Subject: [PATCH 2.6.25 6/6] net: consolidate net namespace related proc files creation
Posted by den on Thu, 15 Nov 2007 16:01:50 GMT
View Forum Message <> Reply to Message

net: consolidate net namespace related proc files creation

Signed-off-by: Denis V. Lunev <den@openvz.org>
Signed-off-by: Pavel Emelyanov <xemul@openvz.org>
---
 fs/proc/proc_net.c | 38 ++++++++++++++++++++++++++++++++++++++
 include/linux/seq_file.h | 13 +++++++++++++
 net/core/dev.c | 28 +++++-----------------------
 net/core/dev_mcast.c | 26 +++++---------------------
 net/netlink/af_netlink.c | 33 +++++++--------------------------
 net/packet/af_packet.c | 26 +++++---------------------
 net/unix/af_unix.c | 31 +++++++------------------------
 net/wireless/wext.c | 24 +++++-------------------
 8 files changed, 80 insertions(+), 139 deletions(-)

diff --git a/fs/proc/proc_net.c b/fs/proc/proc_net.c
index 131f9c6..421ea28 100644
--- a/fs/proc/proc_net.c
+++ b/fs/proc/proc_net.c
@@ -22,10 +22,48 @@
 #include <linux/mount.h>
 #include <linux/nsproxy.h>
 #include <net/net_namespace.h>
+#include <linux/seq_file.h>

 #include "internal.h"

