Subject: [PATCH] sysctl_{,ms_}jiffies: fix oldlen semantics
Posted by adobriyan on Tue, 12 Dec 2006 16:14:12 GMT
View Forum Message <> Reply to Message

currently it's
1) if *oldlenp == 0,
 don't writeback anything

2) if *oldlenp >= table->maxlen,
 don't writeback more than table->maxlen bytes and rewrite *oldlenp
 don't look at underlying type granularity

3) if 0 < *oldlenp < table->maxlen,
  cough
 string sysctls don't writeback more than *oldlenp bytes.
 OK, that's because sizeof(char) == 1

 int sysctls writeback anything in (0, table->maxlen] range
 Though accept integers divisible by sizeof(int) for writing.

sysctl_jiffies and sysctl_ms_jiffies don't writeback anything but
sizeof(int), which violates 1) and 2).

So, make sysctl_jiffies and sysctl_ms_jiffies accept
a) *oldlenp == 0, not doing writeback
b) *oldlenp >= sizeof(int), writing one integer.

-EINVAL still returned for *oldlenp == 1, 2, 3.

Signed-off-by: Alexey Dobriyan <adobriyan@openvz.org>
---

 kernel/sysctl.c | 44 ++++++++++++++++++++++++++++++++++----------
 1 file changed, 28 insertions(+), 16 deletions(-)

--- a/kernel/sysctl.c
+++ b/kernel/sysctl.c
@@ -2752,17 +2752,23 @@ int sysctl_jiffies(ctl_table *table, int
   void __user *oldval, size_t __user *oldlenp,
   void __user *newval, size_t newlen, void **context)
 {
- if (oldval) {
+ if (oldval && oldlenp) {
   size_t olen;
- if (oldlenp) {
-   if (get_user(olen, oldlenp))
+
+ if (get_user(olen, oldlenp))

```
+   return -EFAULT;
+  if (olen) {
+   int val;
+
+   if (olen < sizeof(int))
+    return -EINVAL;
+
+   val = *(int *)(table->data) / HZ;
+   if (put_user(val, (int __user *)oldval))
+    return -EFAULT;
+   if (put_user(sizeof(int), oldlenp))
     return -EFAULT;
-   if (olen!=sizeof(int))
-    return -EINVAL;
   }
-  if (put_user(*(int *)(table->data)/HZ, (int __user *)oldval) ||
-      (oldlenp && put_user(sizeof(int),oldlenp)))
-   return -EFAULT;
  }
  if (newval && newlen) {
   int new;
@@ -2780,17 +2786,23 @@ int sysctl_ms_jiffies(ctl_table *table,
   void __user *oldval, size_t __user *oldlenp,
   void __user *newval, size_t newlen, void **context)
 {
- if (oldval) {
+ if (oldval && oldlenp) {
   size_t olen;
-  if (oldlenp) {
-   if (get_user(olen, oldlenp))
+
+  if (get_user(olen, oldlenp))
+   return -EFAULT;
+  if (olen) {
+   int val;
+
+   if (olen < sizeof(int))
+    return -EINVAL;
+
+   val = jiffies_to_msecs(*(int *)(table->data));
+   if (put_user(val, (int __user *)oldval))
+    return -EFAULT;
+   if (put_user(sizeof(int), oldlenp))
     return -EFAULT;
-   if (olen!=sizeof(int))
-    return -EINVAL;
   }
-  if (put_user(jiffies_to_msecs(*(int *)(table->data)), (int __user *)oldval) ||
```

```
-      (oldlenp && put_user(sizeof(int),oldlenp)))
-    return -EFAULT;
   }
   if (newval && newlen) {
    int new;
```