Subject: patch driver-core-implement-shadow-directory-support-for-device-classes.patch added to gregkh-2.6 tr
Posted by gregkh on Sat, 21 Jul 2007 06:36:02 GMT
View Forum Message <> Reply to Message

This is a note to let you know that I've just added the patch titled

Subject: [PATCH 4/4] driver core: Implement shadow directory support for device classes.

to my gregkh-2.6 tree. Its filename is

driver-core-implement-shadow-directory-support-for-device-classes.patch

This tree can be found at
http://www.kernel.org/pub/linux/kernel/people/gregkh/gregkh-2.6/patches/

>From ebiederm@xmission.com Fri Jul 20 23:21:18 2007
From: ebiederm@xmission.com (Eric W. Biederman)
Date: Wed, 18 Jul 2007 22:47:27 -0600
Subject: [PATCH 4/4] driver core: Implement shadow directory support for device classes.
To: Greg KH <greg@kroah.com>
Cc: Greg KH <gregkh@suse.de>, Dave Hansen <hansendc@us.ibm.com>, Benjamin Thery <benjamin.thery@bull.net>, Linux Containers <containers@lists.osdl.org>, Tejun Heo <htejun@gmail.com>
Message-ID: <m1r6n52e5c.fsf_-_@ebiederm.dsl.xmission.com>

This patch enables shadowing on every class directory if struct class
has shadow_ops.

In addition device_del and device_rename were modified to use
sysfs_delete_link and sysfs_rename_link respectively to ensure
when these operations happen on devices whos classes have
shadow operations that they work properly.

Signed-off-by: Eric W. Biederman <ebiederm@xmission.com>
Cc: Tejun Heo <htejun@gmail.com>
Signed-off-by: Greg Kroah-Hartman <gregkh@suse.de>

```
 drivers/base/class.c   |   30 ++++++++++++++++++++++++++----
 drivers/base/core.c    |   45 +++++++++++++++++++++++++++----------------------
 include/linux/device.h |    2 ++
 3 files changed, 52 insertions(+), 25 deletions(-)
```

```
--- a/drivers/base/class.c
+++ b/drivers/base/class.c
@@ -134,6 +134,17 @@ static void remove_class_attrs(struct cl
  }
 }

+static int class_setup_shadowing(struct class *cls)
+{
+ const struct shadow_dir_operations *shadow_ops;
+
+ shadow_ops = cls->shadow_ops;
+ if (!shadow_ops)
+  return 0;
+
+ return sysfs_enable_shadowing(&cls->subsys.kobj, shadow_ops);
+}
+
 int class_register(struct class * cls)
 {
  int error;
@@ -152,11 +163,22 @@ int class_register(struct class * cls)
  subsys_set_kset(cls, class_subsys);

  error = subsystem_register(&cls->subsys);
- if (!error) {
-  error = add_class_attrs(class_get(cls));
-  class_put(cls);
- }
+ if (error)
+  goto out;
+
+ error = class_setup_shadowing(cls);
+ if (error)
+  goto out_unregister;
+
+ error = add_class_attrs(cls);
+ if (error)
+  goto out_unregister;
+
+out:
  return error;
+out_unregister:
+ subsystem_unregister(&cls->subsys);
+ goto out;
 }

 void class_unregister(struct class * cls)
--- a/drivers/base/core.c
```

```
+++ b/drivers/base/core.c
@@ -645,8 +645,14 @@ static struct kobject * get_device_paren
    return kobj;

   /* or create a new class-directory at the parent device */
-  return kobject_kset_add_dir(&dev->class->class_dirs,
+  kobj = kobject_kset_add_dir(&dev->class->class_dirs,
         parent_kobj, dev->class->name);
+
+  /* If we created a new class-directory setup shadowing */
+  if (kobj && dev->class->shadow_ops)
+   sysfs_enable_shadowing(kobj, dev->class->shadow_ops);
+
+  return kobj;
  }

  if (parent)
@@ -844,8 +850,8 @@ int device_add(struct device *dev)
   /* If this is not a "fake" compatible device, remove the
    * symlink from the class to the device. */
   if (dev->kobj.parent != &dev->class->subsys.kobj)
-   sysfs_remove_link(&dev->class->subsys.kobj,
-       dev->bus_id);
+   sysfs_delete_link(&dev->class->subsys.kobj,
+       &dev->kobj, dev->bus_id);
   if (parent && parent->bus) {
 #ifdef CONFIG_SYSFS_DEPRECATED
    char *class_name = make_class_name(dev->class->name,
@@ -1243,6 +1249,13 @@ int device_rename(struct device *dev, ch
  strlcpy(old_device_name, dev->bus_id, BUS_ID_SIZE);
  strlcpy(dev->bus_id, new_name, BUS_ID_SIZE);

+ if (dev->class && (dev->kobj.parent != &dev->class->subsys.kobj)) {
+  error = sysfs_rename_link(&dev->class->subsys.kobj,
+   &dev->kobj, old_device_name, new_name);
+  if (error)
+   goto out;
+ }
+
  error = kobject_rename(&dev->kobj, new_name);
  if (error) {
   strlcpy(dev->bus_id, old_device_name, BUS_ID_SIZE);
@@ -1251,27 +1264,17 @@ int device_rename(struct device *dev, ch

 #ifdef CONFIG_SYSFS_DEPRECATED
  if (old_class_name) {
+  error = -ENOMEM;
   new_class_name = make_class_name(dev->class->name, &dev->kobj);
```

```
-  if (new_class_name) {
-   error = sysfs_create_link(&dev->parent->kobj,
-         &dev->kobj, new_class_name);
-   if (error)
-    goto out;
-   sysfs_remove_link(&dev->parent->kobj, old_class_name);
-  }
- }
-#endif
+  if (!new_class_name)
+   goto out;

- if (dev->class) {
-  sysfs_remove_link(&dev->class->subsys.kobj, old_device_name);
-  error = sysfs_create_link(&dev->class->subsys.kobj, &dev->kobj,
-        dev->bus_id);
-  if (error) {
-   /* Uh... how to unravel this if restoring can fail? */
-   dev_err(dev, "%s: sysfs_create_symlink failed (%d)\n",
-    __FUNCTION__, error);
-  }
+  error = sysfs_rename_link(&dev->parent->kobj, &dev->kobj,
+        old_class_name, new_class_name);
+  if (error)
+   goto out;
  }
+#endif
 out:
  put_device(dev);

--- a/include/linux/device.h
+++ b/include/linux/device.h
@@ -200,6 +200,8 @@ struct class {

  int (*suspend)(struct device *, pm_message_t state);
  int (*resume)(struct device *);
+
+ const struct shadow_dir_operations *shadow_ops;
 };

 extern int __must_check class_register(struct class *);
```