+int seq_open_net(struct inode *ino, struct file *f,
+   const struct seq_operations *ops, int size)
+{
+ struct net *net;
+ struct seq_net_private *p;
+
+ BUG_ON(size < sizeof(*p));
+
+ net = get_proc_net(ino);
+ if (net == NULL)
+ return -ENXIO;
+
+ p = __seq_open_private(f, ops, size);
+ if (p == NULL) {
+ put_net(net);

```
+  return -ENOMEM;
+ }
+ p->net = net;
+ return 0;
+}
+EXPORT_SYMBOL_GPL(seq_open_net);
+
+int seq_release_net(struct inode *ino, struct file *f)
+{
+ struct seq_file *seq;
+ struct seq_net_private *p;
+
+ seq = f->private_data;
+ p = seq->private;
+
+ put_net(p->net);
+ seq_release_private(ino, f);
+ return 0;
+}
+EXPORT_SYMBOL_GPL(seq_release_net);
+
+
 struct proc_dir_entry *proc_net_fops_create(struct net *net,
  const char *name, mode_t mode, const struct file_operations *fops)
 {
diff --git a/include/linux/seq_file.h b/include/linux/seq_file.h
index ebbc02b..648dfeb 100644
--- a/include/linux/seq_file.h
+++ b/include/linux/seq_file.h
@@ -63,5 +63,18 @@ extern struct list_head *seq_list_start_head(struct list_head *head,
 extern struct list_head *seq_list_next(void *v, struct list_head *head,
   loff_t *ppos);

+struct net;
+struct seq_net_private {
+ struct net *net;
+};
+
+int seq_open_net(struct inode *, struct file *,
+   const struct seq_operations *, int);
+int seq_release_net(struct inode *, struct file *);
+static inline struct net *seq_file_net(struct seq_file *seq)
+{
+ return ((struct seq_net_private *)seq->private)->net;
+}
+
 #endif
 #endif
```

```
diff --git a/net/core/dev.c b/net/core/dev.c
index 86d6261..043e2f8 100644
--- a/net/core/dev.c
+++ b/net/core/dev.c
@@ -2360,7 +2360,7 @@ static int dev_ifconf(struct net *net, char __user *arg)
  */
 void *dev_seq_start(struct seq_file *seq, loff_t *pos)
 {
- struct net *net = seq->private;
+ struct net *net = seq_file_net(seq);
  loff_t off;
  struct net_device *dev;

@@ -2378,7 +2378,7 @@ void *dev_seq_start(struct seq_file *seq, loff_t *pos)

 void *dev_seq_next(struct seq_file *seq, void *v, loff_t *pos)
 {
- struct net *net = seq->private;
+ struct net *net = seq_file_net(seq);
  ++*pos;
  return v == SEQ_START_TOKEN ?
   first_net_device(net) : next_net_device((struct net_device *)v);
@@ -2477,26 +2477,8 @@ static const struct seq_operations dev_seq_ops = {

 static int dev_seq_open(struct inode *inode, struct file *file)
 {
- struct seq_file *seq;
- int res;
- res =  seq_open(file, &dev_seq_ops);
- if (!res) {
-  seq = file->private_data;
-  seq->private = get_proc_net(inode);
-  if (!seq->private) {
-   seq_release(inode, file);
-   res = -ENXIO;
-  }
- }
- return res;
-}
-
-static int dev_seq_release(struct inode *inode, struct file *file)
-{
- struct seq_file *seq = file->private_data;
- struct net *net = seq->private;
- put_net(net);
- return seq_release(inode, file);
+ return seq_open_net(inode, file, &dev_seq_ops,
+       sizeof(struct seq_net_private));
```

```
 }

 static const struct file_operations dev_seq_fops = {
@@ -2504,7 +2486,7 @@ static const struct file_operations dev_seq_fops = {
  .open    = dev_seq_open,
  .read    = seq_read,
  .llseek  = seq_lseek,
- .release = dev_seq_release,
+ .release = seq_release_net,
 };

 static const struct seq_operations softnet_seq_ops = {
diff --git a/net/core/dev_mcast.c b/net/core/dev_mcast.c
index 69fff16..63f0b33 100644
--- a/net/core/dev_mcast.c
+++ b/net/core/dev_mcast.c
@@ -187,7 +187,7 @@ EXPORT_SYMBOL(dev_mc_unsync);
 #ifdef CONFIG_PROC_FS
 static void *dev_mc_seq_start(struct seq_file *seq, loff_t *pos)
 {
- struct net *net = seq->private;
+ struct net *net = seq_file_net(seq);
  struct net_device *dev;
  loff_t off = 0;

@@ -241,26 +241,8 @@ static const struct seq_operations dev_mc_seq_ops = {

 static int dev_mc_seq_open(struct inode *inode, struct file *file)
 {
- struct seq_file *seq;
- int res;
- res = seq_open(file, &dev_mc_seq_ops);
- if (!res) {
-  seq = file->private_data;
-  seq->private = get_proc_net(inode);
-  if (!seq->private) {
-   seq_release(inode, file);
-   res = -ENXIO;
-  }
- }
- return res;
-}
-
-static int dev_mc_seq_release(struct inode *inode, struct file *file)
-{
- struct seq_file *seq = file->private_data;
- struct net *net = seq->private;
- put_net(net);
```

```
- return seq_release(inode, file);
