
Subject: [PATCH 55/59] sysctl: Remove insert_at_head from register_sysctl
Posted by [ebiederm](#) on Tue, 16 Jan 2007 16:40:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

From: Eric W. Biederman <ebiederm@xmission.com> - unquoted

The semantic effect of insert_at_head is that it would allow new registered sysctl entries to override existing sysctl entries of the same name. Which is pain for caching and the proc interface never implemented.

I have done an audit and discovered that none of the current users of register_sysctl care as (except for directories) they do not register duplicate sysctl entries.

So this patch simply removes the support for overriding existing entries in the sys_sysctl interface since no one uses it or cares and it makes future enhancements harder.

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

arch/arm/kernel/isa.c		2 +-
arch/frv/kernel/pm.c		2 +-
arch/frv/kernel/sysctl.c		2 +-
arch/ia64/kernel/crash.c		2 +-
arch/ia64/kernel/perfmon.c		2 +-
arch/ia64/sn/kernel/xpc_main.c		2 +-
arch/mips/au1000/common/power.c		2 +-
arch/mips/lasat/sysctl.c		2 +-
arch/powerpc/kernel/idle.c		2 +-
arch/ppc/kernel/ppc_htab.c		2 +-
arch/s390/appldata/appldata_base.c		4 +++-
arch/s390/kernel/debug.c		2 +-
arch/s390/mm/cmm.c		2 +-
arch/sh64/kernel/traps.c		2 +-
arch/x86_64/ia32/ia32_binfmt.c		2 +-
arch/x86_64/kernel/vsyscall.c		2 +-
arch/x86_64/mm/init.c		2 +-
drivers/cdrom/cdrom.c		2 +-
drivers/char/hpet.c		2 +-
drivers/char/ipmi/ipmi_poweroff.c		2 +-
drivers/char/rtc.c		2 +-
drivers/macintosh/mac_hid.c		2 +-
drivers/md/md.c		2 +-
drivers/net/wireless/arlan-proc.c		2 +-
drivers/parport/procfs.c		6 +++--
drivers/scsi/scsi_sysctl.c		2 +-
fs/coda/sysctl.c		2 +-

```

fs/dquot.c | 2 +-
fs/lockd/svc.c | 2 +-
fs/nfs/sysctl.c | 2 +-
fs/ntfs/sysctl.c | 2 +-
fs/ocfs2/cluster/nodemanager.c | 2 +-
fs/xfs/linux-2.6/xfs_sysctl.c | 2 +-
include/linux/sysctl.h | 4 +++-
ipc/ipc_sysctl.c | 2 +-
ipc/mqueue.c | 2 +-
kernel/sysctl.c | 9 ++-----
kernel/utsname_sysctl.c | 2 +-
net/appletalk/sysctl_net_atalk.c | 2 +-
net/ax25/sysctl_net_ax25.c | 2 +-
net/bridge/br_netfilter.c | 2 +-
net/core/neighbour.c | 2 +-
net/dccp/sysctl.c | 2 +-
net/decnet/dn_dev.c | 2 +-
net/decnet/sysctl_net_decnet.c | 2 +-
net/ipv4/devinet.c | 4 +++-
net/ipv4/ipvs/ip_vs_ctl.c | 2 +-
net/ipv4/ipvs/ip_vs_lblc.c | 2 +-
net/ipv4/ipvs/ip_vs_lblcr.c | 2 +-
net/ipv4/netfilter/ip_conntrack_proto_sctp.c | 2 +-
net/ipv4/netfilter/ip_conntrack_standalone.c | 2 +-
net/ipv4/netfilter/ip_queue.c | 2 +-
net/ipv6/addrconf.c | 4 +++-
net/ipv6/netfilter/ip6_queue.c | 2 +-
net/ipv6/sysctl_net_ipv6.c | 2 +-
net/ipx/sysctl_net_ipx.c | 2 +-
net/irda/irsysctl.c | 2 +-
net/llc/sysctl_net_llc.c | 2 +-
net/netfilter/nf_conntrack_standalone.c | 2 +-
net/netfilter/nf_sysctl.c | 2 +-
net/netrom/sysctl_net_netrom.c | 2 +-
net/rose/sysctl_net_rose.c | 2 +-
net/rxrpc/sysctl.c | 2 +-
net/sctp/sysctl.c | 2 +-
net/sunrpc/sysctl.c | 2 +-
net/sunrpc/xprtsock.c | 2 +-
net/unix/sysctl_net_unix.c | 2 +-
net/x25/sysctl_net_x25.c | 2 +-
68 files changed, 75 insertions(+), 80 deletions(-)

```

diff --git a/arch/arm/kernel/isa.c b/arch/arm/kernel/isa.c

index 54bbd9f..50a30bc 100644

--- a/arch/arm/kernel/isa.c

+++ b/arch/arm/kernel/isa.c

@@ -70,5 +70,5 @@ register_isa_ports(unsigned int membase, unsigned int portbase, unsigned

```

int por
    isa_membase = membase;
    isa_portbase = portbase;
    isa_portshift = portshift;
- isa_sysctl_header = register_sysctl_table(ctl_bus, 0);
+ isa_sysctl_header = register_sysctl_table(ctl_bus);
}
diff --git a/arch/frv/kernel/pm.c b/arch/frv/kernel/pm.c
index aa50333..c57ce3f 100644
--- a/arch/frv/kernel/pm.c
+++ b/arch/frv/kernel/pm.c
@@ -455,7 +455,7 @@ static struct ctl_table pm_dir_table[] =
    */
    static int __init pm_init(void)
    {
- register_sysctl_table(pm_dir_table, 0);
+ register_sysctl_table(pm_dir_table);
        return 0;
    }

diff --git a/arch/frv/kernel/sysctl.c b/arch/frv/kernel/sysctl.c
index 577ad16..3e9d7e0 100644
--- a/arch/frv/kernel/sysctl.c
+++ b/arch/frv/kernel/sysctl.c
@@ -216,7 +216,7 @@ static struct ctl_table frv_dir_table[] =
    */
    static int __init frv_sysctl_init(void)
    {
- register_sysctl_table(frv_dir_table, 0);
+ register_sysctl_table(frv_dir_table);
        return 0;
    }

diff --git a/arch/ia64/kernel/crash.c b/arch/ia64/kernel/crash.c
index bc2f64d..8a2cb76 100644
--- a/arch/ia64/kernel/crash.c
+++ b/arch/ia64/kernel/crash.c
@@ -214,7 +214,7 @@ machine_crash_setup(void)
    if((ret = register_die_notifier(&kdump_init_notifier_nb)) != 0)
        return ret;
#ifdef CONFIG_SYSCTL
- register_sysctl_table(sys_table, 0);
+ register_sysctl_table(sys_table);
#endif
    return 0;
}
diff --git a/arch/ia64/kernel/perfmon.c b/arch/ia64/kernel/perfmon.c
index 8c679ab..8a15377 100644

