Subject: [RFC][PATCH v2 5/7] Invoke the ipcns notifier chain as a work item
Posted by Nadia Derbey on Thu, 31 Jan 2008 13:40:23 GMT
View Forum Message <> Reply to Message

[PATCH 05/07]

This patche makes the memory hotplug chain's mutex held for a shorter time:
when memory is offlined or onlined a work item is added to the global
workqueue.
When the work item is run, it notifies the ipcns notifier chain with the
IPCNS_MEMCHANGED event.

Signed-off-by: Nadia Derbey <Nadia.Derbey@bull.net>

---
 ipc/util.c | 17 ++++++++++++++--
 1 file changed, 15 insertions(+), 2 deletions(-)

Index: linux-2.6.24/ipc/util.c
===
--- linux-2.6.24.orig/ipc/util.c 2008-01-31 11:04:51.000000000 +0100
+++ linux-2.6.24/ipc/util.c 2008-01-31 11:41:01.000000000 +0100
@@ -57,6 +57,14 @@ atomic_t nr_ipc_ns = ATOMIC_INIT(1);

 #ifdef CONFIG_MEMORY_HOTPLUG

+static void ipc_memory_notifier(struct work_struct *work)
+{
+ ipcns_notify(IPCNS_MEMCHANGED);
+}
+
+static DECLARE_WORK(ipc_memory_wq, ipc_memory_notifier);
+
+
 static int ipc_memory_callback(struct notifier_block *self,
     unsigned long action, void *arg)
 {
@@ -65,9 +73,14 @@ static int ipc_memory_callback(struct no
  case MEM_OFFLINE: /* or offline: it's time to recompute msgmni */
   /*
    * This is done by invoking the ipcns notifier chain with the
-   * IPC_MEMCHANGED event
+   * IPC_MEMCHANGED event.
+   * In order not to keep the lock on the hotplug memory chain
+   * for too long, queue a work item that will, when waken up,
+   * activate the ipcns notification chain.
+   * No need to keep several ipc work items on the queue.
    */

```
-  ipcns_notify(IPCNS_MEMCHANGED);
+  if (!work_pending(&ipc_memory_wq))
+   schedule_work(&ipc_memory_wq);
   break;
  case MEM_GOING_ONLINE:
  case MEM_GOING_OFFLINE:

--
```

_____

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