
Subject: [PATCH 1/4] sysctl: Simplify sysctl_uts_string
Posted by [ebiederm](#) on Mon, 27 Nov 2006 05:05:08 GMT

[View Forum Message](#) <> [Reply to Message](#)

This patch introduces get_uts and put_uts (used later)
and removes most of the special cases for when UTS namespace is compiled in.

Signed-off-by: Eric W. Biederman <ebiederm@xmission.com>

kernel/sysctl.c | 128 ++++++-----
1 files changed, 26 insertions(+), 102 deletions(-)

```
diff --git a/kernel/sysctl.c b/kernel/sysctl.c
index 09e569f..0521884 100644
--- a/kernel/sysctl.c
+++ b/kernel/sysctl.c
@@ -162,6 +162,28 @@ #ifdef HAVE_ARCH_PICK_MMAP_LAYOUT
int sysctl_legacy_va_layout;
#endif

+static void *get_uts(ctl_table *table, int write)
+{
+ char *which = table->data;
+#ifdef CONFIG_UTS_NS
+ struct uts_namespace *uts_ns = current->nsproxy->uts_ns;
+ which = (which - (char *)&init_uts_ns) + (char *)uts_ns;
#endif
+ if (!write)
+ down_read(&uts_sem);
+ else
+ down_write(&uts_sem);
+ return which;
+}
+
+static void put_uts(ctl_table *table, int write, void *which)
+{
+ if (!write)
+ up_read(&uts_sem);
+ else
+ up_write(&uts_sem);
+}
+
/* /proc declarations: */

#endif CONFIG_PROC_SYSCTL
@@ -228,7 +250,6 @@ #endif
};
```

```

static ctl_table kern_table[] = {
-#ifndef CONFIG_UTS_NS
{
    .ctl_name = KERN_OSTYPE,
    .procname = "ostype",
@@ @ -274,54 +295,6 @@ #ifndef CONFIG_UTS_NS
    .proc_handler = &proc_do_uts_string,
    .strategy = &sysctl_string,
},
-#else /* !CONFIG_UTS_NS */
{
    .ctl_name = KERN_OSTYPE,
    .procname = "ostype",
    .data = NULL,
    /* could maybe use __NEW_UTS_LEN here? */
    . maxlen = FIELD_SIZEOF(struct new_utsname, sysname),
    .mode = 0444,
    .proc_handler = &proc_do_uts_string,
    .strategy = &sysctl_string,
},
{
    .ctl_name = KERN_OSRELEASE,
    .procname = "osrelease",
    .data = NULL,
    . maxlen = FIELD_SIZEOF(struct new_utsname, release),
    .mode = 0444,
    .proc_handler = &proc_do_uts_string,
    .strategy = &sysctl_string,
},
{
    .ctl_name = KERN_VERSION,
    .procname = "version",
    .data = NULL,
    . maxlen = FIELD_SIZEOF(struct new_utsname, version),
    .mode = 0444,
    .proc_handler = &proc_do_uts_string,
    .strategy = &sysctl_string,
},
{
    .ctl_name = KERN_NODENAME,
    .procname = "hostname",
    .data = NULL,
    . maxlen = FIELD_SIZEOF(struct new_utsname, nodename),
    .mode = 0644,
    .proc_handler = &proc_do_uts_string,
    .strategy = &sysctl_string,
},
{

```

```

- .ctl_name = KERN_DOMAINNAME,
- .procname = "domainname",
- .data = NULL,
- . maxlen = FIELD_SIZEOF(struct new_utsname, domainname),
- .mode = 0644,
- .proc_handler = &proc_do_uts_string,
- .strategy = &sysctl_string,
- },
#endif /* !CONFIG_UTS_NS */
{
.ctl_name = KERN_PANIC,
.procname = "panic",
@@ -1755,66 +1728,17 @@ int proc_destring(ctl_table *table, int
 * Special case of destring for the UTS structure. This has locks
 * to observe. Should this be in kernel/sys.c ****
*/
-
#ifndef CONFIG_UTS_NS
static int proc_do_uts_string(ctl_table *table, int write, struct file *filp,
- void __user *buffer, size_t *lenp, loff_t *ppos)
-{
- int r;
-
- if (!write) {
- down_read(&uts_sem);
- r=proc_destring(table,0,filp,buffer,lenp, ppos);
- up_read(&uts_sem);
- } else {
- down_write(&uts_sem);
- r=proc_destring(table,1,filp,buffer,lenp, ppos);
- up_write(&uts_sem);
- }
- return r;
-}
#endif /* !CONFIG_UTS_NS */
static int proc_do_uts_string(ctl_table *table, int write, struct file *filp,
    void __user *buffer, size_t *lenp, loff_t *ppos)
{
int r;
- struct uts_namespace* uts_ns = current->nsproxy->uts_ns;
- char* which;
-
- switch (table->ctl_name) {
- case KERN_OSTYPE:
- which = uts_ns->name.sysname;
- break;
- case KERN_NODENAME:
- which = uts_ns->name.nodename;

```

```

- break;
- case KERN_OSRELEASE:
- which = uts_ns->name.release;
- break;
- case KERN_VERSION:
- which = uts_ns->name.version;
- break;
- case KERN_DOMAINNAME:
- which = uts_ns->name.domainname;
- break;
- default:
- r = -EINVAL;
- goto out;
- }

-
- if (!write) {
- down_read(&uts_sem);
- r=_proc_do_string(which,table->maxlen,0,filp,buffer,lenp, ppos);
- up_read(&uts_sem);
- } else {
- down_write(&uts_sem);
- r=_proc_do_string(which,table->maxlen,1,filp,buffer,lenp, ppos);
- up_write(&uts_sem);
- }
- out:
+ void *which;
+ which = get_uts(table, write);
+ r = _proc_do_string(which, table->maxlen,write,filp,buffer,lenp, ppos);
+ put_uts(table, write, which);
 return r;
}
#endif /* !CONFIG_UTS_NS */

```

```

static int do_proc_dointvec_conv(int *negp, unsigned long *lvalp,
 int *valp,
--
```

1.4.2.rc3.g7e18e-dirty

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