
Subject: [patch 18/20] [Network namespace] For debug purpose only. Show ref count for current namespace and |

Posted by [Daniel Lezcano](#) on Sun, 10 Dec 2006 21:58:35 GMT

[View Forum Message](#) <> [Reply to Message](#)

Signed-off-by: Daniel Lezcano <dlezcano@fr.ibm.com>

```
fs/debugfs/net_ns.c | 72 ++++++-----  
1 files changed, 71 insertions(+), 1 deletion(-)
```

Index: 2.6.19-rc6-mm2/fs/debugfs/net_ns.c

```
--- 2.6.19-rc6-mm2.orig/fs/debugfs/net_ns.c
```

```
+++ 2.6.19-rc6-mm2/fs/debugfs/net_ns.c
```

```
@@ -21,6 +21,7 @@
```

```
 static struct dentry *net_ns_dentry;
```

```
 static struct dentry *net_ns_dentry_dev;
```

```
 static struct dentry *net_ns_dentry_start;
```

```
+static struct dentry *net_ns_dentry_info;
```

```
 static ssize_t net_ns_dev_read_file(struct file *file, char __user *user_buf,  
         size_t count, loff_t *ppos)
```

```
@@ -109,6 +110,63 @@ static ssize_t net_ns_start_write_file(s  
     return count;
```

```
}
```

```
+static int net_ns_info_open_file(struct inode *inode, struct file *file)
```

```
+
```

```
+ return 0;
```

```
+
```

```
+static ssize_t net_ns_info_read_file(struct file *file, char __user *user_buf,
```

```
+     size_t count, loff_t *ppos)
```

```
+
```

```
+ const unsigned int length = 256;
```

```
+ size_t len;
```

```
+ char buff[length];
```

```
+ char *level;
```

```
+ struct net_namespace *net_ns = current_net_ns;
```

```
+
```

```
+ if (*ppos < 0)
```

```
+     return -EINVAL;
```

```
+ if (*ppos >= count)
```

```
+     return 0;
```

```
+ if (!count)
```

```
+     return 0;
```

```

+
+ switch (net_ns->level) {
+ case NET_NS_LEVEL2:
+ level = "layer 2";
+ break;
+ case NET_NS_LEVEL3:
+ level = "layer 3";
+ break;
+ default:
+ level = "unknown";
+ break;
+ }
+
+ sprintf(buff,"refcnt: %d\nlevel: %s\n",
+ atomic_read(&net_ns->kref.refcount), level);
+
+ len = strlen(buff);
+ if (len > count)
+ len = count;
+
+ if (copy_to_user(user_buf, buff, len))
+ return -EINVAL;
+
+ *ppos += count;
+
+ return count;
+}
+
+static ssize_t net_ns_info_write_file(struct file *file,
+ const char __user *user_buf,
+ size_t count, loff_t *ppos)
+{
+
+ return -EPERM;
+}
+
+
static struct file_operations net_ns_dev_fops = {
    .read =      net_ns_dev_read_file,
    .write =     net_ns_dev_write_file,
@@ -121,11 +179,17 @@ static struct file_operations net_ns_sta
    .open =      net_ns_start_open_file,
};

+static struct file_operations net_ns_info_fops = {
+    .read =      net_ns_info_read_file,
+    .write =     net_ns_info_write_file,
+    .open =      net_ns_info_open_file,

```

```

+};

+
static int __init net_ns_init(void)
{
    net_ns_dentry = debugfs_create_dir("net_ns", NULL);

- net_ns_dentry_dev = debugfs_create_file("dev", 0666,
+ net_ns_dentry_dev = debugfs_create_file("dev", 0444,
    net_ns_dentry,
    NULL,
    &net_ns_dev_fops);
@@ -135,11 +199,17 @@ static int __init net_ns_init(void)
    NULL,
    &net_ns_start_fops);

+ net_ns_dentry_info = debugfs_create_file("info", 0444,
+     net_ns_dentry,
+     NULL,
+     &net_ns_info_fops);
+
    return 0;
}

static void __exit net_ns_exit(void)
{
+ debugfs_remove(net_ns_dentry_info);
    debugfs_remove(net_ns_dentry_start);
    debugfs_remove(net_ns_dentry_dev);
    debugfs_remove(net_ns_dentry);

--
```

Containers mailing list
 Containers@lists.osdl.org
<https://lists.osdl.org/mailman/listinfo/containers>
