Subject: [PATCH] n_r3964: Use struct pid to track user space clients.
Posted by ebiederm on Mon, 11 Dec 2006 13:11:59 GMT
View Forum Message <> Reply to Message

Currently this driver tracks user space clients
it should send signals to. In the presenct of file
descriptor passing this is appears susceptible to
confusion from pid wrap around issues.

Replacing this with a struct pid prevents us from
getting confused, and prepares for a pid namespace
implementation.

Signed-off-by: Eric W. Biederman<ebiederm@xmission.com>
---
drivers/char/n_r3964.c | 37 ++++++++++++++++++++----------------
include/linux/n_r3964.h | 2 +-
2 files changed, 18 insertions(+), 21 deletions(-)

```
diff --git a/drivers/char/n_r3964.c b/drivers/char/n_r3964.c
index 103d338..dc6d418 100644
--- a/drivers/char/n_r3964.c
+++ b/drivers/char/n_r3964.c
@@ -125,8 +125,8 @@ static void transmit_block(struct r3964_info *pInfo);
 static void receive_char(struct r3964_info *pInfo, const unsigned char c);
 static void receive_error(struct r3964_info *pInfo, const char flag);
 static void on_timeout(unsigned long priv);
-static int enable_signals(struct r3964_info *pInfo, pid_t pid, int arg);
-static int read_telegram(struct r3964_info *pInfo, pid_t pid, unsigned char __user *buf);
+static int enable_signals(struct r3964_info *pInfo, struct pid *pid, int arg);
+static int read_telegram(struct r3964_info *pInfo, struct pid *pid, unsigned char __user *buf);
 static void add_msg(struct r3964_client_info *pClient, int msg_id, int arg,
             int error_code, struct r3964_block_header *pBlock);
 static struct r3964_message* remove_msg(struct r3964_info *pInfo,
@@ -829,7 +829,7 @@ static void on_timeout(unsigned long priv)
 }

 static struct r3964_client_info *findClient(
-  struct r3964_info *pInfo, pid_t pid)
+  struct r3964_info *pInfo, struct pid *pid)
 {
   struct r3964_client_info *pClient;

@@ -843,7 +843,7 @@ static struct r3964_client_info *findClient(
   return NULL;
 }

-static int enable_signals(struct r3964_info *pInfo, pid_t pid, int arg)
```

```
+static int enable_signals(struct r3964_info *pInfo, struct pid *pid, int arg)
 {
    struct r3964_client_info *pClient;
    struct r3964_client_info **ppClient;
@@ -858,7 +858,7 @@ static int enable_signals(struct r3964_info *pInfo, pid_t pid, int arg)

          if(pClient->pid == pid)
          {
-            TRACE_PS("removing client %d from client list", pid);
+            TRACE_PS("removing client %d from client list", pid_nr(pid));
             *ppClient = pClient->next;
             while(pClient->msg_count)
             {
@@ -869,6 +869,7 @@ static int enable_signals(struct r3964_info *pInfo, pid_t pid, int arg)
                 TRACE_M("enable_signals - msg kfree %p",pMsg);
               }
             }
+     put_pid(pClient->pid);
             kfree(pClient);
             TRACE_M("enable_signals - kfree %p",pClient);
             return 0;
@@ -892,10 +893,10 @@ static int enable_signals(struct r3964_info *pInfo, pid_t pid, int arg)
          if(pClient==NULL)
            return -ENOMEM;

-         TRACE_PS("add client %d to client list", pid);
+         TRACE_PS("add client %d to client list", pid_nr(pid));
   spin_lock_init(&pClient->lock);
          pClient->sig_flags=arg;
-         pClient->pid = pid;
+         pClient->pid = get_pid(pid);
          pClient->next=pInfo->firstClient;
          pClient->first_msg = NULL;
          pClient->last_msg = NULL;
@@ -908,7 +909,7 @@ static int enable_signals(struct r3964_info *pInfo, pid_t pid, int arg)
    return 0;
 }

-static int read_telegram(struct r3964_info *pInfo, pid_t pid, unsigned char __user *buf)
+static int read_telegram(struct r3964_info *pInfo, struct pid *pid, unsigned char __user *buf)
