
Subject: [PATCH net-2.6.25 2/2][INET] Eliminate difference in actions of sysctl and proc handler for conf.all

Posted by [Pavel Emelianov](#) on Mon, 03 Dec 2007 14:40:48 GMT

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

AFAIS the net.ipv4.conf. <dev>, all and default sysctls should work like this when changed (besides changing the value itself):

<dev> : optionally do smth else
all : walk devices
default : walk devices

The proc handler for net.ipv4.conf.all works like this:

<dev> : flush rt cache
all : walk devices and flush rt cache
default : nothing

while the sysctl handler works like this:

<dev> : nothing
all : nothing
default : walk devices but don't flush the cache

All this looks strange. Am I right that regardless of whatever handler (proc or syscall) is called the behavior should be:

<dev> : flush rt cache
all : walk the devices and flush the cache
default : walk the devices and flush the cache

?

If yes, then this patch should fix it up.

Signed-off-by: Pavel Emelyanov <xemul@openvz.org>

```
diff --git a/net/ipv4/devinet.c b/net/ipv4/devinet.c
index 0b5f042..1934a06 100644
--- a/net/ipv4/devinet.c
+++ b/net/ipv4/devinet.c
@@ -1282,6 +1282,17 @@ static void inet_forward_change(void)
     rcu_read_unlock();
 }
 read_unlock(&dev_base_lock);
+}
```

```

+
+static void fixup_forward_change(struct ctl_table *table)
+{
+ struct ipv4_devconf *cnf;
+
+ conf = table->extra1;
+ if (conf == &ipv4_devconf)
+ inet_forward_change();
+ else if (conf == &ipv4_devconf_dflt)
+ devinet_copy_dflt_conf(NET_IPV4_CONF_FORWARDING - 1);

    rt_cache_flush(0);
}
@@ -1305,9 +1316,9 @@ static int devinet_conf_proc(ctl_table *ctl, int write,
    return ret;
}

-static int devinet_conf_sysctl(ctl_table *table, int __user *name, int nlen,
+static int __devinet_conf_sysctl(ctl_table *table, int __user *name, int nlen,
    void __user *oldval, size_t __user *oldlenp,
-    void __user *newval, size_t newlen)
+    void __user *newval, size_t newlen, int *idx)
{
    struct ipv4_devconf *cnf;
    int *valp = table->data;
@@ -1346,16 +1357,27 @@ static int devinet_conf_sysctl(ctl_table *table, int __user *name, int
nlen,

    cnf = table->extra1;
    i = (int *)table->data - cnf->data;
-
    set_bit(i, cnf->state);
+ *idx = i;
+ return 1;
+}
+
+static int devinet_conf_sysctl(ctl_table *table, int __user *name, int nlen,
+    void __user *oldval, size_t __user *oldlenp,
+    void __user *newval, size_t newlen)
+{
+ int ret, i;

- if (cnf == &ipv4_devconf_dflt)
+ if (cnf == &ipv4_devconf_dflt)
+ ret = __devinet_conf_sysctl(table, name, nlen, oldval, oldlenp,
+ newval, newlen, &i);
+
+ if (ret == 1 && table->extra1 == &ipv4_devconf_dflt)
    devinet_copy_dflt_conf(i);

```

```

- return 1;
+ return ret;
}

-static int devinet_sysctl_forward(ctl_table *ctl, int write,
+static int devinet_forward_proc(ctl_table *ctl, int write,
    struct file* filp, void __user *buffer,
    size_t *lenp, loff_t *ppos)
{
@@ -1363,16 +1385,25 @@ static int devinet_sysctl_forward(ctl_table *ctl, int write,
    int val = *valp;
    int ret = proc_dointvec(ctl, write, filp, buffer, lenp, ppos);

- if (write && *valp != val) {
- if (valp == &IPV4_DEVCONF_ALL(FORWARDING))
- inet_forward_change();
- else if (valp != &IPV4_DEVCONF_DFLT(FORWARDING))
- rt_cache_flush(0);
- }
+ if (write && *valp != val)
+ fixup_forward_change(ctl);

    return ret;
}

+static int devinet_forward_sysctl(ctl_table *table, int __user *name, int nlen,
+    void __user *oldval, size_t __user *oldlenp,
+    void __user *newval, size_t newlen)
+{
+ int ret, i;
+
+ ret = __devinet_conf_sysctl(table, name, nlen, oldval, oldlenp,
+    newval, newlen, &i);
+ if (ret == 1)
+ fixup_forward_change(table);
+ return ret;
+}
+
+int ipv4_doint_and_flush(ctl_table *ctl, int write,
+    struct file* filp, void __user *buffer,
+    size_t *lenp, loff_t *ppos)
@@ -1436,8 +1467,8 @@ static struct devinet_sysctl_table {
} devinet_sysctl = {
    .devinet_vars = {
        DEVINET_SYSCTL_COMPLEX_ENTRY(FORWARDING, "forwarding",
-            devinet_sysctl_forward,
-            devinet_conf_sysctl),

```

```

+     devinet_forward_proc,
+     devinet_forward_sysctl),
  DEVINET_SYSCTL_RO_ENTRY(MC_FORWARDING, "mc_forwarding"),

  DEVINET_SYSCTL_RW_ENTRY(ACCEPT_REDIRECTS, "accept_redirects"),
@@ -1545,8 +1576,8 @@ static struct ctl_table ctl_forward_entry[] = {
    NET_IPV4_CONF_FORWARDING - 1],
    .maxlen = sizeof(int),
    .mode = 0644,
-   .proc_handler = devinet_sysctl_forward,
-   .strategy = devinet_conf_sysctl,
+   .proc_handler = devinet_forward_proc,
+   .strategy = devinet_forward_sysctl,
    .extra1 = &ipv4_devconf,
  },
  { },

```
