
Subject: [PATCH RFC 17/31] net: Factor out __dev_alloc_name from dev_alloc_name

Posted by [ebiederm](#) on Thu, 25 Jan 2007 19:00:19 GMT

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

From: Eric W. Biederman <ebiederm@xmission.com> - unquoted

When forcibly changing the network namespace of a device I need something that can generate a name for the device in the new namespace without overwriting the old name.

__dev_alloc_name provides me that functionality.

Signed-off-by: Eric W. Biederman <ebiederm@xmission.com>

net/core/dev.c | 44 ++++++-----
1 files changed, 33 insertions(+), 11 deletions(-)

diff --git a/net/core/dev.c b/net/core/dev.c

index 32fe905..fc0d2af 100644

--- a/net/core/dev.c

+++ b/net/core/dev.c

```
@@ -655,9 +655,10 @@ int dev_valid_name(const char *name)
}
```

```
/**
```

```
- * dev_alloc_name - allocate a name for a device
```

```
- * @dev: device
```

```
+ * __dev_alloc_name - allocate a name for a device
```

```
+ * @net: network namespace to allocate the device name in
```

```
+ * @name: name format string
```

```
+ * @buf: scratch buffer and result name string
```

```
+ *
```

```
+ * Passed a format string - eg "lt%d" it will try and find a suitable
```

```
+ * id. It scans list of devices to build up a free map, then chooses
```

```
@@ -668,18 +669,13 @@ int dev_valid_name(const char *name)
```

```
+ * Returns the number of the unit assigned or a negative errno code.
```

```
+ */
```

```
-int dev_alloc_name(struct net_device *dev, const char *name)
```

```
+static int __dev_alloc_name(net_t net, const char *name, char buf[IFNAMSIZ])
```

```
{
```

```
    int i = 0;
```

```
- char buf[IFNAMSIZ];
```

```
    const char *p;
```

```
    const int max_netdevices = 8*PAGE_SIZE;
```

```
    long *inuse;
```

```
    struct net_device *d;
```

```

- net_t net;
-
- BUG_ON(null_net(dev->nd_net));
- net = dev->nd_net;

p = strchr(name, IFNAMSIZ-1, '%');
if (p) {
@@ -713,10 +709,8 @@ int dev_alloc_name(struct net_device *dev, const char *name)
}

snprintf(buf, sizeof(buf), name, i);
- if (!__dev_get_by_name(net, buf)) {
- strcpy(dev->name, buf, IFNAMSIZ);
+ if (!__dev_get_by_name(net, buf))
    return i;
- }

/* It is possible to run out of possible slots
 * when the name is long and there isn't enough space left
@@ -725,6 +719,34 @@ int dev_alloc_name(struct net_device *dev, const char *name)
    return -ENFILE;
}

+/**
+ * dev_alloc_name - allocate a name for a device
+ * @dev: device
+ * @name: name format string
+ *
+ * Passed a format string - eg "lt%d" it will try and find a suitable
+ * id. It scans list of devices to build up a free map, then chooses
+ * the first empty slot. The caller must hold the dev_base or rtnl lock
+ * while allocating the name and adding the device in order to avoid
+ * duplicates.
+ * Limited to bits_per_byte * page size devices (ie 32K on most platforms).
+ * Returns the number of the unit assigned or a negative errno code.
+ */
+
+int dev_alloc_name(struct net_device *dev, const char *name)
+{
+ char buf[IFNAMSIZ];
+ net_t net;
+ int ret;
+
+ BUG_ON(null_net(dev->nd_net));
+ net = dev->nd_net;
+ ret = __dev_alloc_name(net, name, buf);
+ if (ret >= 0)
+ strcpy(dev->name, buf, IFNAMSIZ);

```

```
+ return ret;  
+}  
+
```

```
/**  
 * dev_change_name - change name of a device  
--  
1.4.4.1.g278f
```

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