
Subject: [PATCH 7/8] Do not recompute msgmni anymore if explicitly set by user
Posted by [Nadia Derby](#) on Mon, 11 Feb 2008 14:16:53 GMT
[View Forum Message](#) <> [Reply to Message](#)

[PATCH 07/08]

This patch makes msgmni not recomputed anymore upon ipc namespace creation / removal or memory add/remove, as soon as it has been set from userland.

As soon as msgmni is explicitly set via procfs or sysctl(), the associated callback routine is unregistered from the ipc namespace notifier chain.

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

ipc/ipc_sysctl.c | 43 ++++++
1 file changed, 41 insertions(+), 2 deletions(-)

Index: linux-2.6.24-mm1/ipc/ipc_sysctl.c

=====

--- linux-2.6.24-mm1.orig/ipc/ipc_sysctl.c 2008-02-08 16:07:15.000000000 +0100

+++ linux-2.6.24-mm1/ipc/ipc_sysctl.c 2008-02-08 16:08:32.000000000 +0100

@@ -35,6 +35,24 @@ static int proc_ipc_dointvec(ctl_table *
return proc_dointvec(&ipc_table, write, filp, buffer, lenp, ppos);
}

+static int proc_ipc_callback_dointvec(ctl_table *table, int write,
+ struct file *filp, void __user *buffer, size_t *lenp, loff_t *ppos)

+{

+ size_t lenp_bef = *lenp;

+ int rc;

+

+ rc = proc_ipc_dointvec(table, write, filp, buffer, lenp, ppos);

+

+ if (write && !rc && lenp_bef == *lenp)

+ /*

+ * Tunable has successfully been changed from userland:

+ * disable its automatic recomputing.

+ */

+ unregister_ipcns_notifier(current->nsproxy->ipc_ns);

+

+ return rc;

+}

+

static int proc_ipc_doulongvec_minmax(ctl_table *table, int write,
struct file *filp, void __user *buffer, size_t *lenp, loff_t *ppos)

{

```

@@ -49,6 +67,7 @@ static int proc_ipc_doulongvec_minmax(ct
#else
#define proc_ipc_doulongvec_minmax NULL
#define proc_ipc_dointvec NULL
+#define proc_ipc_callback_dointvec NULL
#endif

#ifdef CONFIG_SYSCTL_SYSCALL
@@ -90,8 +109,28 @@ static int sysctl_ipc_data(ctl_table *ta
}
return 1;
}
+
+static int sysctl_ipc_registered_data(ctl_table *table, int __user *name,
+ int nlen, void __user *oldval, size_t __user *oldlenp,
+ void __user *newval, size_t newlen)
+{
+ int rc;
+
+ rc = sysctl_ipc_data(table, name, nlen, oldval, oldlenp, newval,
+ newlen);
+
+ if (newval && newlen && rc > 0)
+ /*
+  * Tunable has successfully been changed from userland:
+  * disable its automatic recomputing.
+  */
+ unregister_ipcns_notifier(current->nsproxy->ipc_ns);
+
+ return rc;
+}
#else
#define sysctl_ipc_data NULL
+#define sysctl_ipc_registered_data NULL
#endif

static struct ctl_table ipc_kern_table[] = {
@@ -137,8 +176,8 @@ static struct ctl_table ipc_kern_table[]
.data = &init_ipc_ns.msg_ctlmni,
.maxlen = sizeof (init_ipc_ns.msg_ctlmni),
.mode = 0644,
- .proc_handler = proc_ipc_dointvec,
- .strategy = sysctl_ipc_data,
+ .proc_handler = proc_ipc_callback_dointvec,
+ .strategy = sysctl_ipc_registered_data,
},
{
ctl_name = KERN_MSGMNB,

```

--

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