

# [PATCH 2.4.22-pre1][NET] timer cleanups

**Source:** <http://linux.derkeiler.com/Mailing-Lists/Kernel/2003-08/8104.html>

---

**From:** Vinay K Nallamothu (*vinay-rc\_at\_naturesoft.net*)

**Date:** 08/30/03

To: netdev@oss.sgi.com

Date: Sat, 30 Aug 2003 21:11:37 +0530

Hi,

This patch does the following timer code cleanup:

1. Change del\_timer/add\_timer to mod\_timer
2. Use static timer initialisation wherever applicable

```
drivers/net/hamradio/6pack.c | 38 ++++++++-----  
drivers/net/wan/sdla_ppp.c | 4 ----  
net/atm/lec.c | 5 -----  
net/ipv4/netfilter/ip_contrack_core.c | 12 +++++-----  
net/ipv4/netfilter/ip_fw_compat_redir.c | 12 +++-----  
net/ipv6/ip6_flowlabel.c | 12 ++++-----  
net/netrom/nr_loopback.c | 19 +++-----  
net/rose/rose_loopback.c | 20 +++-----  
net/sched/sch_cbq.c | 7 +-  
net/sched/sch_sfq.c | 11 +++++-----  
net/wanrouter/af_wanpipe.c | 7 +-----  
11 files changed, 46 insertions(+), 101 deletions(-)
```

```
diff -urN linux-2.4.22/drivers/net/hamradio/6pack.c linux-2.4.22-nvk/drivers/net/hamradio/6pack.c
```

```
---- linux-2.4.22/drivers/net/hamradio/6pack.c 2003-06-14 10:03:18.000000000 +0530
```

```
+++ linux-2.4.22-nvk/drivers/net/hamradio/6pack.c 2003-08-30 15:26:04.000000000 +0530
```

```
@@ -140,7 +140,7 @@
```

```
MODULE_PARM(sixpack_maxdev, "i");
```

```
MODULE_PARM_DESC(sixpack_maxdev, "number of 6PACK devices");
```

```
-static void sp_start_tx_timer(struct sixpack *);
```

```
+static inline void sp_start_tx_timer(struct sixpack *);
```

```
static void sp_xmit_on_air(unsigned long);
```

```
static void resync_tnc(unsigned long);
```

```
static void sixpack_decode(struct sixpack *, unsigned char[], int);
```

```
@@ -480,7 +480,12 @@
```

```
netif_start_queue(dev);
```

```
init_timer(&sp->tx_t);
```

```
+ sp->tx_t.data = (unsigned long) sp;
```

Linux-Kernel: [PATCH 2.4.22-pre1][NET] timer cleanups

```
+ sp->tx_t.function = sp_xmit_on_air;
+
+   init_timer(&sp->resync_t);
+ sp->resync_t.data = (unsigned long) sp;
+ sp->resync_t.function = resync_tnc;
+   return 0;
}
```

@@ -803,15 +808,10 @@

```
/* -----> 6pack timer interrupt handler and friends. <----- */
-static void sp_start_tx_timer(struct sixpack *sp)
+static inline void sp_start_tx_timer(struct sixpack *sp)
{
+   int when = sp->slottime;
-
- del_timer(&sp->tx_t);
- sp->tx_t.data = (unsigned long) sp;
- sp->tx_t.function = sp_xmit_on_air;
- sp->tx_t.expires = jiffies + ((when+1)*HZ)/100;
- add_timer(&sp->tx_t);
+ mod_timer(&sp->tx_t, jiffies + ((when+1)*HZ)/100);
}
```

@@ -883,12 +883,7 @@

```
sp->tty->driver.write(sp->tty, 0, &inbyte, 1);

- del_timer(&sp->resync_t);
- sp->resync_t.data = (unsigned long) sp;
- sp->resync_t.function = resync_tnc;
- sp->resync_t.expires = jiffies + SIXP_RESYNC_TIMEOUT;
- add_timer(&sp->resync_t);
-
+ mod_timer(&sp->resync_t.expires, jiffies + SIXP_RESYNC_TIMEOUT);
+   return 0;
}
```