```

```

--- a/arch/ia64/kernel/perfmon.c
+++ b/arch/ia64/kernel/perfmon.c
@@ -6727,7 +6727,7 @@ pfm_init(void)
/*
 * create /proc/sys/kernel/perfmon (for debugging purposes)
 */
- pfm_sysctl_header = register_sysctl_table(pfm_sysctl_root, 0);
+ pfm_sysctl_header = register_sysctl_table(pfm_sysctl_root);

/*
 * initialize all our spinlocks
diff --git a/arch/ia64/sn/kernel/xpc_main.c b/arch/ia64/sn/kernel/xpc_main.c
index e04f7b5..68355ef 100644
--- a/arch/ia64/sn/kernel/xpc_main.c
+++ b/arch/ia64/sn/kernel/xpc_main.c
@@ -1241,7 +1241,7 @@ xpc_init(void)
    snprintf(xpc_part->bus_id, BUS_ID_SIZE, "part");
    snprintf(xpc_chan->bus_id, BUS_ID_SIZE, "chan");

- xpc_sysctl = register_sysctl_table(xpc_sys_dir, 0);
+ xpc_sysctl = register_sysctl_table(xpc_sys_dir);

/*
 * The first few fields of each entry of xpc_partitions[] need to
diff --git a/arch/mips/au1000/common/power.c b/arch/mips/au1000/common/power.c
index 31256b8..3901e8e 100644
--- a/arch/mips/au1000/common/power.c
+++ b/arch/mips/au1000/common/power.c
@@ -461,7 +461,7 @@ static struct ctl_table pm_dir_table[] = {
 */
static int __init pm_init(void)
{
- register_sysctl_table(pm_dir_table, 0);
+ register_sysctl_table(pm_dir_table);
    return 0;
}

diff --git a/arch/mips/lasat/sysctl.c b/arch/mips/lasat/sysctl.c
index c04e82f..699ab18 100644
--- a/arch/mips/lasat/sysctl.c
+++ b/arch/mips/lasat/sysctl.c
@@ -432,7 +432,7 @@ static int __init lasat_register_sysctl(void)
    struct ctl_table_header *lasat_table_header;

    lasat_table_header =
- register_sysctl_table(lasat_root_table, 0);
+ register_sysctl_table(lasat_root_table);

```

```

    return 0;
}
diff --git a/arch/powerpc/kernel/idle.c b/arch/powerpc/kernel/idle.c
index 8b27bb1..6e7f509 100644
--- a/arch/powerpc/kernel/idle.c
+++ b/arch/powerpc/kernel/idle.c
@@ -125,7 +125,7 @@ static ctl_table powersave_nap_sysctl_root[] = {
    static int __init
    register_powersave_nap_sysctl(void)
    {
- register_sysctl_table(powersave_nap_sysctl_root, 0);
+ register_sysctl_table(powersave_nap_sysctl_root);

    return 0;
}
diff --git a/arch/ppc/kernel/ppc_htab.c b/arch/ppc/kernel/ppc_htab.c
index 77b20ff..0a7e42d 100644
--- a/arch/ppc/kernel/ppc_htab.c
+++ b/arch/ppc/kernel/ppc_htab.c
@@ -457,7 +457,7 @@ static ctl_table htab_sysctl_root[] = {
    static int __init
    register_ppc_htab_sysctl(void)
    {
- register_sysctl_table(htab_sysctl_root, 0);
+ register_sysctl_table(htab_sysctl_root);

    return 0;
}
diff --git a/arch/s390/appldata/appldata_base.c b/arch/s390/appldata/appldata_base.c
index cdc4109..b1ff93a 100644
--- a/arch/s390/appldata/appldata_base.c
+++ b/arch/s390/appldata/appldata_base.c
@@ -506,7 +506,7 @@ int appldata_register_ops(struct appldata_ops *ops)

    ops->ctl_table[3].ctl_name = 0;

- ops->sysctl_header = register_sysctl_table(ops->ctl_table,0);
+ ops->sysctl_header = register_sysctl_table(ops->ctl_table);

    P_INFO("%s-ops registered!\n", ops->name);
    return 0;
@@ -606,7 +606,7 @@ static int __init appldata_init(void)
    /* Register cpu hotplug notifier */
    register_hotcpu_notifier(&appldata_nb);

- appldata_sysctl_header = register_sysctl_table(appldata_dir_table, 0);
+ appldata_sysctl_header = register_sysctl_table(appldata_dir_table);
#ifdef MODULE

```

```

    appldata_dir_table[0].de->owner = THIS_MODULE;
    appldata_table[0].de->owner = THIS_MODULE;
diff --git a/arch/s390/kernel/debug.c b/arch/s390/kernel/debug.c
index d38cb27..00f0382 100644
--- a/arch/s390/kernel/debug.c
+++ b/arch/s390/kernel/debug.c
@@ -1053,7 +1053,7 @@ __init debug_init(void)
{
    int rc = 0;

- s390dbf_sysctl_header = register_sysctl_table(s390dbf_dir_table, 0);
+ s390dbf_sysctl_header = register_sysctl_table(s390dbf_dir_table);
    down(&debug_lock);
    debug_debugfs_root_entry = debugfs_create_dir(DEBUG_DIR_ROOT, NULL);
    printk(KERN_INFO "debug: Initialization complete\n");
diff --git a/arch/s390/mm/cmm.c b/arch/s390/mm/cmm.c
index 5f83a3f..8a5c71f 100644
--- a/arch/s390/mm/cmm.c
+++ b/arch/s390/mm/cmm.c
@@ -418,7 +418,7 @@ cmm_init (void)
    int rc = -ENOMEM;

#ifdef CONFIG_CMM_PROC
- cmm_sysctl_header = register_sysctl_table(cmm_dir_table, 0);
+ cmm_sysctl_header = register_sysctl_table(cmm_dir_table);
    if (!cmm_sysctl_header)
        goto out;
#endif
diff --git a/arch/sh64/kernel/traps.c b/arch/sh64/kernel/traps.c
index 02cca74..c346d7e 100644
--- a/arch/sh64/kernel/traps.c
+++ b/arch/sh64/kernel/traps.c
@@ -960,7 +960,7 @@ static ctl_table sh64_root[] = {
    static struct ctl_table_header *sysctl_header;
    static int __init init_sysctl(void)
    {
- sysctl_header = register_sysctl_table(sh64_root, 0);
+ sysctl_header = register_sysctl_table(sh64_root);
        return 0;
    }
}

diff --git a/arch/x86_64/ia32/ia32_binfmt.c b/arch/x86_64/ia32/ia32_binfmt.c
index 644b203..aab2c91 100644
--- a/arch/x86_64/ia32/ia32_binfmt.c
+++ b/arch/x86_64/ia32/ia32_binfmt.c
@@ -418,7 +418,7 @@ static ctl_table abi_root_table2[] = {

    static __init int ia32_binfmt_init(void)