+ return seq_open_net(inode, file, &dev_mc_seq_ops,
+       sizeof(struct seq_net_private));
 }

 static const struct file_operations dev_mc_seq_fops = {
@@ -268,7 +250,7 @@ static const struct file_operations dev_mc_seq_fops = {
  .open    = dev_mc_seq_open,
  .read    = seq_read,
  .llseek  = seq_lseek,
- .release = dev_mc_seq_release,
+ .release = seq_release_net,
 };

 #endif
diff --git a/net/netlink/af_netlink.c b/net/netlink/af_netlink.c
index de3988b..1518244 100644
--- a/net/netlink/af_netlink.c
+++ b/net/netlink/af_netlink.c
@@ -1681,7 +1681,7 @@ int nlmsg_notify(struct sock *sk, struct sk_buff *skb, u32 pid,

 #ifdef CONFIG_PROC_FS
 struct nl_seq_iter {
- struct net *net;
+ struct seq_net_private p;
  int link;
  int hash_idx;
 };
@@ -1699,7 +1699,7 @@ static struct sock *netlink_seq_socket_idx(struct seq_file *seq, loff_t
pos)

   for (j = 0; j <= hash->mask; j++) {
    sk_for_each(s, node, &hash->table[j]) {
-    if (iter->net != s->sk_net)
+    if (iter->p.net != s->sk_net)
      continue;
     if (off == pos) {
      iter->link = i;
@@ -1734,7 +1734,7 @@ static void *netlink_seq_next(struct seq_file *seq, void *v, loff_t *pos)
  s = v;
  do {
   s = sk_next(s);
- } while (s && (iter->net != s->sk_net));
+ } while (s && (iter->p.net != s->sk_net));
  if (s)
   return s;

@@ -1746,7 +1746,7 @@ static void *netlink_seq_next(struct seq_file *seq, void *v, loff_t *pos)
```

```
   for (; j <= hash->mask; j++) {
    s = sk_head(&hash->table[j]);
-   while (s && (iter->net != s->sk_net))
+   while (s && (iter->p.net != s->sk_net))
     s = sk_next(s);
    if (s) {
     iter->link = i;
@@ -1802,27 +1802,8 @@ static const struct seq_operations netlink_seq_ops = {

 static int netlink_seq_open(struct inode *inode, struct file *file)
 {
- struct nl_seq_iter *iter;
-
- iter = __seq_open_private(file, &netlink_seq_ops, sizeof(*iter));
- if (!iter)
-  return -ENOMEM;
-
- iter->net = get_proc_net(inode);
- if (!iter->net) {
-  seq_release_private(inode, file);
-  return -ENXIO;
- }
-
- return 0;
-}
-
-static int netlink_seq_release(struct inode *inode, struct file *file)
-{
- struct seq_file *seq = file->private_data;
- struct nl_seq_iter *iter = seq->private;
- put_net(iter->net);
- return seq_release_private(inode, file);
+ return seq_open_net(inode, file, &netlink_seq_ops,
+     sizeof(struct nl_seq_iter));
 }

 static const struct file_operations netlink_seq_fops = {
@@ -1830,7 +1811,7 @@ static const struct file_operations netlink_seq_fops = {
   .open  = netlink_seq_open,
   .read  = seq_read,
   .llseek  = seq_lseek,
- .release = netlink_seq_release,
+ .release = seq_release_net,
 };

 #endif
diff --git a/net/packet/af_packet.c b/net/packet/af_packet.c
```

```
index 45e3cbc..ace29f1 100644
--- a/net/packet/af_packet.c
+++ b/net/packet/af_packet.c
@@ -1871,7 +1871,7 @@ static inline struct sock *packet_seq_idx(struct net *net, loff_t off)

 static void *packet_seq_start(struct seq_file *seq, loff_t *pos)
 {
- struct net *net = seq->private;
+ struct net *net = seq_file_net(seq);
  read_lock(&net->packet_sklist_lock);
  return *pos ? packet_seq_idx(net, *pos - 1) : SEQ_START_TOKEN;
 }
@@ -1924,26 +1924,8 @@ static const struct seq_operations packet_seq_ops = {

 static int packet_seq_open(struct inode *inode, struct file *file)
 {
- struct seq_file *seq;
- int res;
- res = seq_open(file, &packet_seq_ops);
- if (!res) {
-  seq = file->private_data;
-  seq->private = get_proc_net(inode);
-  if (!seq->private) {
-   seq_release(inode, file);
-   res = -ENXIO;
-  }
- }
- return res;
-}
-
-static int packet_seq_release(struct inode *inode, struct file *file)
-{
- struct seq_file *seq=  file->private_data;
- struct net *net = seq->private;
- put_net(net);
- return seq_release(inode, file);
+ return seq_open_net(inode, file, &packet_seq_ops,
+       sizeof(struct seq_net_private));
 }

 static const struct file_operations packet_seq_fops = {
@@ -1951,7 +1933,7 @@ static const struct file_operations packet_seq_fops = {
  .open  = packet_seq_open,
  .read  = seq_read,
  .llseek  = seq_lseek,
- .release = packet_seq_release,
+ .release = seq_release_net,
 };
```

```
 #endif
diff --git a/net/unix/af_unix.c b/net/unix/af_unix.c
index 93d7e55..6be6d87 100644
--- a/net/unix/af_unix.c
+++ b/net/unix/af_unix.c
