Subject: [PATCH 10/14][VLAN]: Create proc files in proper net.
Posted by Pavel Emelianov on Thu, 10 Apr 2008 14:55:44 GMT
View Forum Message <> Reply to Message

This is essentially a PATCH #9 part 2 - use the net
pointer passed in.

Signed-off-by: Pavel Emelyanov <xemul@openvz.org>

---
 net/8021q/vlan.h | 6 ++++++
 net/8021q/vlanproc.c | 44 ++++++++++++++++++++++++++++----------------
 2 files changed, 23 insertions(+), 27 deletions(-)

diff --git a/net/8021q/vlan.h b/net/8021q/vlan.h
index f27d8d6..7258357 100644
--- a/net/8021q/vlan.h
+++ b/net/8021q/vlan.h
@@ -52,7 +52,13 @@ static inline int is_vlan_dev(struct net_device *dev)

 extern int vlan_net_id;

+struct proc_dir_entry;
+
 struct vlan_net {
+ /* /proc/net/vlan */
+ struct proc_dir_entry *proc_vlan_dir;
+ /* /proc/net/vlan/config */
+ struct proc_dir_entry *proc_vlan_conf;
 };

 #endif /* !(__BEN_VLAN_802_1Q_INC__) */
diff --git a/net/8021q/vlanproc.c b/net/8021q/vlanproc.c
index 4d13aeb..995544b 100644
--- a/net/8021q/vlanproc.c
+++ b/net/8021q/vlanproc.c
@@ -34,6 +34,7 @@
 #include <linux/netdevice.h>
 #include <linux/if_vlan.h>
 #include <net/net_namespace.h>
+#include <net/netns/generic.h>
 #include "vlanproc.h"
 #include "vlan.h"

@@ -111,18 +112,6 @@ static const struct file_operations vlandev_fops = {
  * Proc filesystem derectory entries.
  */

```
-/*
- * /proc/net/vlan
- */
-
-static struct proc_dir_entry *proc_vlan_dir;
-
-/*
- * /proc/net/vlan/config
- */
-
-static struct proc_dir_entry *proc_vlan_conf;
-
 /* Strings */
 static const char *vlan_name_type_str[VLAN_NAME_TYPE_HIGHEST] = {
     [VLAN_NAME_TYPE_RAW_PLUS_VID]        = "VLAN_NAME_TYPE_RAW_PLUS_VID",
@@ -140,14 +129,13 @@ static const char
*vlan_name_type_str[VLAN_NAME_TYPE_HIGHEST] = {

 void vlan_proc_cleanup(struct net *net)
 {
- if (net != &init_net)
-  return;
+ struct vlan_net *vn = net_generic(net, vlan_net_id);

- if (proc_vlan_conf)
-  remove_proc_entry(name_conf, proc_vlan_dir);
+ if (vn->proc_vlan_conf)
+  remove_proc_entry(name_conf, vn->proc_vlan_dir);

- if (proc_vlan_dir)
-  proc_net_remove(&init_net, name_root);
+ if (vn->proc_vlan_dir)
+  proc_net_remove(net, name_root);

  /* Dynamically added entries should be cleaned up as their vlan_device
   * is removed, so we should not have to take care of it here...
@@ -160,16 +148,15 @@ void vlan_proc_cleanup(struct net *net)

 int vlan_proc_init(struct net *net)
 {
- if (net != &init_net)
-  return 0;
+ struct vlan_net *vn = net_generic(net, vlan_net_id);

- proc_vlan_dir = proc_mkdir(name_root, init_net.proc_net);
- if (!proc_vlan_dir)
+ vn->proc_vlan_dir = proc_net_mkdir(net, name_root, net->proc_net);
+ if (!vn->proc_vlan_dir)
```

```
   goto err;

- proc_vlan_conf = proc_create(name_conf, S_IFREG|S_IRUSR|S_IWUSR,
-         proc_vlan_dir, &vlan_fops);
- if (!proc_vlan_conf)
+ vn->proc_vlan_conf = proc_create(name_conf, S_IFREG|S_IRUSR|S_IWUSR,
+         vn->proc_vlan_dir, &vlan_fops);
+ if (!vn->proc_vlan_conf)
   goto err;
  return 0;

@@ -186,9 +173,10 @@ err:
 int vlan_proc_add_dev(struct net_device *vlandev)
 {
  struct vlan_dev_info *dev_info = vlan_dev_info(vlandev);
+ struct vlan_net *vn = net_generic(dev_net(vlandev), vlan_net_id);

  dev_info->dent = proc_create(vlandev->name, S_IFREG|S_IRUSR|S_IWUSR,
-         proc_vlan_dir, &vlandev_fops);
+         vn->proc_vlan_dir, &vlandev_fops);
  if (!dev_info->dent)
   return -ENOBUFS;

@@ -201,10 +189,12 @@ int vlan_proc_add_dev(struct net_device *vlandev)
  */
 int vlan_proc_rem_dev(struct net_device *vlandev)
 {
+ struct vlan_net *vn = net_generic(dev_net(vlandev), vlan_net_id);
+
  /** NOTE:  This will consume the memory pointed to by dent, it seems. */
  if (vlan_dev_info(vlandev)->dent) {
   remove_proc_entry(vlan_dev_info(vlandev)->dent->name,
-      proc_vlan_dir);
+      vn->proc_vlan_dir);
   vlan_dev_info(vlandev)->dent = NULL;
  }
  return 0;
--
1.5.3.4
```

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