```

```

{
- register_sysctl_table(abi_root_table2, 0);
+ register_sysctl_table(abi_root_table2);
  return 0;
}
__initcall(ia32_binfmt_init);
diff --git a/arch/x86_64/kernel/vsyscall.c b/arch/x86_64/kernel/vsyscall.c
index c0e2b48..313dc6a 100644
--- a/arch/x86_64/kernel/vsyscall.c
+++ b/arch/x86_64/kernel/vsyscall.c
@@ -301,7 +301,7 @@ static int __init vsyscall_init(void)
  BUG_ON((unsigned long) &vgetcpu != VSYSCALL_ADDR(__NR_vgetcpu));
  map_vsyscall();
#ifdef CONFIG_SYSCTL
- register_sysctl_table(kernel_root_table2, 0);
+ register_sysctl_table(kernel_root_table2);
#endif
  on_each_cpu(cpu_vsyscall_init, NULL, 0, 1);
  hotcpu_notifier(cpu_vsyscall_notifier, 0);
diff --git a/arch/x86_64/mm/init.c b/arch/x86_64/mm/init.c
index a04535d..079bcaa 100644
--- a/arch/x86_64/mm/init.c
+++ b/arch/x86_64/mm/init.c
@@ -734,7 +734,7 @@ static ctl_table debug_root_table2[] = {

static __init int x8664_sysctl_init(void)
{
- register_sysctl_table(debug_root_table2, 0);
+ register_sysctl_table(debug_root_table2);
  return 0;
}
__initcall(x8664_sysctl_init);
diff --git a/drivers/cdrom/cdrom.c b/drivers/cdrom/cdrom.c
index 14f72c4..b36f44d 100644
--- a/drivers/cdrom/cdrom.c
+++ b/drivers/cdrom/cdrom.c
@@ -3553,7 +3553,7 @@ static void cdrom_sysctl_register(void)
  if (initialized == 1)
    return;

- cdrom_sysctl_header = register_sysctl_table(cdrom_root_table, 0);
+ cdrom_sysctl_header = register_sysctl_table(cdrom_root_table);

  /* set the defaults */
  cdrom_sysctl_settings.autoclose = autoclose;
diff --git a/drivers/char/hpet.c b/drivers/char/hpet.c
index 81be1db..0be700f 100644
--- a/drivers/char/hpet.c

```

```

+++ b/drivers/char/hpet.c
@@ -1018,7 +1018,7 @@ static int __init hpet_init(void)
    if (result < 0)
        return -ENODEV;

- sysctl_header = register_sysctl_table(dev_root, 0);
+ sysctl_header = register_sysctl_table(dev_root);

    result = acpi_bus_register_driver(&hpet_acpi_driver);
    if (result < 0) {
diff --git a/drivers/char/ipmi/ipmi_poweroff.c b/drivers/char/ipmi/ipmi_poweroff.c
index b3ae65e..e02893b 100644
--- a/drivers/char/ipmi/ipmi_poweroff.c
+++ b/drivers/char/ipmi/ipmi_poweroff.c
@@ -686,7 +686,7 @@ static int ipmi_poweroff_init (void)
    printk(KERN_INFO PFX "Power cycle is enabled.\n");

#ifdef CONFIG_PROC_FS
- ipmi_table_header = register_sysctl_table(ipmi_root_table, 0);
+ ipmi_table_header = register_sysctl_table(ipmi_root_table);
    if (!ipmi_table_header) {
        printk(KERN_ERR PFX "Unable to register powercycle sysctl\n");
        rv = -ENOMEM;
diff --git a/drivers/char/rtc.c b/drivers/char/rtc.c
index df11289..98a2f5e 100644
--- a/drivers/char/rtc.c
+++ b/drivers/char/rtc.c
@@ -316,7 +316,7 @@ static struct ctl_table_header *sysctl_header;

static int __init init_sysctl(void)
{
-    sysctl_header = register_sysctl_table(dev_root, 0);
+    sysctl_header = register_sysctl_table(dev_root);
    return 0;
}

diff --git a/drivers/macintosh/mac_hid.c b/drivers/macintosh/mac_hid.c
index c676740..030399f 100644
--- a/drivers/macintosh/mac_hid.c
+++ b/drivers/macintosh/mac_hid.c
@@ -138,7 +138,7 @@ int __init mac_hid_init(void)
    return err;

#ifdef CONFIG_SYSCTL
-    mac_hid_sysctl_header = register_sysctl_table(mac_hid_root_dir, 0);
+    mac_hid_sysctl_header = register_sysctl_table(mac_hid_root_dir);
#endif /* CONFIG_SYSCTL */