@@ -940,13 +935,8 @@

```
/* if the state byte has been received, the TNC is present,
   so the resync timer can be reset. */

- if (sp->tnc_ok == 1) {
- del_timer(&sp->resync_t);
- sp->resync_t.data = (unsigned long) sp;
- sp->resync_t.function = resync_tnc;
- sp->resync_t.expires = jiffies + SIXP_INIT_RESYNC_TIMEOUT;
- add_timer(&sp->resync_t);
- }
```

## Linux-Kernel: [PATCH 2.4.22-pre1][NET] timer cleanups

```
+ if (sp->tnc_ok == 1)
+ mod_timer(&sp->resync_t, jiffies + SIXP_INIT_RESYNC_TIMEOUT);
```

```
    sp->status1 = cmd & SIXP_PRIO_DATA_MASK;
}
@@ -982,11 +972,7 @@
```

```
    /* Start resync timer again -- the TNC might be still absent */
```

```
- del_timer(&sp->resync_t);
- sp->resync_t.data = (unsigned long) sp;
- sp->resync_t.function = resync_tnc;
- sp->resync_t.expires = jiffies + SIXP_RESYNC_TIMEOUT;
- add_timer(&sp->resync_t);
+ mod_timer(&sp->resync_t, jiffies + SIXP_RESYNC_TIMEOUT);
}
```

```
diff -urN linux-2.4.22/drivers/net/wan/sdla_ppp.c linux-2.4.22-nvk/drivers/net/wan/sdla_ppp.c
--- linux-2.4.22/drivers/net/wan/sdla_ppp.c 2003-06-14 10:03:17.000000000 +0530
+++ linux-2.4.22-nvk/drivers/net/wan/sdla_ppp.c 2003-08-26 11:12:56.000000000 +0530
@@ -841,9 +841,7 @@
```

```
    /* Start the PPP configuration after 1sec delay.
```

```
    * This will give the interface initialization time
```

```
    * to finish its configuration */
```

```
- del_timer(&ppp_priv_area->poll_delay_timer);
- ppp_priv_area->poll_delay_timer.expires = jiffies+HZ;
- add_timer(&ppp_priv_area->poll_delay_timer);
+ mod_timer(&ppp_priv_area->poll_delay_timer, jiffies + HZ);
    return 0;
}
```

```
diff -urN linux-2.4.22/net/atm/lec.c linux-2.4.22-nvk/net/atm/lec.c
--- linux-2.4.22/net/atm/lec.c 2003-08-26 10:08:26.000000000 +0530
+++ linux-2.4.22-nvk/net/atm/lec.c 2003-08-30 16:37:57.000000000 +0530
@@ -1489,8 +1489,6 @@
```

```
    entry = (struct lec_arp_table *)data;
```

```
- del_timer(&entry->timer);
```

```
-
```

```
    DPRINTK("lec_arp_expire_arp\n");
```

```
    if (entry->status == ESI_ARP_PENDING) {
```

```
        if (entry->no_tries <= entry->priv->max_retry_count) {
```

```
@@ -1500,8 +1498,7 @@
```

```
            send_to_lecd(entry->priv, l_arp_xmt, entry->mac_addr, NULL, NULL);
```

```
            entry->no_tries++;
```

```
        }
```

```
- entry->timer.expires = jiffies + (1*HZ);
- add_timer(&entry->timer);
+ mod_timer(&entry->timer, jiffies + 1 * HZ);
```