@@ -2011,7 +2011,7 @@ static unsigned int unix_poll(struct file * file, struct socket *sock,
poll_tabl

 #ifdef CONFIG_PROC_FS
 struct unix_iter_state {
- struct net *net;
+ struct seq_net_private p;
  int i;
 };
 static struct sock *unix_seq_idx(struct unix_iter_state *iter, loff_t pos)
@@ -2020,7 +2020,7 @@ static struct sock *unix_seq_idx(struct unix_iter_state *iter, loff_t pos)
  struct sock *s;

  for (s = first_unix_socket(&iter->i); s; s = next_unix_socket(&iter->i, s)) {
-  if (s->sk_net != iter->net)
+  if (s->sk_net != iter->p.net)
     continue;
   if (off == pos)
     return s;
@@ -2047,7 +2047,7 @@ static void *unix_seq_next(struct seq_file *seq, void *v, loff_t *pos)
   sk = first_unix_socket(&iter->i);
  else
   sk = next_unix_socket(&iter->i, sk);
- while (sk && (sk->sk_net != iter->net))
+ while (sk && (sk->sk_net != iter->p.net))
   sk = next_unix_socket(&iter->i, sk);
  return sk;
 }
@@ -2111,27 +2111,8 @@ static const struct seq_operations unix_seq_ops = {

 static int unix_seq_open(struct inode *inode, struct file *file)
 {
- struct unix_iter_state *it;
-
- it = __seq_open_private(file, &unix_seq_ops,
-    sizeof(struct unix_iter_state));
- if (it == NULL)
-  return -ENOMEM;
-
- it->net = get_proc_net(inode);
- if (it->net == NULL) {
-  seq_release_private(inode, file);
```

```
-  return -ENXIO;
- }
- return 0;
-}
-
-static int unix_seq_release(struct inode *inode, struct file *file)
-{
- struct seq_file *seq = file->private_data;
- struct unix_iter_state *iter = seq->private;
- put_net(iter->net);
- return seq_release_private(inode, file);
+ return seq_open_net(inode, file, &unix_seq_ops,
+       sizeof(struct unix_iter_state));
 }

 static const struct file_operations unix_seq_fops = {
@@ -2139,7 +2120,7 @@ static const struct file_operations unix_seq_fops = {
  .open  = unix_seq_open,
  .read  = seq_read,
  .llseek  = seq_lseek,
- .release = unix_seq_release,
+ .release = seq_release_net,
 };

 #endif
diff --git a/net/wireless/wext.c b/net/wireless/wext.c
index 85805f2..3ffbd1b 100644
--- a/net/wireless/wext.c
+++ b/net/wireless/wext.c
@@ -673,26 +673,8 @@ static const struct seq_operations wireless_seq_ops = {

 static int wireless_seq_open(struct inode *inode, struct file *file)
 {
- struct seq_file *seq;
- int res;
- res = seq_open(file, &wireless_seq_ops);
- if (!res) {
-  seq = file->private_data;
-  seq->private = get_proc_net(inode);
-  if (!seq->private) {
-   seq_release(inode, file);
-   res = -ENXIO;
-  }
- }
- return res;
-}
-
-static int wireless_seq_release(struct inode *inode, struct file *file)
```

```
-{
- struct seq_file *seq = file->private_data;
- struct net *net = seq->private;
- put_net(net);
- return seq_release(inode, file);
+ return seq_open_net(inode, file, &wireless_seq_ops,
+       sizeof(struct seq_net_private));
 }

 static const struct file_operations wireless_seq_fops = {
@@ -700,7 +682,7 @@ static const struct file_operations wireless_seq_fops = {
  .open    = wireless_seq_open,
  .read    = seq_read,
  .llseek  = seq_lseek,
- .release = wireless_seq_release,
+ .release = seq_release_net,
 };

 int wext_proc_init(struct net *net)
--
1.5.3.rc5
```

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

---

Subject: Re: [PATCH 2.6.25 6/6] net: consolidate net namespace related proc files creation
Posted by ebiederm on Mon, 19 Nov 2007 20:15:19 GMT
View Forum Message <> Reply to Message

"Denis V. Lunev" <den@openvz.org> writes:

> net: consolidate net namespace related proc files creation
>
> Signed-off-by: Denis V. Lunev <den@openvz.org>
> Signed-off-by: Pavel Emelyanov <xemul@openvz.org>

Acked-by: "Eric W. Biederman" <ebiederm@xmission.com>

Looks good to me, and it looks like there are a lot more places
we can use this as more code is converted.

Eric

_____
Containers mailing list

Subject: Re: [PATCH 2.6.25 6/6] net: consolidate net namespace related proc files creation
Posted by davem on Tue, 20 Nov 2007 06:33:54 GMT
View Forum Message <> Reply to Message

From: "Denis V. Lunev" <den@openvz.org>
Date: Thu, 15 Nov 2007 19:01:50 +0300

> net: consolidate net namespace related proc files creation
>
> Signed-off-by: Denis V. Lunev <den@openvz.org>
> Signed-off-by: Pavel Emelyanov <xemul@openvz.org>

Also applied, thanks a lot.

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