```

```

return 0;
diff --git a/drivers/md/md.c b/drivers/md/md.c
index 966e8be..72b378e 100644
--- a/drivers/md/md.c
+++ b/drivers/md/md.c
@@ -5551,7 +5551,7 @@ static int __init md_init(void)
    md_probe, NULL, NULL);

    register_reboot_notifier(&md_notifier);
- raid_table_header = register_sysctl_table(raid_root_table, 0);
+ raid_table_header = register_sysctl_table(raid_root_table);

    md_geninit();
    return (0);
diff --git a/drivers/net/wireless/arlanc-proc.c b/drivers/net/wireless/arlanc-proc.c
index 20499a6..015abd9 100644
--- a/drivers/net/wireless/arlanc-proc.c
+++ b/drivers/net/wireless/arlanc-proc.c
@@ -1244,7 +1244,7 @@ int __init init_arlanc_proc(void)
    return 0;
    for (i = 0; i < MAX_ARLANS && arlanc_device[i]; i++)
        arlanc_table[i].ctl_name = i + 1;
- arlanc_device_sysctl_header = register_sysctl_table(arlanc_root_table, 0);
+ arlanc_device_sysctl_header = register_sysctl_table(arlanc_root_table);
    if (!arlanc_device_sysctl_header)
        return -1;

diff --git a/drivers/parport/procfs.c b/drivers/parport/procfs.c
index 5337789..afdd392 100644
--- a/drivers/parport/procfs.c
+++ b/drivers/parport/procfs.c
@@ -518,7 +518,7 @@ int parport_proc_register(struct parport *port)
    t->parport_dir[0].child = t->port_dir;
    t->dev_dir[0].child = t->parport_dir;

- t->sysctl_header = register_sysctl_table(t->dev_dir, 0);
+ t->sysctl_header = register_sysctl_table(t->dev_dir);
    if (t->sysctl_header == NULL) {
        kfree(t);
        t = NULL;
@@ -574,7 +574,7 @@ int parport_device_proc_register(struct pardevice *device)
    t->device_dir[0].child = t->vars;
    t->vars[0].data = &device->timeslice;

- t->sysctl_header = register_sysctl_table(t->dev_dir, 0);
+ t->sysctl_header = register_sysctl_table(t->dev_dir);
    if (t->sysctl_header == NULL) {
        kfree(t);

```

```

t = NULL;
@@ -597,7 +597,7 @@ int parport_device_proc_unregister(struct pardevice *device)
static int __init parport_default_proc_register(void)
{
    parport_default_sysctl_table.sysctl_header =
- register_sysctl_table(parport_default_sysctl_table.dev_dir, 0);
+ register_sysctl_table(parport_default_sysctl_table.dev_dir);
    return 0;
}

```

```

diff --git a/drivers/scsi/scsi_sysctl.c b/drivers/scsi/scsi_sysctl.c
index b16b775..6cfaaa2 100644
--- a/drivers/scsi/scsi_sysctl.c
+++ b/drivers/scsi/scsi_sysctl.c
@@ -41,7 +41,7 @@ static struct ctl_table_header *scsi_table_header;

```

```

int __init scsi_init_sysctl(void)
{
- scsi_table_header = register_sysctl_table(scsi_root_table, 0);
+ scsi_table_header = register_sysctl_table(scsi_root_table);
    if (!scsi_table_header)
        return -ENOMEM;
    return 0;

```

```

diff --git a/fs/coda/sysctl.c b/fs/coda/sysctl.c
index df682e2..77d7563 100644
--- a/fs/coda/sysctl.c
+++ b/fs/coda/sysctl.c
@@ -269,7 +269,7 @@ void coda_sysctl_init(void)

```

```

#ifdef CONFIG_SYSCTL
    if ( !fs_table_header )
- fs_table_header = register_sysctl_table(fs_table, 0);
+ fs_table_header = register_sysctl_table(fs_table);
#endif
}

```

```

diff --git a/fs/dquot.c b/fs/dquot.c
index 0952cc4..9a25927 100644
--- a/fs/dquot.c
+++ b/fs/dquot.c
@@ -1822,7 +1822,7 @@ static int __init dquot_init(void)

    printk(KERN_NOTICE "VFS: Disk quotas %s\n", __DQUOT_VERSION__);

- register_sysctl_table(sys_table, 0);
+ register_sysctl_table(sys_table);

    dquot_cachep = kmem_cache_create("dquot",

```

```

    sizeof(struct dquot), sizeof(unsigned long) * 4,
diff --git a/fs/lockd/svc.c b/fs/lockd/svc.c
index 8ca1808..67aa93e 100644
--- a/fs/lockd/svc.c
+++ b/fs/lockd/svc.c
@@ -506,7 +506,7 @@ module_param(nsm_use_hostnames, bool, 0644);

static int __init init_nlm(void)
{
- nlm_sysctl_table = register_sysctl_table(nlm_sysctl_root, 0);
+ nlm_sysctl_table = register_sysctl_table(nlm_sysctl_root);
    return nlm_sysctl_table ? 0 : -ENOMEM;
}

diff --git a/fs/nfs/sysctl.c b/fs/nfs/sysctl.c
index 3ea50ac..fcdcafb 100644
--- a/fs/nfs/sysctl.c
+++ b/fs/nfs/sysctl.c
@@ -75,7 +75,7 @@ static ctl_table nfs_cb_sysctl_root[] = {

int nfs_register_sysctl(void)
{
- nfs_callback_sysctl_table = register_sysctl_table(nfs_cb_sysctl_root, 0);
+ nfs_callback_sysctl_table = register_sysctl_table(nfs_cb_sysctl_root);
    if (nfs_callback_sysctl_table == NULL)
        return -ENOMEM;
    return 0;
diff --git a/fs/ntfs/sysctl.c b/fs/ntfs/sysctl.c
index bc217de..7edb370 100644
--- a/fs/ntfs/sysctl.c
+++ b/fs/ntfs/sysctl.c
@@ -70,7 +70,7 @@ int ntfs_sysctl(int add)
{
    if (add) {
        BUG_ON(sysctls_root_table);
- sysctls_root_table = register_sysctl_table(sysctls_root, 0);
+ sysctls_root_table = register_sysctl_table(sysctls_root);
        if (!sysctls_root_table)
            return -ENOMEM;
#ifdef CONFIG_PROC_FS
diff --git a/fs/ocfs2/cluster/nodemanager.c b/fs/ocfs2/cluster/nodemanager.c
index df763c7..9f5ad0f 100644
--- a/fs/ocfs2/cluster/nodemanager.c
+++ b/fs/ocfs2/cluster/nodemanager.c
@@ -922,7 +922,7 @@ static int __init init_o2nm(void)
    o2hb_init();
    o2net_init();