Patches currently in gregkh-2.6 which might be from greg@kroah.com are

bad/pci-domain/pci-device-ensure-sysdata-initialised.patch
bad/pci-domain/pci-fix-the-x86-pci-domain-support-fix.patch
bad/relayfs/sysfs-update-relay-file-support-for-generic-relay-api.patch

bad/relayfs/relay-consolidate-relayfs-core-into-kernel-relay.c.patch
bad/relayfs/relay-relay-header-cleanup.patch
bad/relayfs/relayfs-remove-relayfs-in-favour-of-config_relay.patch
bad/relayfs/sysfs-add-__attr_relay-helper-for-relay-attributes.patch
bad/relayfs/sysfs-relay-channel-buffers-as-sysfs-attributes.patch
bad/usbip/usb-usbip-more-dead-code-fix.patch
bad/usbip/usb-usbip-build-fix.patch
bad/usbip/usb-usbip-warning-fixes.patch
bad/ndevfs.patch
bad/battery-class-driver.patch
bad/driver-model-convert-driver-model-to-mutexes.patch
bad/gpl_future-test.patch
bad/gregkh-debugfs_example.patch
bad/speakup-kconfig-fix.patch
bad/speakup-build-fix.patch
bad/pci-use-new-multi-phase-suspend-infrastructure.patch
bad/shot-accross-the-bow.patch
bad/no-more-non-gpl-modules.patch
bad/spi-device.patch
bad/ata_piix-multithread.patch
bad/uio-irq.patch
bad/pci-two-drivers-on-one-pci-device.patch
bad/pci-dynamic-id-cleanup.patch
bad/input-device.patch
bad/usb-stimulus.patch
driver/nozomi.patch
driver/kobject-put-kobject_actions-in-kobject.h.patch
driver/sysfs-implement-sysfs-manged-shadow-directory-support.patch
driver/sysfs-implement-sysfs_delete_link-and-sysfs_rename_link.patch
driver/sysfs-remove-first-pass-at-shadow-directory-support.patch
driver/driver-core-implement-shadow-directory-support-for-device-classes.patch
gregkh/gkh-version.patch
gregkh/sysfs-test.patch
gregkh/sysrq-u-laptop.patch
pci/pci_bridge-device.patch
pci/pci-piggy-bus.patch
pci/pci-move-prototypes-for-pci_bus_find_capability-to-include-linux-pci.h.patch
pci/pci-document-pci_iomap.patch
usb/usb-gotemp.patch
usb/kobject-put-kobject_actions-in-kobject.h.patch
usb/usb-add-the-concept-of-default-authorization-to-usb-hosts.patch
usb/usb-cleanup-usb_register_bus-and-hook-up-sysfs-group.patch
usb/usb-initialize-authorization-and-wusb-bits-in-usb-devices.patch
usb/usb-introduce-usb_device-authorization-bits.patch
usb/usb-usb_set_configuration-obeys-authorization.patch
usb/usb-usb.h-kernel-doc-additions.patch
HOWTO

Containers mailing list
Containers@lists.linux-foundation.org
https://lists.linux-foundation.org/mailman/listinfo/containers