } else if (pid == -1) {
+
+ if (pid > 0) {
+ rcu_read_lock();
+ ret = kill_pid_info(sig, info, find_vpid(pid));
+ rcu_read_unlock();
+ return ret;
+ }
+
+ read_lock(&tasklist_lock);

```
+ if (pid != -1) {
+  ret = __kill_pgrp_info(sig, info,
+    pid ? find_vpid(-pid) : task_pgrp(current));
+ } else {
   int retval = 0, count = 0;
   struct task_struct * p;

-  read_lock(&tasklist_lock);
   for_each_process(p) {
    if (p->pid > 1 && !same_thread_group(p, current)) {
     int err = group_send_sig_info(sig, info, p);
@@ -1150,14 +1158,10 @@ static int kill_something_info(int sig, struct siginfo *info, int pid)
      retval = err;
    }
   }
-  read_unlock(&tasklist_lock);
   ret = count ? retval : -ESRCH;
- } else if (pid < 0) {
-  ret = kill_pgrp_info(sig, info, find_vpid(-pid));
- } else {
-  ret = kill_pid_info(sig, info, find_vpid(pid));
  }
- rcu_read_unlock();
+ read_unlock(&tasklist_lock);
+
  return ret;
 }

--
1.5.3.4
```