```

```

- ocfs2_table_header = register_sysctl_table(ocfs2_root_table, 0);
+ ocfs2_table_header = register_sysctl_table(ocfs2_root_table);
  if (!ocfs2_table_header) {
    printk(KERN_ERR "nodemanager: unable to register sysctl\n");
    ret = -ENOMEM; /* or something. */
diff --git a/fs/xfs/linux-2.6/xfs_sysctl.c b/fs/xfs/linux-2.6/xfs_sysctl.c
index 5a0eefc..3ac4dab 100644
--- a/fs/xfs/linux-2.6/xfs_sysctl.c
+++ b/fs/xfs/linux-2.6/xfs_sysctl.c
@@ -251,7 +251,7 @@ STATIC ctl_table xfs_root_table[] = {
  void
  xfs_sysctl_register(void)
  {
- xfs_table_header = register_sysctl_table(xfs_root_table, 0);
+ xfs_table_header = register_sysctl_table(xfs_root_table);
  }

  void
diff --git a/include/linux/sysctl.h b/include/linux/sysctl.h
index c99e4bb..6113f3b 100644
--- a/include/linux/sysctl.h
+++ b/include/linux/sysctl.h
@@ -1034,8 +1034,8 @@ struct ctl_table_header
  struct completion *unregistering;
 };

-struct ctl_table_header * register_sysctl_table(ctl_table * table,
- int insert_at_head);
+struct ctl_table_header * register_sysctl_table(ctl_table * table);
+
  void unregister_sysctl_table(struct ctl_table_header * table);

  #else /* __KERNEL__ */
diff --git a/ipc/ipc_sysctl.c b/ipc/ipc_sysctl.c
index 9018009..a491c60 100644
--- a/ipc/ipc_sysctl.c
+++ b/ipc/ipc_sysctl.c
@@ -175,7 +175,7 @@ static struct ctl_table ipc_root_table[] = {

  static int __init ipc_sysctl_init(void)
  {
- register_sysctl_table(ipc_root_table, 0);
+ register_sysctl_table(ipc_root_table);
    return 0;
  }

diff --git a/ipc/mqueue.c b/ipc/mqueue.c
index 02717f7..00b842a 100644

```

```

--- a/ipc/mqueue.c
+++ b/ipc/mqueue.c
@@ -1255,7 +1255,7 @@ static int __init init_mqueue_fs(void)
    return -ENOMEM;

    /* ignore failues - they are not fatal */
- mq_sysctl_table = register_sysctl_table(mq_sysctl_root, 0);
+ mq_sysctl_table = register_sysctl_table(mq_sysctl_root);

    error = register_filesystem(&mqueue_fs_type);
    if (error)
diff --git a/kernel/sysctl.c b/kernel/sysctl.c
index 2c3703d..5beee1f 100644
--- a/kernel/sysctl.c
+++ b/kernel/sysctl.c
@@ -1240,7 +1240,6 @@ int do_sysctl_strategy (ctl_table *table,
/**
 * register_sysctl_table - register a sysctl hierarchy
 * @table: the top-level table structure
- * @insert_at_head: whether the entry should be inserted in front or at the end
 *
 * Register a sysctl table hierarchy. @table should be a filled in ctl_table
 * array. An entry with a ctl_name of 0 terminates the table.
@@ -1306,8 +1305,7 @@ int do_sysctl_strategy (ctl_table *table,
 * This routine returns %NULL on a failure to register, and a pointer
 * to the table header on success.
 */
-struct ctl_table_header *register_sysctl_table(ctl_table * table,
-      int insert_at_head)
+struct ctl_table_header *register_sysctl_table(ctl_table * table)
{
    struct ctl_table_header *tmp;
    tmp = kmalloc(sizeof(struct ctl_table_header), GFP_KERNEL);
@@ -1318,10 +1316,7 @@ struct ctl_table_header *register_sysctl_table(ctl_table * table,
    tmp->used = 0;
    tmp->unregistering = NULL;
    spin_lock(&sysctl_lock);
- if (insert_at_head)
-     list_add(&tmp->ctl_entry, &root_table_header.ctl_entry);
- else
-     list_add_tail(&tmp->ctl_entry, &root_table_header.ctl_entry);
+ list_add_tail(&tmp->ctl_entry, &root_table_header.ctl_entry);
    spin_unlock(&sysctl_lock);
#ifdef CONFIG_PROC_SYSCTL
    register_proc_table(table, proc_sys_root, tmp);
diff --git a/kernel/utsname_sysctl.c b/kernel/utsname_sysctl.c
index 324aa13..f22b9db 100644
--- a/kernel/utsname_sysctl.c

```

```

+++ b/kernel/utsname_sysctl.c
@@ -139,7 +139,7 @@ static struct ctl_table uts_root_table[] = {

static int __init utsname_sysctl_init(void)
{
- register_sysctl_table(uts_root_table, 0);
+ register_sysctl_table(uts_root_table);
return 0;
}

diff --git a/net/appletalk/sysctl_net_atalk.c b/net/appletalk/sysctl_net_atalk.c
index 4f806b6..7df1778 100644
--- a/net/appletalk/sysctl_net_atalk.c
+++ b/net/appletalk/sysctl_net_atalk.c
@@ -73,7 +73,7 @@ static struct ctl_table_header *atalk_table_header;

void atalk_register_sysctl(void)
{
- atalk_table_header = register_sysctl_table(atalk_root_table, 0);
+ atalk_table_header = register_sysctl_table(atalk_root_table);
}

void atalk_unregister_sysctl(void)
diff --git a/net/ax25/sysctl_net_ax25.c b/net/ax25/sysctl_net_ax25.c
index afdba04..443a836 100644
--- a/net/ax25/sysctl_net_ax25.c
+++ b/net/ax25/sysctl_net_ax25.c
@@ -245,7 +245,7 @@ void ax25_register_sysctl(void)

ax25_dir_table[0].child = ax25_table;

- ax25_table_header = register_sysctl_table(ax25_root_table, 0);
+ ax25_table_header = register_sysctl_table(ax25_root_table);
}

void ax25_unregister_sysctl(void)
diff --git a/net/bridge/br_netfilter.c b/net/bridge/br_netfilter.c
index ea3337a..77998b6 100644
--- a/net/bridge/br_netfilter.c
+++ b/net/bridge/br_netfilter.c
@@ -966,7 +966,7 @@ int br_netfilter_init(void)
}

#ifdef CONFIG_SYSCTL
- brnf_sysctl_header = register_sysctl_table(brnf_net_table, 0);
+ brnf_sysctl_header = register_sysctl_table(brnf_net_table);
if (brnf_sysctl_header == NULL) {
printk(KERN_WARNING

