Subject: [patch 07/10][NETNS][IP6_FIB] make gc timer mindless changes
Posted by Daniel Lezcano on Thu, 15 Nov 2007 14:01:50 GMT
View Forum Message <> Reply to Message

The patch prepares to dynamically allocate the timer device.
It changes all reference to the global static variable to a
globally defined pointer to the global static variable.

That makes all the code to manage a pointer instead of a static
variable.

Signed-off-by: Daniel Lezcano <dlezcano@fr.ibm.com>
Signed-off-by: Benjamin Thery <benjamin.thery@bull.net>
---
 net/ipv6/ip6_fib.c | 21 ++++++++++++----------
 1 file changed, 11 insertions(+), 10 deletions(-)

Index: linux-2.6-netns/net/ipv6/ip6_fib.c
===
--- linux-2.6-netns.orig/net/ipv6/ip6_fib.c
+++ linux-2.6-netns/net/ipv6/ip6_fib.c
@@ -95,7 +95,8 @@ static __u32 rt_sernum;

 static void fib6_gc_timer_cb(unsigned long arg);

-static DEFINE_TIMER(ip6_fib_timer, fib6_gc_timer_cb, 0, (unsigned long)&init_net);
+static DEFINE_TIMER(__ip6_fib_timer, fib6_gc_timer_cb, 0, (unsigned long)&init_net);
+static struct timer_list *ip6_fib_timer = &__ip6_fib_timer;

 static struct fib6_walker_t fib6_walker_list = {
   .prev = &fib6_walker_list,
@@ -666,15 +667,15 @@ static int fib6_add_rt2node(struct fib6_

 static __inline__ void fib6_start_gc(struct rt6_info *rt)
 {
- if (ip6_fib_timer.expires == 0 &&
+ if (ip6_fib_timer->expires == 0 &&
      (rt->rt6i_flags & (RTF_EXPIRES|RTF_CACHE)))
- mod_timer(&ip6_fib_timer, jiffies + ip6_rt_gc_interval);
+ mod_timer(ip6_fib_timer, jiffies + ip6_rt_gc_interval);
 }

 void fib6_force_start_gc(void)
 {
- if (ip6_fib_timer.expires == 0)
- mod_timer(&ip6_fib_timer, jiffies + ip6_rt_gc_interval);
+ if (ip6_fib_timer->expires == 0)
+ mod_timer(ip6_fib_timer, jiffies + ip6_rt_gc_interval);

```
 }

 /*
@@ -1441,7 +1442,7 @@ void fib6_run_gc(unsigned long expires,
  } else {
   local_bh_disable();
   if (!spin_trylock(&fib6_gc_lock)) {
-   mod_timer(&ip6_fib_timer, jiffies + HZ);
+   mod_timer(ip6_fib_timer, jiffies + HZ);
    local_bh_enable();
    return;
   }
@@ -1454,10 +1455,10 @@ void fib6_run_gc(unsigned long expires,
  fib6_clean_all(net, fib6_age, 0, NULL);

  if (gc_args.more)
-  mod_timer(&ip6_fib_timer, jiffies + ip6_rt_gc_interval);
+  mod_timer(ip6_fib_timer, jiffies + ip6_rt_gc_interval);
  else {
-  del_timer(&ip6_fib_timer);
-  ip6_fib_timer.expires = 0;
+  del_timer(ip6_fib_timer);
+  ip6_fib_timer->expires = 0;
  }
  spin_unlock_bh(&fib6_gc_lock);
 }
@@ -1533,7 +1534,7 @@ void __init fib6_init(void)

 void fib6_gc_cleanup(void)
 {
- del_timer(&ip6_fib_timer);
+ del_timer(ip6_fib_timer);
  unregister_pernet_subsys(&fib6_net_ops);
  kmem_cache_destroy(fib6_node_kmem);
 }

--
```

_____

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