Subject: [PATCH][IPVS] Don't leak sysctl tables if the scheduler registration fails
Posted by Pavel Emelianov on Mon, 03 Dec 2007 10:04:53 GMT
View Forum Message <> Reply to Message

In case we load lblc or lblcr module we can leak some sysctl
tables if the call to register_ip_vs_scheduler() fails.

I've looked at the register_ip_vs_scheduler() code and saw, that
the only reason to fail is the name collision, so I think that
with some 3rd party schedulers this becomes a relevant issue. No?

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

```
diff --git a/net/ipv4/ipvs/ip_vs_lblc.c b/net/ipv4/ipvs/ip_vs_lblc.c
index b843a11..ad89644 100644
--- a/net/ipv4/ipvs/ip_vs_lblc.c
+++ b/net/ipv4/ipvs/ip_vs_lblc.c
@@ -580,9 +580,14 @@ static struct ip_vs_scheduler ip_vs_lblc_scheduler =

 static int __init ip_vs_lblc_init(void)
 {
+ int ret;
+
  INIT_LIST_HEAD(&ip_vs_lblc_scheduler.n_list);
  sysctl_header = register_sysctl_table(lblc_root_table);
- return register_ip_vs_scheduler(&ip_vs_lblc_scheduler);
+ ret = register_ip_vs_scheduler(&ip_vs_lblc_scheduler);
+ if (ret)
+  unregister_sysctl_table(sysctl_header);
+ return ret;
 }


diff --git a/net/ipv4/ipvs/ip_vs_lblcr.c b/net/ipv4/ipvs/ip_vs_lblcr.c
index e5b323a..2a5ed85 100644
--- a/net/ipv4/ipvs/ip_vs_lblcr.c
+++ b/net/ipv4/ipvs/ip_vs_lblcr.c
@@ -769,9 +769,14 @@ static struct ip_vs_scheduler ip_vs_lblcr_scheduler =

 static int __init ip_vs_lblcr_init(void)
 {
+ int ret;
+
  INIT_LIST_HEAD(&ip_vs_lblcr_scheduler.n_list);
  sysctl_header = register_sysctl_table(lblcr_root_table);
- return register_ip_vs_scheduler(&ip_vs_lblcr_scheduler);
```

```
+ ret = register_ip_vs_scheduler(&ip_vs_lblcr_scheduler);
+ if (ret)
+  unregister_sysctl_table(sysctl_header);
+ return ret;
 }
```