

[PATCH] Fix user data corrupted by old value return of sysctl

Source: <http://linux.derkeiler.com/Mailing-Lists/Kernel/2005-12/msg08351.html>

- *From:* Yi Yang <yang.y.yi@xxxxxxxxxx>
 - *Date:* Fri, 30 Dec 2005 16:40:39 +0800
-

If the user reads a sysctl entry which is of string type by sysctl syscall, this call probably corrupts the user data right after the old value buffer, the issue lies in sysctl_string setting 0 to oldval[len], len is the available buffer size specified by the user, obviously, this will write to the first byte of the user memory place immediate after the old value buffer, the correct way is that sysctl_string doesn't set 0, the user should do it by self in the program.

The following program verifies this point:

```
#include <linux/unistd.h>
#include <linux/types.h>
#include <linux/sysctl.h>
#include <errno.h>

_syscall1(int, _sysctl, struct __sysctl_args *, args);
int sysctl(int *name, int nlen, void *oldval, size_t *oldlenp,
          void *newval, size_t newlen)
{
    struct __sysctl_args args
        = {name, nlen, oldval, oldlenp, newval, newlen};

    return _sysctl(&args);
}

#define SIZE(x) sizeof(x)/sizeof(x[0])
#define OSNAMESZ 4

struct mystruct {
```

[PATCH] Fix user data corrupted by old value return of sysctl

```
char osname[OSNAMESZ];
int target;
int osnamelth;
} myos;

int name[] = { CTL_KERN, KERN_NODENAME };

int main(int argc, char * argv[])
{
    myos.target = 1;
    printf("target = %d\n", myos.target);
    myos.osnamelth = SIZE(myos.osname);
    if (sysctl(name, SIZE(name), myos.osname,
              &myos.osnamelth, 0, 0))
        perror("sysctl");
    else {
        printf("Current host name: %s\n", myos.osname);
    }
    printf("target = %d\n", myos.target);
    return 0;
}
```

Copy it to file sysctl-safe.c, then

```
$ hostname
mylocalmachine
$ gcc sysctl-safe.c
$ ./a.out
target = 1
Current host name: mylo
target = 0
$
```

After apply this patch:

```
$ hostname
mylocalmachine
$ gcc sysctl-safe.c
$ ./a.out
target = 1
Current host name: mylo
target = 1
```

Signed-off-by: Yi Yang <yang.y.yi@xxxxxxxxxx>

```
--- a/kernel/sysctl.c.orig      2005-12-30 09:21:34.000000000 +0000
+++ b/kernel/sysctl.c          2005-12-30 15:58:15.000000000 +0000
```

[PATCH] Fix user data corrupted by old value return of sysctl

```
@@ -2207,8 +2207,6 @@ int sysctl_string(ctl_table *table, int
                        len = table->maxlen;
                        if(copy_to_user(oldval, table->data, len))
                            return -EFAULT;
-
-                        if(put_user(0, ((char __user *) oldval) + len))
                            return -EFAULT;
                        if(put_user(len, oldlenp))
                            return -EFAULT;
                    }
```

-
To unsubscribe from this list: send the line "unsubscribe linux-kernel" in
the body of a message to majordomo@xxxxxxxxxxxxxxxxxxxxx
More majordomo info at <http://vger.kernel.org/majordomo-info.html>
Please read the FAQ at <http://www.tux.org/lkml/>