```

```

    "br_netfilter: can't register to sysctl.\n");
diff --git a/net/core/neighbour.c b/net/core/neighbour.c
index e7300b6..8437678 100644
--- a/net/core/neighbour.c
+++ b/net/core/neighbour.c
@@ -2701,7 +2701,7 @@ int neigh_sysctl_register(struct net_device *dev, struct neigh_parms
*p,
    t->neigh_proto_dir[0].child = t->neigh_neigh_dir;
    t->neigh_root_dir[0].child = t->neigh_proto_dir;

- t->sysctl_header = register_sysctl_table(t->neigh_root_dir, 0);
+ t->sysctl_header = register_sysctl_table(t->neigh_root_dir);
    if (!t->sysctl_header) {
        err = -ENOBUFFS;
        goto free_procname;
diff --git a/net/dccp/sysctl.c b/net/dccp/sysctl.c
index 3391631..1260aab 100644
--- a/net/dccp/sysctl.c
+++ b/net/dccp/sysctl.c
@@ -127,7 +127,7 @@ static struct ctl_table_header *dccp_table_header;

int __init dccp_sysctl_init(void)
{
- dccp_table_header = register_sysctl_table(dccp_root_table, 0);
+ dccp_table_header = register_sysctl_table(dccp_root_table);

    return dccp_table_header != NULL ? 0 : -ENOMEM;
}
diff --git a/net/decnet/dn_dev.c b/net/decnet/dn_dev.c
index fc6f3c0..baaa02e 100644
--- a/net/decnet/dn_dev.c
+++ b/net/decnet/dn_dev.c
@@ -282,7 +282,7 @@ static void dn_dev_sysctl_register(struct net_device *dev, struct
dn_dev_parms *
    t->dn_dev_root_dir[0].de = NULL;
    t->dn_dev_vars[0].extra1 = (void *)dev;

- t->sysctl_header = register_sysctl_table(t->dn_dev_root_dir, 0);
+ t->sysctl_header = register_sysctl_table(t->dn_dev_root_dir);
    if (t->sysctl_header == NULL)
        kfree(t);
    else
diff --git a/net/decnet/sysctl_net_decnet.c b/net/decnet/sysctl_net_decnet.c
index 81469fd..37fff9a 100644
--- a/net/decnet/sysctl_net_decnet.c
+++ b/net/decnet/sysctl_net_decnet.c
@@ -491,7 +491,7 @@ static ctl_table dn_root_table[] = {

```

```

void dn_register_sysctl(void)
{
- dn_table_header = register_sysctl_table(dn_root_table, 0);
+ dn_table_header = register_sysctl_table(dn_root_table);
}

void dn_unregister_sysctl(void)
diff --git a/net/ipv4/devinet.c b/net/ipv4/devinet.c
index 480ace9..b731a0c 100644
--- a/net/ipv4/devinet.c
+++ b/net/ipv4/devinet.c
@@ -1603,7 +1603,7 @@ static void devinet_sysctl_register(struct in_device *in_dev,
    t->devinet_root_dir[0].child = t->devinet_proto_dir;
    t->devinet_root_dir[0].de    = NULL;

- t->sysctl_header = register_sysctl_table(t->devinet_root_dir, 0);
+ t->sysctl_header = register_sysctl_table(t->devinet_root_dir);
    if (!t->sysctl_header)
        goto free_procname;

@@ -1637,7 +1637,7 @@ void __init devinet_init(void)
    rtnetlink_links[PF_INET] = inet_rtnetlink_table;
#ifdef CONFIG_SYSCTL
    devinet_sysctl.sysctl_header =
- register_sysctl_table(devinet_sysctl.devinet_root_dir, 0);
+ register_sysctl_table(devinet_sysctl.devinet_root_dir);
    devinet_sysctl_register(NULL, &ipv4_devconf_dflt);
#endif
}
diff --git a/net/ipv4/ipvs/ip_vs_ctl.c b/net/ipv4/ipvs/ip_vs_ctl.c
index 9b93338..c4e4237 100644
--- a/net/ipv4/ipvs/ip_vs_ctl.c
+++ b/net/ipv4/ipvs/ip_vs_ctl.c
@@ -2359,7 +2359,7 @@ int ip_vs_control_init(void)
    proc_net_fops_create("ip_vs", 0, &ip_vs_info_fops);
    proc_net_fops_create("ip_vs_stats", 0, &ip_vs_stats_fops);

- sysctl_header = register_sysctl_table(vs_root_table, 0);
+ sysctl_header = register_sysctl_table(vs_root_table);

    /* Initialize ip_vs_svc_table, ip_vs_svc_fwm_table, ip_vs_rtable */
    for(idx = 0; idx < IP_VS_SVC_TAB_SIZE; idx++) {
diff --git a/net/ipv4/ipvs/ip_vs_lblc.c b/net/ipv4/ipvs/ip_vs_lblc.c
index a4385a2..0e9cdfc 100644
--- a/net/ipv4/ipvs/ip_vs_lblc.c
+++ b/net/ipv4/ipvs/ip_vs_lblc.c
@@ -583,7 +583,7 @@ static struct ip_vs_scheduler ip_vs_lblc_scheduler =
static int __init ip_vs_lblc_init(void)

