

Re: [PATCH 2.6.20-rc2] [BUGFIX] drivers/atm/firestream.c: Fix infinite recursion when alignment passed is 0.

Re: [PATCH 2.6.20-rc2] [BUGFIX] drivers/atm/firestream.c: Fix infinite recursion when alignment passed is 0.

Source: <http://linux.derkeiler.com/Mailing-Lists/Kernel/2006-12/msg08087.html>

- *From:* Mitchell Blank Jr <mitch@xxxxxxxxxx>
 - *Date:* Sat, 30 Dec 2006 21:59:07 -0800
-

First, if you want to get patches merged you should send them to the subsystem maintained (in this case Chas, who I've cc:'ed) Also if you feel it needs to be sent to mailing list you should usually use a more specific list first (like the ATM list or maybe netdev) Please see Documentation/SubsubmittingPatches

Amit Choudhary wrote:

Description: Fix infinite recursion when alignment passed is 0 in function aligned_kmalloc(),

I'm curious how you hit this -- the only caller to aligned_kmalloc() passes a constant "0x10"

Looking at aligned_kmalloc() it seems to be pretty badly broken -- its fallback if it gets a non-aligned buffer is to just try a larger size which doesn't necessarily fix the problem. It looks like explicitly aligning the buffer is a better solution.

Could you test this patch? If it works feel free to forward it on to Chas.

-Mitch

[PATCH] [ATM] remove firestream.c's aligned_kmalloc()

Signed-off-by: Mitchell Blank Jr <mitch@xxxxxxxxxx>

```
diff --git a/drivers/atm/firestream.c b/drivers/atm/firestream.c
index 9c67df5..df8b0c0 100644
--- a/drivers/atm/firestream.c
+++ b/drivers/atm/firestream.c
@@ -1379,38 +1379,22 @@ static void reset_chip (struct fs_dev *d
 }
 }
```

```
-static void __devinit *aligned_kmalloc (int size, gfp_t flags, int alignment)
- {
```

Re: [PATCH 2.6.20-rc2] [BUGFIX] drivers/atm/firestream.c: Fix infinite recursion when alignment passed is

Re: [PATCH 2.6.20-rc2] [BUGFIX] drivers/atm/firestream.c: Fix infinite recursion when alignment passed is 0.

```
- void *t;
-
- if (alignment <= 0x10) {
- t = kmalloc (size, flags);
- if ((unsigned long)t & (alignment-1)) {
- printk ("Kmalloc doesn't align things correctly! %p\n", t);
- kfree (t);
- return aligned_kmalloc (size, flags, alignment * 4);
- }
- return t;
- }
- printk (KERN_ERR "Request for > 0x10 alignment not yet implemented (hard!)\n");
- return NULL;
-}
-
static int __devinit init_q (struct fs_dev *dev,
struct queue *txq, int queue, int nentries, int is_rq)
{
- int sz = nentries * sizeof (struct FS_QENTRY);
+ unsigned sz = nentries * sizeof (struct FS_QENTRY);
struct FS_QENTRY *p;
+ void *vp;

func_enter ();

fs_dprintk (FS_DEBUG_INIT, "Inititing queue at %x: %d entries:\n",
queue, nentries);
-
- p = aligned_kmalloc (sz, GFP_KERNEL, 0x10);
+ vp = kmalloc (sz + 0xF, GFP_KERNEL);
+ p = (struct FS_QENTRY *) ALIGN((unsigned long) vp, 0x10);
fs_dprintk (FS_DEBUG_ALLOC, "Alloc queue: %p(%d)\n", p, sz);

- if (!p) return 0;
+ if (!vp) return 0;

write_fs (dev, Q_SA(queue), virt_to_bus(p));
write_fs (dev, Q_EA(queue), virt_to_bus(p+nentries-1));
@@ -1423,8 +1407,7 @@ static int __devinit init_q (struct fs_d
write_fs (dev, Q_CNF(queue), 0 );
}

- txq->sa = p;
- txq->ea = p;
+ txq->buf = vp;
txq->offset = queue;

func_exit ();
@@ -1529,8 +1512,8 @@ static void __devexit free_queue (struct
write_fs (dev, Q_WP(txq->offset), 0);
/* Configuration ? */
```

Re: [PATCH 2.6.20-rc2] [BUGFIX] drivers/atm/firestream.c: Fix infinite recursion when alignment passed is

Re: [PATCH 2.6.20-rc2] [BUGFIX] drivers/atm/firestream.c: Fix infinite recursion when alignment passed is 0.

```
- fs_dprintk (FS_DEBUG_ALLOC, "Free queue: %p\n", txq->sa);
- kfree (txq->sa);
+ fs_dprintk (FS_DEBUG_ALLOC, "Free queue: %p\n", txq->buf);
+ kfree (txq->buf);
```

```
func_exit ();
```

```
}
```

```
diff --git a/drivers/atm/firestream.h b/drivers/atm/firestream.h
```

```
index 49e783e..5408766 100644
```

```
--- a/drivers/atm/firestream.h
```

```
+++ b/drivers/atm/firestream.h
```

```
@@ -455,7 +455,7 @@ struct fs_vcc {
```

```
struct queue {
```

```
- struct FS_QENTRY *sa, *ea;
```

```
+ void *buf;
```

```
int offset;
```

```
};
```

-

To unsubscribe from this list: send the line "unsubscribe linux-kernel" in the body of a message to majordomo@xxxxxxxxxxxxxxxxxxx

More majordomo info at <http://vger.kernel.org/majordomo-info.html>

Please read the FAQ at <http://www.tux.org/lkml/>

Re: [PATCH 2.6.20-rc2] [BUGFIX] drivers/atm/firestream.c: Fix infinite recursion when alignment passed is 0.