
Subject: [PATCH 01/10] sysctl: Update sysctl_check_table
Posted by [ebiederm](#) on Fri, 10 Aug 2007 00:49:46 GMT

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

Well it turns out after I dug into the problems a little more I was returning a few false positives so this patch updates my logic to remove them.

- Don't complain about 0 ctl_names in sysctl_check_binary_path
It is valid for someone to remove the sysctl binary interface and still keep the same sysctl proc interface.
- Count ctl_names and procnames as matching if they both don't exist.
- Only warn about missing min&max when the generic functions care.

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

kernel/sysctl_check.c | 30 ++++++-----
1 files changed, 16 insertions(+), 14 deletions(-)

```
diff --git a/kernel/sysctl_check.c b/kernel/sysctl_check.c
index 389c4ba..930a514 100644
--- a/kernel/sysctl_check.c
+++ b/kernel/sysctl_check.c
@@ -1420,12 +1420,14 @@ static int sysctl_check_dir(struct ctl_table *table)
    ref = sysctl_check_lookup(table);
    if (ref) {
        int match = 0;
-       if (table->procname && ref->procname &&
-           (strcmp(table->procname, ref->procname) == 0))
+       if ((!table->procname && !ref->procname) ||
+           (table->procname && ref->procname &&
+           (strcmp(table->procname, ref->procname) == 0)))
            match++;
-       if (table->ctl_name && ref->ctl_name &&
-           (table->ctl_name == ref->ctl_name))
+       if ((!table->ctl_name && !ref->ctl_name) ||
+           (table->ctl_name && ref->ctl_name &&
+           (table->ctl_name == ref->ctl_name)))
            match++;
        if (match != 2) {
@@ -1462,8 +1464,8 @@ static void sysctl_check_bin_path(struct ctl_table *table, const char
**fail)
        (strcmp(table->procname, ref->procname) != 0)))
```

```

set_fail(fail, table, "procname does not match binary path procname");

- if (ref->ctl_name &&
-     (!table->ctl_name || table->ctl_name != ref->ctl_name))
+ if (ref->ctl_name && table->ctl_name &&
+     (table->ctl_name != ref->ctl_name))
    set_fail(fail, table, "ctl_name does not match binary path ctl_name");
}
}

@@ -1499,7 +1501,7 @@ int sysctl_check_table(struct ctl_table *table)
    if (table->extra2)
        set_fail(&fail, table, "Directory with extra2");
    if (sysctl_check_dir(table))
-    set_fail(&fail, table, "Inconsistent directory");
+    set_fail(&fail, table, "Inconsistent directory names");
} else {
    if ((table->strategy == sysctl_data) ||
        (table->strategy == sysctl_string) ||
@@ -1520,14 +1522,14 @@ int sysctl_check_table(struct ctl_table *table)
    if (!table->maxlen)
        set_fail(&fail, table, "No maxlen");
}
- if ((table->strategy == sysctl_intvec) ||
-     (table->proc_handler == proc_dointvec_minmax) ||
-     (table->proc_handler == proc_doulongvec_minmax) ||
+ if ((table->proc_handler == proc_doulongvec_minmax) ||
+     (table->proc_handler == proc_doulongvec_ms_jiffies_minmax)) {
-     if (!table->extra1)
-         set_fail(&fail, table, "No min");
-     if (!table->extra2)
-         set_fail(&fail, table, "No max");
+     if (table->maxlen > sizeof (unsigned long)) {
+         if (!table->extra1)
+             set_fail(&fail, table, "No min");
+         if (!table->extra2)
+             set_fail(&fail, table, "No max");
+     }
}
if (table->ctl_name && !table->strategy)
    set_fail(&fail, table, "Missing strategy");
--
```

1.5.1.1.181.g2de0
