

---

Subject: [PATCH 3/4] sysctl: Simplify ipc ns specific sysctls  
Posted by [ebiederm](#) on Mon, 27 Nov 2006 05:05:10 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

This patch refactors the ipc sysctl support so that it is simpler, more readable, and prepares for fixing the bug with the wrong values being returned in the sys\_sysctl interface.

The function proc\_do\_ipc\_string was misnamed as it never handled strings. It's magic of when to work with strings and when to work with longs belonged in the sysctl table. I couldn't tell if the code would work if you disabled the ipc namespace but it certainly looked like it would have problems.

Signed-off-by: Eric W. Biederman <[ebiederm@xmission.com](mailto:ebiederm@xmission.com)>

---

kernel/sysctl.c | 106 ++++++-----  
1 files changed, 49 insertions(+), 57 deletions(-)

diff --git a/kernel/sysctl.c b/kernel/sysctl.c

index 63db5a5..638aa14 100644

--- a/kernel/sysctl.c

+++ b/kernel/sysctl.c

@@ -91,7 +91,9 @@ #ifdef CONFIG\_CHR\_DEV\_SG

extern int sg\_big\_buff;

#endif

#ifdef CONFIG\_SYSVIPC

-static int proc\_do\_ipc\_string(ctl\_table \*table, int write, struct file \*filp,

+static int proc\_ipc\_dointvec(ctl\_table \*table, int write, struct file \*filp,

+ void \_\_user \*buffer, size\_t \*lenp, loff\_t \*ppos);

+static int proc\_ipc\_doulongvec\_minmax(ctl\_table \*table, int write, struct file \*filp,

void \_\_user \*buffer, size\_t \*lenp, loff\_t \*ppos);

#endif

@@ -188,6 +190,18 @@ static void put\_uts(ctl\_table \*table, in

up\_write(&uts\_sem);

}

+#ifdef CONFIG\_SYSVIPC

+static void \*get\_ipc(ctl\_table \*table, int write)

+{

+ char \*which = table->data;

+ struct ipc\_namespace \*ipc\_ns = current->nsproxy->ipc\_ns;

+ which = (which - (char \*)&init\_ipc\_ns) + (char \*)ipc\_ns;

+ return which;

+}

+#else

+#define get\_ipc(T,W) ((T)->data)

```

+ #endif
+
+ /* /proc declarations: */

#ifdef CONFIG_PROC_SYSCTL
@@ -457,58 +471,58 @@ #ifdef CONFIG_SYSVIPC
{
    .ctl_name = KERN_SHMMAX,
    .procname = "shmmax",
-   .data = NULL,
-   .maxlen = sizeof (size_t),
+   .data = &init_ipc_ns.shm_ctlmax,
+   .maxlen = sizeof (init_ipc_ns.shm_ctlmax),
    .mode = 0644,
-   .proc_handler = &proc_do_ipc_string,
+   .proc_handler = &proc_ipc_doulongvec_minmax,
},
{
    .ctl_name = KERN_SHMALL,
    .procname = "shmall",
-   .data = NULL,
-   .maxlen = sizeof (size_t),
+   .data = &init_ipc_ns.shm_ctlall,
+   .maxlen = sizeof (init_ipc_ns.shm_ctlall),
    .mode = 0644,
-   .proc_handler = &proc_do_ipc_string,
+   .proc_handler = &proc_ipc_doulongvec_minmax,
},
{
    .ctl_name = KERN_SHMMNI,
    .procname = "shmmni",
-   .data = NULL,
-   .maxlen = sizeof (int),
+   .data = &init_ipc_ns.shm_ctlmni,
+   .maxlen = sizeof (init_ipc_ns.shm_ctlmni),
    .mode = 0644,
-   .proc_handler = &proc_do_ipc_string,
+   .proc_handler = &proc_ipc_dointvec,
},
{
    .ctl_name = KERN_MSGMAX,
    .procname = "msgmax",
-   .data = NULL,
-   .maxlen = sizeof (int),
+   .data = &init_ipc_ns.msg_ctlmax,
+   .maxlen = sizeof (init_ipc_ns.msg_ctlmax),
    .mode = 0644,
-   .proc_handler = &proc_do_ipc_string,

