
Subject: [PATCH] Don't create tunnels with '%' in name.
Posted by [Pavel Emelianov](#) on Thu, 21 Feb 2008 12:05:27 GMT
[View Forum Message](#) <> [Reply to Message](#)

Four tunnel drivers (ip_gre, ipip, ip6_tunnel and sit) can receive a pre-defined name for a device from the userspace. Since these drivers call the register_netdevice() after this (rtnl_lock is held), the device's name may contain a '%' character.

Not sure how bad is this to have a device with a '%' in its name, but all the other places either use the register_netdev(), or explicitly call dev_alloc_name() before registering, i.e. do not allow for such names.

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

```
diff --git a/net/ipv4/ip_gre.c b/net/ipv4/ip_gre.c
index 63f6917..6b9744f 100644
--- a/net/ipv4/ip_gre.c
+++ b/net/ipv4/ip_gre.c
@@ @ -274,19 +274,24 @@ static struct ip_tunnel * ipgre_tunnel_locate(struct ip_tunnel_parm
*parms, int
if (!dev)
    return NULL;

+ if (strchr(name, '%')) {
+     if (dev_alloc_name(dev, name) < 0)
+         goto failed_free;
+ }
+
    dev->init = ipgre_tunnel_init;
    nt = netdev_priv(dev);
    nt->parms = *parms;

- if (register_netdevice(dev) < 0) {
-     free_netdev(dev);
-     goto failed;
- }
+ if (register_netdevice(dev) < 0)
+     goto failed_free;

    dev_hold(dev);
    ipgre_tunnel_link(nt);
    return nt;
```

```

+failed_free:
+ free_netdev(dev);
failed:
    return NULL;
}
diff --git a/net/ipv4/iph.c b/net/ipv4/iph.c
index da28158..118e7d9 100644
--- a/net/ipv4/iph.c
+++ b/net/ipv4/iph.c
@@ -236,19 +236,24 @@ static struct ip_tunnel *iph_tunnel_locate(struct ip_tunnel_parm
*parms, int c
if (dev == NULL)
    return NULL;

+ if (strchr(name, '%')) {
+ if (dev_alloc_name(dev, name) < 0)
+ goto failed_free;
+
+ nt = netdev_priv(dev);
dev->init = iph_tunnel_init;
nt->parms = *parms;

- if (register_netdevice(dev) < 0) {
- free_netdev(dev);
- goto failed;
- }
+ if (register_netdevice(dev) < 0)
+ goto failed_free;

dev_hold(dev);
iph_tunnel_link(nt);
return nt;

+failed_free:
+ free_netdev(dev);
failed:
    return NULL;
}
diff --git a/net/ipv6/ip6_tunnel.c b/net/ipv6/ip6_tunnel.c
index cd94064..fa83d70 100644
--- a/net/ipv6/ip6_tunnel.c
+++ b/net/ipv6/ip6_tunnel.c
@@ -245,17 +245,24 @@ static struct ip6_tnl *ip6_tnl_create(struct ip6_tnl_parm *p)
if (dev == NULL)
    goto failed;

+ if (strchr(name, '%')) {

```

```

+ if (dev_alloc_name(dev, name) < 0)
+ goto failed_free;
+
+ t = netdev_priv(dev);
dev->init = ip6_tnl_dev_init;
t->parms = *p;

- if ((err = register_netdevice(dev)) < 0) {
- free_netdev(dev);
- goto failed;
-
+ if ((err = register_netdevice(dev)) < 0)
+ goto failed_free;
+
dev_hold(dev);
ip6_tnl_link(t);
return t;
+
+failed_free:
+ free_netdev(dev);
failed:
    return NULL;
}
diff --git a/net/ipv6/sit.c b/net/ipv6/sit.c
index e77239d..a09a6b0 100644
--- a/net/ipv6/sit.c
+++ b/net/ipv6/sit.c
@@ -179,6 +179,11 @@ static struct ip_tunnel *ipip6_tunnel_locate(struct ip_tunnel_parm
*parms, int
if (dev == NULL)
    return NULL;

+ if (strchr(name, '%')) {
+ if (dev_alloc_name(dev, name) < 0)
+ goto failed_free;
+
nt = netdev_priv(dev);
dev->init = ipip6_tunnel_init;
nt->parms = *parms;
@@ -186,16 +191,16 @@ static struct ip_tunnel *ipip6_tunnel_locate(struct ip_tunnel_parm
*parms, int
if (parms->i_flags & SIT_ISATAP)
    dev->priv_flags |= IFF_ISATAP;

- if (register_netdevice(dev) < 0) {
- free_netdev(dev);

```

```
- goto failed;
- }
+ if (register_netdevice(dev) < 0)
+ goto failed_free;

dev_hold(dev);

ipip6_tunnel_link(nt);
return nt;

+failed_free:
+ free_netdev(dev);
failed:
return NULL;
}
```