```

```

{
    INIT_LIST_HEAD(&ip_vs_lblcr_scheduler.n_list);
- sysctl_header = register_sysctl_table(lblcr_root_table, 0);
+ sysctl_header = register_sysctl_table(lblcr_root_table);
    return register_ip_vs_scheduler(&ip_vs_lblcr_scheduler);
}

diff --git a/net/ipv4/ipvs/ip_vs_lblcr.c b/net/ipv4/ipvs/ip_vs_lblcr.c
index fe1af5d..22004f8 100644
--- a/net/ipv4/ipvs/ip_vs_lblcr.c
+++ b/net/ipv4/ipvs/ip_vs_lblcr.c
@@ -841,7 +841,7 @@ static struct ip_vs_scheduler ip_vs_lblcr_scheduler =
static int __init ip_vs_lblcr_init(void)
{
    INIT_LIST_HEAD(&ip_vs_lblcr_scheduler.n_list);
- sysctl_header = register_sysctl_table(lblcr_root_table, 0);
+ sysctl_header = register_sysctl_table(lblcr_root_table);
#ifdef CONFIG_IP_VS_LBLCR_DEBUG
    proc_net_create("ip_vs_lblcr", 0, ip_vs_lblcr_getinfo);
#endif
diff --git a/net/ipv4/netfilter/ip_conntrack_proto_sctp.c
b/net/ipv4/netfilter/ip_conntrack_proto_sctp.c
index 2443322..44513b4 100644
--- a/net/ipv4/netfilter/ip_conntrack_proto_sctp.c
+++ b/net/ipv4/netfilter/ip_conntrack_proto_sctp.c
@@ -623,7 +623,7 @@ static int __init ip_conntrack_proto_sctp_init(void)
}

#ifdef CONFIG_SYSCTL
- ip_ct_sysctl_header = register_sysctl_table(ip_ct_net_table, 0);
+ ip_ct_sysctl_header = register_sysctl_table(ip_ct_net_table);
    if (ip_ct_sysctl_header == NULL) {
        ret = -ENOMEM;
        printk("ip_conntrack_proto_sctp: can't register to sysctl.\n");
diff --git a/net/ipv4/netfilter/ip_conntrack_standalone.c
b/net/ipv4/netfilter/ip_conntrack_standalone.c
index 86efb54..9d89469 100644
--- a/net/ipv4/netfilter/ip_conntrack_standalone.c
+++ b/net/ipv4/netfilter/ip_conntrack_standalone.c
@@ -849,7 +849,7 @@ static int __init ip_conntrack_standalone_init(void)
    goto cleanup_proc_stat;
}
#ifdef CONFIG_SYSCTL
- ip_ct_sysctl_header = register_sysctl_table(ip_ct_net_table, 0);
+ ip_ct_sysctl_header = register_sysctl_table(ip_ct_net_table);
    if (ip_ct_sysctl_header == NULL) {
        printk("ip_conntrack: can't register to sysctl.\n");
        ret = -ENOMEM;

```

diff --git a/net/ipv4/netfilter/ip_queue.c b/net/ipv4/netfilter/ip_queue.c

index cd520df..3446d4a 100644

--- a/net/ipv4/netfilter/ip_queue.c

+++ b/net/ipv4/netfilter/ip_queue.c

```
@@ -693,7 +693,7 @@ static int __init ip_queue_init(void)
{
```

```
    register_netdevice_notifier(&ipq_dev_notifier);
- ipq_sysctl_header = register_sysctl_table(ipq_root_table, 0);
+ ipq_sysctl_header = register_sysctl_table(ipq_root_table);
```

```
    status = nf_register_queue_handler(PF_INET, &nfqh);
    if (status < 0) {
```

diff --git a/net/ipv6/addrconf.c b/net/ipv6/addrconf.c

index 171e5b5..791aaba 100644

--- a/net/ipv6/addrconf.c

+++ b/net/ipv6/addrconf.c

```
@@ -4004,7 +4004,7 @@ static void addrconf_sysctl_register(struct inet6_dev *idev, struct
ipv6_devconf
```

```
    t->addrconf_root_dir[0].child = t->addrconf_proto_dir;
    t->addrconf_root_dir[0].de = NULL;
```

```
- t->sysctl_header = register_sysctl_table(t->addrconf_root_dir, 0);
+ t->sysctl_header = register_sysctl_table(t->addrconf_root_dir);
    if (t->sysctl_header == NULL)
        goto free_procname;
    else
```

```
@@ -4089,7 +4089,7 @@ int __init addrconf_init(void)
```

```
    rtnetlink_links[PF_INET6] = inet6_rtnetlink_table;
```

```
#ifdef CONFIG_SYSCTL
```

```
    addrconf_sysctl.sysctl_header =
```

```
- register_sysctl_table(addrconf_sysctl.addrconf_root_dir, 0);
```

```
+ register_sysctl_table(addrconf_sysctl.addrconf_root_dir);
```

```
    addrconf_sysctl_register(NULL, &ipv6_devconf_dflt);
```

```
#endif
```

diff --git a/net/ipv6/netfilter/ip6_queue.c b/net/ipv6/netfilter/ip6_queue.c

index d4d9f18..e774be7 100644

--- a/net/ipv6/netfilter/ip6_queue.c

+++ b/net/ipv6/netfilter/ip6_queue.c

```
@@ -683,7 +683,7 @@ static int __init ip6_queue_init(void)
{
```

```
    register_netdevice_notifier(&ipq_dev_notifier);
- ipq_sysctl_header = register_sysctl_table(ipq_root_table, 0);
+ ipq_sysctl_header = register_sysctl_table(ipq_root_table);
```

```
    status = nf_register_queue_handler(PF_INET6, &nfqh);
```

```

    if (status < 0) {
diff --git a/net/ipv6/sysctl_net_ipv6.c b/net/ipv6/sysctl_net_ipv6.c
index 7a4639d..31e2494 100644
--- a/net/ipv6/sysctl_net_ipv6.c
+++ b/net/ipv6/sysctl_net_ipv6.c
@@ -107,7 +107,7 @@ static ctl_table ipv6_root_table[] = {

void ipv6_sysctl_register(void)
{
- ipv6_sysctl_header = register_sysctl_table(ipv6_root_table, 0);
+ ipv6_sysctl_header = register_sysctl_table(ipv6_root_table);
}

void ipv6_sysctl_unregister(void)
diff --git a/net/ipx/sysctl_net_ipx.c b/net/ipx/sysctl_net_ipx.c
index 0442f44..55a7d96 100644
--- a/net/ipx/sysctl_net_ipx.c
+++ b/net/ipx/sysctl_net_ipx.c
@@ -52,7 +52,7 @@ static struct ctl_table_header *ipx_table_header;

void ipx_register_sysctl(void)
{
- ipx_table_header = register_sysctl_table(ipx_root_table, 0);
+ ipx_table_header = register_sysctl_table(ipx_root_table);
}

void ipx_unregister_sysctl(void)
diff --git a/net/irda/irsysctl.c b/net/irda/irsysctl.c
index 86805c3..b60b72f 100644
--- a/net/irda/irsysctl.c
+++ b/net/irda/irsysctl.c
@@ -274,7 +274,7 @@ static struct ctl_table_header *irda_table_header;
 */
int __init irda_sysctl_register(void)
{
- irda_table_header = register_sysctl_table(irda_root_table, 0);
+ irda_table_header = register_sysctl_table(irda_root_table);
  if (!irda_table_header)
    return -ENOMEM;

diff --git a/net/llc/sysctl_net_llc.c b/net/llc/sysctl_net_llc.c
index 4aab676..154a3e6 100644
--- a/net/llc/sysctl_net_llc.c
+++ b/net/llc/sysctl_net_llc.c
@@ -116,7 +116,7 @@ static struct ctl_table_header *llc_table_header;

int __init llc_sysctl_init(void)
{