```

```

+ .proc_handler = &proc_ipc_dointvec,
},
{
    .ctl_name = KERN_MSGMNI,
    .procname = "msgmni",
- .data = NULL,
- .maxlen = sizeof (int),
+ .data = &init_ipc_ns.msg_ctlmni,
+ .maxlen = sizeof (init_ipc_ns.msg_ctlmni),
    .mode = 0644,
- .proc_handler = &proc_do_ipc_string,
+ .proc_handler = &proc_ipc_dointvec,
},
{
    .ctl_name = KERN_MSGMNB,
    .procname = "msgmnb",
- .data = NULL,
- .maxlen = sizeof (int),
+ .data = &init_ipc_ns.msg_ctlmnb,
+ .maxlen = sizeof (init_ipc_ns.msg_ctlmnb),
    .mode = 0644,
- .proc_handler = &proc_do_ipc_string,
+ .proc_handler = &proc_ipc_dointvec,
},
{
    .ctl_name = KERN_SEM,
    .procname = "sem",
- .data = NULL,
+ .data = &init_ipc_ns.sem_ctls,
    .maxlen = 4*sizeof (int),
    .mode = 0644,
- .proc_handler = &proc_do_ipc_string,
+ .proc_handler = &proc_ipc_dointvec,
},
#endif
#ifdef CONFIG_MAGIC_SYSRQ
@@ -2321,46 +2335,24 @@ int proc_dointvec_ms_jiffies(ctl_table *
}

#ifdef CONFIG_SYSVIPC
-static int proc_do_ipc_string(ctl_table *table, int write, struct file *filp,
- void __user *buffer, size_t *lenp, loff_t *ppos)
+static int proc_ipc_dointvec(ctl_table *table, int write, struct file *filp,
+ void __user *buffer, size_t *lenp, loff_t *ppos)
{
- void *data;
- struct ipc_namespace *ns;
-

```

```

- ns = current->nsproxy->ipc_ns;
-
- switch (table->ctl_name) {
- case KERN_SHMMAX:
- data = &ns->shm_ctlmax;
- goto proc_minmax;
- case KERN_SHMALL:
- data = &ns->shm_ctlall;
- goto proc_minmax;
- case KERN_SHMMNI:
- data = &ns->shm_ctlmni;
- break;
- case KERN_MSGMAX:
- data = &ns->msg_ctlmax;
- break;
- case KERN_MSGMNI:
- data = &ns->msg_ctlmni;
- break;
- case KERN_MSGMNB:
- data = &ns->msg_ctlmnb;
- break;
- case KERN_SEM:
- data = &ns->sem_ctls;
- break;
- default:
- return -EINVAL;
- }
-
- return __do_proc_dointvec(data, table, write, filp, buffer,
+ void *which;
+ which = get_ipc(table, write);
+ return __do_proc_dointvec(which, table, write, filp, buffer,
+ lenp, ppos, NULL, NULL);
-proc_minmax:
- return __do_proc_doulongvec_minmax(data, table, write, filp, buffer,
+}
+
+static int proc_ipc_doulongvec_minmax(ctl_table *table, int write,
+ struct file *filp, void __user *buffer, size_t *lenp, loff_t *ppos)
+{
+ void *which;
+ which = get_ipc(table, write);
+ return __do_proc_doulongvec_minmax(which, table, write, filp, buffer,
+ lenp, ppos, 1l, 1l);
+}
+
+
#endif

```

```
static int proc_do_cad_pid(ctl_table *table, int write, struct file *filp,
```

```
--
```

1.4.2.rc3.g7e18e-dirty

---

Containers mailing list

Containers@lists.osdl.org

<https://lists.osdl.org/mailman/listinfo/containers>

---