

## [PATCH] Use after free in drivers/media/video/videodev.c

*Source:* <http://linux.derkeiler.com/Mailing-Lists/Kernel/2003-08/8137.html>

---

**From:** Kronos (*kronos\_at\_kronoz.cjb.net*)

**Date:** 08/30/03

Date: Sat, 30 Aug 2003 21:55:29 +0200

To: linux-kernel@vger.kernel.org

Hi,

I think that there's a bug in videodev.c. Look at  
video\_unregister\_device:

```
void video_unregister_device(struct video_device *vfd) {
    [...]
    class_device_unregister(&vfd->class_dev);
    devfs_remove(vfd->devfs_name);
    video_device[vfd->minor]=NULL;
}
```

The class\_device\_unregister will call video\_release. This function will call a ->release callback. As far as I can see drivers do their own cleanup outside video\_unregister\_device so there is no problem.

However, if a driver switch to dynamically allocated video\_device this ->release callback will free the struct video\_device (look at video\_device\_release) and possibly its container. So after class\_device\_unregister vfd may be a pointer to deallocated memory.

I think that class\_device\_unregister should be moved down:

--- 2.6.0.orig/drivers/media/video/videodev.c Tue Aug 12 17:02:29 2003

+++ 2.6.0/drivers/media/video/videodev.c Sat Aug 30 21:13:29 2003

@@ -349,9 +349,9 @@

```
    if(video_device[vfd->minor]!=vfd)
        panic("videodev: bad unregister");
```

```
- class_device_unregister(&vfd->class_dev);
  devfs_remove(vfd->devfs_name);
  video_device[vfd->minor]=NULL;
+ class_device_unregister(&vfd->class_dev);
  up(&videodev_lock);
}
```

Linux-Kernel: [PATCH] Use after free in drivers/media/video/videodev.c

Luca

--

Reply-To: kronos@kronoz.cjb.net

Home: <http://kronoz.cjb.net>

The trouble with computers is that they do what you tell them,  
not what you want.

D. Cohen

-

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

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

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