```

```

- llc_table_header = register_sysctl_table(llc_root_table, 0);
+ llc_table_header = register_sysctl_table(llc_root_table);

    return llc_table_header ? 0 : -ENOMEM;
}
diff --git a/net/netfilter/nf_conntrack_standalone.c b/net/netfilter/nf_conntrack_standalone.c
index f1cb60f..2587b49 100644
--- a/net/netfilter/nf_conntrack_standalone.c
+++ b/net/netfilter/nf_conntrack_standalone.c
@@ -445,7 +445,7 @@ static int __init nf_conntrack_standalone_init(void)
    proc_stat->owner = THIS_MODULE;
#endif
#ifdef CONFIG_SYSCTL
- nf_ct_sysctl_header = register_sysctl_table(nf_ct_net_table, 0);
+ nf_ct_sysctl_header = register_sysctl_table(nf_ct_net_table);
    if (nf_ct_sysctl_header == NULL) {
        printk("nf_conntrack: can't register to sysctl.\n");
        ret = -ENOMEM;
diff --git a/net/netfilter/nf_sysctl.c b/net/netfilter/nf_sysctl.c
index 06ddddb..ee34589 100644
--- a/net/netfilter/nf_sysctl.c
+++ b/net/netfilter/nf_sysctl.c
@@ -56,7 +56,7 @@ nf_register_sysctl_table(struct ctl_table *path, struct ctl_table *table)
    path = path_dup(path, table);
    if (path == NULL)
        return NULL;
- header = register_sysctl_table(path, 0);
+ header = register_sysctl_table(path);
    if (header == NULL)
        path_free(path, table);
    return header;
diff --git a/net/netrom/sysctl_net_netrom.c b/net/netrom/sysctl_net_netrom.c
index 09f4246..4cbc309 100644
--- a/net/netrom/sysctl_net_netrom.c
+++ b/net/netrom/sysctl_net_netrom.c
@@ -192,7 +192,7 @@ static ctl_table nr_root_table[] = {

void __init nr_register_sysctl(void)
{
- nr_table_header = register_sysctl_table(nr_root_table, 0);
+ nr_table_header = register_sysctl_table(nr_root_table);
}

void nr_unregister_sysctl(void)
diff --git a/net/rose/sysctl_net_rose.c b/net/rose/sysctl_net_rose.c
index 0190a07..c30e1c7 100644
--- a/net/rose/sysctl_net_rose.c
+++ b/net/rose/sysctl_net_rose.c

```

```

@@ -160,7 +160,7 @@ static ctl_table rose_root_table[] = {

void __init rose_register_sysctl(void)
{
- rose_table_header = register_sysctl_table(rose_root_table, 0);
+ rose_table_header = register_sysctl_table(rose_root_table);
}

void rose_unregister_sysctl(void)
diff --git a/net/rxrpc/sysctl.c b/net/rxrpc/sysctl.c
index 6374df7..3bad91a 100644
--- a/net/rxrpc/sysctl.c
+++ b/net/rxrpc/sysctl.c
@@ -97,7 +97,7 @@ static ctl_table rxrpc_dir_sysctl_table[] = {
int rxrpc_sysctl_init(void)
{
#ifdef CONFIG_SYSCTL
- rxrpc_sysctl = register_sysctl_table(rxrpc_dir_sysctl_table, 0);
+ rxrpc_sysctl = register_sysctl_table(rxrpc_dir_sysctl_table);
if (!rxrpc_sysctl)
return -ENOMEM;
#endif /* CONFIG_SYSCTL */
diff --git a/net/sctp/sysctl.c b/net/sctp/sysctl.c
index 633cd17..e2c679b 100644
--- a/net/sctp/sysctl.c
+++ b/net/sctp/sysctl.c
@@ -254,7 +254,7 @@ static struct ctl_table_header * sctp_sysctl_header;
/* Sysctl registration. */
void sctp_sysctl_register(void)
{
- sctp_sysctl_header = register_sysctl_table(sctp_root_table, 0);
+ sctp_sysctl_header = register_sysctl_table(sctp_root_table);
}

/* Sysctl deregistration. */
diff --git a/net/sunrpc/sysctl.c b/net/sunrpc/sysctl.c
index 6a82ed2..df70c8c 100644
--- a/net/sunrpc/sysctl.c
+++ b/net/sunrpc/sysctl.c
@@ -36,7 +36,7 @@ void
rpc_register_sysctl(void)
{
if (!sunrpc_table_header)
- sunrpc_table_header = register_sysctl_table(sunrpc_table, 0);
+ sunrpc_table_header = register_sysctl_table(sunrpc_table);
}

void

```

```

diff --git a/net/sunrpc/xprtsock.c b/net/sunrpc/xprtsock.c
index 51964cf..a314c86 100644
--- a/net/sunrpc/xprtsock.c
+++ b/net/sunrpc/xprtsock.c
@@ -1630,7 +1630,7 @@ int init_socket_xprt(void)
{
#ifdef RPC_DEBUG
if (!sunrpc_table_header)
- sunrpc_table_header = register_sysctl_table(sunrpc_table, 0);
+ sunrpc_table_header = register_sysctl_table(sunrpc_table);
#endif

return 0;
diff --git a/net/unix/sysctl_net_unix.c b/net/unix/sysctl_net_unix.c
index 690ffa5..eb0bd57 100644
--- a/net/unix/sysctl_net_unix.c
+++ b/net/unix/sysctl_net_unix.c
@@ -50,7 +50,7 @@ static struct ctl_table_header * unix_sysctl_header;

void unix_sysctl_register(void)
{
- unix_sysctl_header = register_sysctl_table(unix_root_table, 0);
+ unix_sysctl_header = register_sysctl_table(unix_root_table);
}

void unix_sysctl_unregister(void)
diff --git a/net/x25/sysctl_net_x25.c b/net/x25/sysctl_net_x25.c
index 94aff67..324ae1c 100644
--- a/net/x25/sysctl_net_x25.c
+++ b/net/x25/sysctl_net_x25.c
@@ -98,7 +98,7 @@ static struct ctl_table x25_root_table[] = {

void __init x25_register_sysctl(void)
{
- x25_table_header = register_sysctl_table(x25_root_table, 0);
+ x25_table_header = register_sysctl_table(x25_root_table);
}

void x25_unregister_sysctl(void)
--
1.4.4.1.g278f

```

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