 {
     struct r3964_client_info *pClient;
     struct r3964_block_header *block;
@@ -1005,7 +1006,7 @@ queue_the_message:
    /* Send SIGIO signal to client process: */
    if(pClient->sig_flags & R3964_USE_SIGIO)
    {
-      kill_proc(pClient->pid, SIGIO, 1);
```

```
+      kill_pid(pClient->pid, SIGIO, 1);
    }
  }

@@ -1042,7 +1043,7 @@ static void remove_client_block(struct r3964_info *pInfo,
  {
     struct r3964_block_header *block;

-    TRACE_PS("remove_client_block PID %d", pClient->pid);
+    TRACE_PS("remove_client_block PID %d", pid_nr(pClient->pid));

     block=pClient->next_block_to_read;
     if(block)
@@ -1157,6 +1158,7 @@ static void r3964_close(struct tty_struct *tty)
             TRACE_M("r3964_close - msg kfree %p",pMsg);
           }
         }
+       put_pid(pClient->pid);
         kfree(pClient);
         TRACE_M("r3964_close - client kfree %p",pClient);
         pClient=pNext;
@@ -1193,12 +1195,11 @@ static ssize_t r3964_read(struct tty_struct *tty, struct file *file,
    struct r3964_client_message theMsg;
    DECLARE_WAITQUEUE (wait, current);

-   int pid = current->pid;
    int count;

    TRACE_L("read()");

-   pClient=findClient(pInfo, pid);
+   pClient=findClient(pInfo, task_pid(current));
    if(pClient)
    {
      pMsg = remove_msg(pInfo, pClient);
@@ -1252,7 +1253,6 @@ static ssize_t r3964_write(struct tty_struct * tty, struct file * file,
    struct r3964_block_header *pHeader;
    struct r3964_client_info *pClient;
    unsigned char *new_data;
-   int pid;

    TRACE_L("write request, %d characters", count);
  /*
@@ -1295,9 +1295,7 @@ static ssize_t r3964_write(struct tty_struct * tty, struct file * file,
    pHeader->locks = 0;
    pHeader->owner = NULL;

-   pid=current->pid;
```

```
-
-   pClient=findClient(pInfo, pid);
+   pClient=findClient(pInfo, task_pid(current));
    if(pClient)
    {
      pHeader->owner = pClient;
@@ -1328,7 +1326,7 @@ static int r3964_ioctl(struct tty_struct * tty, struct file * file,
    switch(cmd)
    {
      case R3964_ENABLE_SIGNALS:
-       return enable_signals(pInfo, current->pid, arg);
+       return enable_signals(pInfo, task_pid(current), arg);
      case R3964_SETPRIORITY:
        if(arg<R3964_MASTER || arg>R3964_SLAVE)
          return -EINVAL;
@@ -1341,7 +1339,7 @@ static int r3964_ioctl(struct tty_struct * tty, struct file * file,
          pInfo->flags &= ~R3964_BCC;
        return 0;
      case R3964_READ_TELEGRAM:
-       return read_telegram(pInfo, current->pid, (unsigned char __user *)arg);
+       return read_telegram(pInfo, task_pid(current), (unsigned char __user *)arg);
      default:
        return -ENOIOCTLCMD;
    }
@@ -1357,7 +1355,6 @@ static unsigned int r3964_poll(struct tty_struct * tty, struct file * file,
 {
    struct r3964_info *pInfo=(struct r3964_info*)tty->disc_data;
-   int pid=current->pid;
    struct r3964_client_info *pClient;
    struct r3964_message *pMsg=NULL;
    unsigned long flags;
@@ -1365,7 +1362,7 @@ static unsigned int r3964_poll(struct tty_struct * tty, struct file * file,

    TRACE_L("POLL");

-   pClient=findClient(pInfo,pid);
+   pClient=findClient(pInfo, task_pid(current));
    if(pClient)
      {
        poll_wait(file, &pInfo->read_wait, wait);
diff --git a/include/linux/n_r3964.h b/include/linux/n_r3964.h
index db4f377..de24af7 100644
--- a/include/linux/n_r3964.h
+++ b/include/linux/n_r3964.h
@@ -116,7 +116,7 @@ struct r3964_message;

 struct r3964_client_info {
```

```
  spinlock_t     lock;
- pid_t          pid;
+ struct pid    *pid;
  unsigned int   sig_flags;

  struct r3964_client_info *next;
--
1.4.4.1.g278f
```

_____

Containers mailing list
Containers@lists.osdl.org
https://lists.osdl.org/mailman/listinfo/containers