

Why your OpenGL screensavers no longer work...(Nvidia Driver-FC1)

Source: <http://linux.derkeiler.com/Newsgroups/alt.os.linux.redhat/2004-05/0226.html>

From: Jim (jimod_at_jimod.plus.com)

Date: 05/11/04

Date: Tue, 11 May 2004 11:08:07 +0100

Nvidia's proprietary video drivers do not come in RPM package format, although they did at one point in time, I'm not sure when they stopped supplying rpm packages for their drivers. Instead there is a binary installer that you run and it unpacks the archive and drops the driver files into the distribution outside of rpm context.

In addition to that, Nvidia's installer deletes various files that are part of XFree86 which ship with Red Hat Linux including the libglx.a X server extension and our Mesa libGL. Those files become replaced by Nvidia's proprietary libglx.a and libGL, of which only work with their proprietary driver. That means that once you install Nvidia's video driver, if you ever use any other video driver at all, be it the distribution supplied "nv" driver, or any other video driver, you have no more OpenGL support at all, as their GLX and libGL only work with their hardware period. Since we do not support Nvidia's proprietary drivers, any OpenGL related problems that occur on any system that has had Nvidia's proprietary driver installed on it are not supported by Red Hat until the user manually reinstalls the Red Hat supplied XFree86-Mesa-libGL and XFree86 packages in order to restore the system supplied files Nvidia's installer gratuitously deletes.

I mention the above in this bug report just for some background into some of the problems caused by Nvidia's current driver installation method, in hopes it is useful to understand the problem reported in this bug report, and perhaps other problems Nvidia users encounter.

The specific problem reported in this bug report will happen for `_all_` Nvidia proprietary driver users on all Nvidia hardware, and it is caused by Nvidia's driver installation mechanism not using RPM as it should be. If they used RPM, then this type of problem would be avoidable by using RPM's conflicts mechanism and stating the following in their spec file:

```
Conflicts: XFree86-Mesa-libGL
```

Their installation program *should* be uninstalling the Red Hat supplied XFree86-Mesa-libGL package in order to remove our libGL, or it should be installing their libGL elsewhere in the system and using LD_PRELOAD script dropped in /etc/profile.d or some other solution than randomly deleting distribution supplied files that are managed by rpm. They also should NOT be deleting any of our XFree86 supplied X server modules as that causes Nvidia users problems if they switch hardware or decide to use the 'nv' driver.

However, currently they do just delete these files, and that mechanism is not foolproof because it wrongly assumes:

- 1) That there is only *_one_* system supplied libGL shared library
- 2) That the libGL supplied by the system is installed in a specific location that will never change

That assumption may have been true in previous operating system releases, but only by random chance. There is nothing anywhere that specifies that multiple libGL's can't be supplied by the operating system, nor where they must be installed. The OpenGL ABI for Linux on x86 states only that /usr/lib/libGL* must either be the system libGL, or that it must be a symbolic link to the system libGL.

The libGL which ships Fedora Core 1 contains several optimizations which were done by Jakub Jelinek to improve application startup time by making libGL prelinkable and dramatically reducing the number of dynamic relocation processing that needs to be done at application startup time. This also improves runtime performance by having less symbols have to go through the GOT. Another performance improvement done by Jakub is thread local storage (TLS) support has been added to libGL. For maximum performance with TLS, i686 instructions are used which are not present on i586 or earlier processors, and are not present on some i686 class hardware which does not implement the optional instructions in the i686 architecture.

As such, in order to both get this performance enhancement, and also still provide libGL compatibility with Cyrix i686 CPUs, AMD K5/K6 etc. CPUs and other similar chips, we have supplied 2 libGL libraries, one which should work on all systems out there, which is optimized for some performance gains, and a second libGL which is specifically optimized for i686 class CPUs. The compat libGL resides in the normal location, and the i686 libGL resides in the "tls" subdirectory under that.

Since Nvidia's installer only looks for the one single libGL when it goes on it's deleting rampage, it misses the second libGL that we supply. When you start any OpenGL application on a true i686 compatible chip which implements the optional i686 instructions,

the system detects if you have an i686 compatible processor or not, and whether or not TLS support is available on your system, and it will use the TLS libGL instead of the /usr/lib/libGL.so.* library if your system supports TLS. Since this library is not the Nvidia libGL library, it will attempt to connect to the DRI extension, however Nvidia does not support the DRI extension or use DRI in any way, so the application will fail with the error:

Xlib: extension "XFree86-DRI" missing

We can't fix Nvidia's installer, however they will likely fix it in a future driver update. Hopefully they will fix it properly and make it uninstall XFree86-Mesa-libGL first, to avoid this and any future problems in future libGL enhancements. Until Nvidia fixes their installer however, or changes their installation and configuration process to not conflict with or delete the XFree86 supplied files we ship, users need to manually work around this problem by doing either:

```
rm -f /usr/X11R6/lib/tls/libGL*
```

or

```
rpm -e --nodeps XFree86-Mesa-libGL
```

Either of the above solutions will work around this Nvidia driver installation bug for the time being until Nvidia is able to provide users with a fixed driver package.

Note that every time you upgrade XFree86 on your system, the XFree86-Mesa-libGL package will be installed in order to meet libGL dependency requirements. That wouldn't occur if Nvidia's drivers were in rpm format and provided libGL that way, as their package would satisfy the libGL requirement then.

Hopefully in the future Nvidia will find a better method of driver installation that is more harmonious with the operating system installation, and users won't experience these problems during upgrades.

Hopefully this will help users who are experience this problem to both work around the issue, and also to understand it and other similar related problems in the future.

```
--  
--  
Jim  
jimod AT jimod DOT plus DOT com
```

This mail was composed and sent using Linux not M\$ Windoze
