
Subject: [PATCH 4/12] Don't operate with pid_t in rtmutex tester
Posted by [Pavel Emelianov](#) on Tue, 29 Jan 2008 13:49:27 GMT
[View Forum Message](#) <> [Reply to Message](#)

The proper behavior to store task's pid and get this task later is to get the struct pid pointer and get the task with the pid_task() call.

Make it for rt_mutex_waiter->deadlock_task_pid field.

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

kernel/rtmutex-debug.c | 12 ++++++++
kernel/rtmutex_common.h | 2 +-
2 files changed, 10 insertions(+), 4 deletions(-)

diff --git a/kernel/rtmutex-debug.c b/kernel/rtmutex-debug.c

index 56d73cb..40ec85c 100644

--- a/kernel/rtmutex-debug.c

+++ b/kernel/rtmutex-debug.c

@@ -130,7 +130,7 @@ void debug_rt_mutex_deadlock(int detect, struct rt_mutex_waiter
*act_waiter,

```
    task = rt_mutex_owner(act_waiter->lock);  
    if (task && task != current) {  
-   act_waiter->deadlock_task_pid = task->pid;  
+   act_waiter->deadlock_task_pid = get_pid(task_pid(task));  
    act_waiter->deadlock_lock = lock;  
    }  
}
```

@@ -142,9 +142,12 @@ void debug_rt_mutex_print_deadlock(struct rt_mutex_waiter *waiter)
 if (!waiter->deadlock_lock || !rt_trace_on)
 return;

```
- task = find_task_by_pid(waiter->deadlock_task_pid);  
- if (!task)  
+ rcu_read_lock();  
+ task = pid_task(waiter->deadlock_task_pid, PIDTYPE_PID);  
+ if (!task) {  
+ rcu_read_unlock();  
    return;  
+ }
```

TRACE_OFF_NOLOCK();

@@ -173,6 +176,7 @@ void debug_rt_mutex_print_deadlock(struct rt_mutex_waiter *waiter)
 current->comm, task_pid_nr(current));

```

    dump_stack();
    debug_show_all_locks();
+ rcu_read_unlock();

    printk("[ turning off deadlock detection."
           "Please report this trace. ]\n\n");
@@ -203,10 +207,12 @@ void debug_rt_mutex_init_waiter(struct rt_mutex_waiter *waiter)
    memset(waiter, 0x11, sizeof(*waiter));
    plist_node_init(&waiter->list_entry, MAX_PRIO);
    plist_node_init(&waiter->pi_list_entry, MAX_PRIO);
+ waiter->deadlock_task_pid = NULL;
}

void debug_rt_mutex_free_waiter(struct rt_mutex_waiter *waiter)
{
+ put_pid(waiter->deadlock_task_pid);
    TRACE_WARN_ON(!plist_node_empty(&waiter->list_entry));
    TRACE_WARN_ON(!plist_node_empty(&waiter->pi_list_entry));
    TRACE_WARN_ON(waiter->task);
diff --git a/kernel/rtmutex_common.h b/kernel/rtmutex_common.h
index 2d3b835..e124bf5 100644
--- a/kernel/rtmutex_common.h
+++ b/kernel/rtmutex_common.h
@@ -51,7 +51,7 @@ struct rt_mutex_waiter {
    struct rt_mutex *lock;
#ifdef CONFIG_DEBUG_RT_MUTEXES
    unsigned long ip;
- pid_t deadlock_task_pid;
+ struct pid *deadlock_task_pid;
    struct rt_mutex *deadlock_lock;
#endif
};
--
1.5.3.4

```
