Subject: [PATCH 14/15] netns: Enable tagging for net_class directories in sysfs
Posted by ebiederm on Fri, 04 Jul 2008 01:22:35 GMT
View Forum Message <> Reply to Message

The problem. Network devices show up in sysfs and with the network
namespace active multiple devices with the same name can show up in
the same directory, ouch!

To avoid that problem and allow existing applications in network namespaces
to see the same interface that is currently presented in sysfs, this
patch enables the tagging directory support in sysfs.

By using the network namespace pointers as tags to separate out the
the sysfs directory entries we ensure that we don't have conflicts
in the directories and applications only see a limited set of
the network devices.

Signed-off-by: Eric W. Biederman <ebiederm@xmission.com>
---
 include/linux/sysfs.h | 1 +
 net/Kconfig | 2 +-
 net/core/net-sysfs.c | 33 +++++++++++++++++++++++++++++++++
 3 files changed, 35 insertions(+), 1 deletions(-)

diff --git a/include/linux/sysfs.h b/include/linux/sysfs.h
index c3a30ce..1ed31bb 100644
--- a/include/linux/sysfs.h
+++ b/include/linux/sysfs.h
@@ -80,6 +80,7 @@ struct sysfs_ops {

 enum sysfs_tag_type {
  SYSFS_TAG_TYPE_NONE = 0,
+ SYSFS_TAG_TYPE_NETNS,
  SYSFS_TAG_TYPES
 };

diff --git a/net/Kconfig b/net/Kconfig
index acbf7c6..9aad03b 100644
--- a/net/Kconfig
+++ b/net/Kconfig
@@ -30,7 +30,7 @@ menu "Networking options"
 config NET_NS
  bool "Network namespace support"
  default n
- depends on EXPERIMENTAL && !SYSFS && NAMESPACES
+ depends on EXPERIMENTAL && NAMESPACES
  help
    Allow user space to create what appear to be multiple instances

```
    of the network stack.
diff --git a/net/core/net-sysfs.c b/net/core/net-sysfs.c
index 4e7b847..6227a28 100644
--- a/net/core/net-sysfs.c
+++ b/net/core/net-sysfs.c
@@ -13,7 +13,9 @@
 #include <linux/kernel.h>
 #include <linux/netdevice.h>
 #include <linux/if_arp.h>
+#include <linux/nsproxy.h>
 #include <net/sock.h>
+#include <net/net_namespace.h>
 #include <linux/rtnetlink.h>
 #include <linux/wireless.h>
 #include <net/iw_handler.h>
@@ -385,6 +387,24 @@ static struct attribute_group wireless_group = {
 };
 #endif

+static const void *net_sysfs_mount_tag(void)
+{
+ return current->nsproxy->net_ns;
+}
+
+static struct sysfs_tag_type_operations net_tag_type_operations = {
+ .mount_tag = net_sysfs_mount_tag,
+};
+
+static void net_sysfs_net_exit(struct net *net)
+{
+ sysfs_exit_tag(SYSFS_TAG_TYPE_NETNS, net);
+}
+
+static struct pernet_operations sysfs_net_ops = {
+ .exit = net_sysfs_net_exit,
+};
+
 #endif /* CONFIG_SYSFS */

 #ifdef CONFIG_HOTPLUG
@@ -421,6 +441,13 @@ static void netdev_release(struct device *d)
  kfree((char *)dev - dev->padded);
 }

+static const void *net_sysfs_tag(struct device *d)
+{
+ struct net_device *dev;
+ dev = container_of(d, struct net_device, dev);
```

```
+ return dev_net(dev);
+}
+
 static struct class net_class = {
  .name = "net",
  .dev_release = netdev_release,
@@ -430,6 +457,8 @@ static struct class net_class = {
 #ifdef CONFIG_HOTPLUG
  .dev_uevent = netdev_uevent,
 #endif
+ .tag_type = SYSFS_TAG_TYPE_NETNS,
+ .sysfs_tag = net_sysfs_tag,
 };

 /* Delete sysfs entries but hold kobject reference until after all
@@ -472,5 +501,9 @@ int netdev_register_kobject(struct net_device *net)

 int netdev_kobject_init(void)
 {
+#ifdef CONFIG_SYSFS
+ sysfs_register_tag_type(SYSFS_TAG_TYPE_NETNS, &net_tag_type_operations);
+ register_pernet_subsys(&sysfs_net_ops);
+#endif
  return class_register(&net_class);
 }
--
1.5.3.rc6.17.g1911
```