[PATCH 2.4.22-pre1][NET] timer cleanups

Linux-Kernel: [PATCH 2.4.22-pre1][NET] timer cleanups

```

    }
}

diff -urN linux-2.4.22/net/ipv4/netfilter/ip_conntrack_core.c
linux-2.4.22-nvk/net/ipv4/netfilter/ip_conntrack_core.c
--- linux-2.4.22/net/ipv4/netfilter/ip_conntrack_core.c 2003-08-26 10:08:30.000000000 +0530
+++ linux-2.4.22-nvk/net/ipv4/netfilter/ip_conntrack_core.c 2003-08-26 21:54:45.000000000 +0530
@@ -954,13 +954,12 @@
     the data filled out by the helper over the old one */
     DEBUGP("expect_related: resent packet\n");
     if (related_to->helper->timeout) {
- if (!del_timer(&old->timeout)) {
+ if (!timer_pending(&old->timeout)) {
         /* expectation is dying. Fall through */
         old = NULL;
     } else {
- old->timeout.expires = jiffies +
- related_to->helper->timeout * HZ;
- add_timer(&old->timeout);
+ mod_timer(&old->timeout, jiffies +
+ related_to->helper->timeout * HZ);
     }
 }

@@ -1182,9 +1181,8 @@
     ct->timeout.expires = extra_jiffies;
     else {
         /* Need del_timer for race avoidance (may already be dying). */
- if (del_timer(&ct->timeout)) {
- ct->timeout.expires = jiffies + extra_jiffies;
- add_timer(&ct->timeout);
+ if (timer_pending(&ct->timeout)) {
+ mod_timer(&ct->timeout, extra_jiffies);
     }
 }
     WRITE_UNLOCK(&ip_conntrack_lock);
diff -urN linux-2.4.22/net/ipv4/netfilter/ip_fw_compat_redir.c
linux-2.4.22-nvk/net/ipv4/netfilter/ip_fw_compat_redir.c
--- linux-2.4.22/net/ipv4/netfilter/ip_fw_compat_redir.c 2003-06-14 10:03:12.000000000 +0530
+++ linux-2.4.22-nvk/net/ipv4/netfilter/ip_fw_compat_redir.c 2003-08-30 14:41:23.000000000 +0530
@@ -270,10 +270,8 @@
     if (redir) {
         DEBUGP("Doing tcp redirect again.\n");
         do_tcp_redir(skb, redir);
- if (del_timer(&redir->destroyme)) {
- redir->destroyme.expires = jiffies + REDIR_TIMEOUT;
- add_timer(&redir->destroyme);
- }
+ if (timer_pending(&redir->destroyme))
+ mod_timer(&redir->destroyme, jiffies + REDIR_TIMEOUT);
     }
}

```

Linux-Kernel: [PATCH 2.4.22-pre1][NET] timer cleanups

```

        UNLOCK_BH(&redir_lock);
    }
@@ -298,10 +296,8 @@
    if (redir) {
        DEBUGP("Doing tcp unredirect.\n");
        do_tcp_unredir(skb, redir);
- if (del_timer(&redir->destroyme)) {
- redir->destroyme.expires = jiffies + REDIR_TIMEOUT;
- add_timer(&redir->destroyme);
- }
+ if (timer_pending(&redir->destroyme))
+ mod_timer(&redir->destroyme, jiffies + REDIR_TIMEOUT);
    }
    UNLOCK_BH(&redir_lock);
}
diff -urN linux-2.4.22/net/ipv6/ip6_flowlabel.c linux-2.4.22-nvk/net/ipv6/ip6_flowlabel.c
--- linux-2.4.22/net/ipv6/ip6_flowlabel.c 2003-06-14 10:03:12.000000000 +0530
+++ linux-2.4.22-nvk/net/ipv6/ip6_flowlabel.c 2003-08-26 21:10:46.000000000 +0530
@@ -48,7 +48,8 @@
static atomic_t fl_size = ATOMIC_INIT(0);
static struct ip6_flowlabel *fl_ht[FL_HASH_MASK+1];

-static struct timer_list ip6_fl_gc_timer;
+static void ip6_fl_gc(unsigned long dummy);
+static struct timer_list ip6_fl_gc_timer = TIMER_INITIALIZER(ip6_fl_gc, 0, 0);

/* FL hash table lock: it protects only of GC */

@@ -103,10 +104,9 @@
        fl->opt = NULL;
        kfree(opt);
    }
- if (!del_timer(&ip6_fl_gc_timer) ||
- (long)(ip6_fl_gc_timer.expires - ttd) > 0)
- ip6_fl_gc_timer.expires = ttd;
- add_timer(&ip6_fl_gc_timer);
+ if (!timer_pending(&ip6_fl_gc_timer) ||
+ time_after(ip6_fl_gc_timer.expires, ttd))
+ mod_timer(&ip6_fl_gc_timer, ttd);
    }
}

@@ -609,8 +609,6 @@

void ip6_flowlabel_init()
{
- init_timer(&ip6_fl_gc_timer);
- ip6_fl_gc_timer.function = ip6_fl_gc;
#ifdef CONFIG_PROC_FS
    create_proc_read_entry("net/ipv6_flowlabel", 0, 0, ip6_fl_read_proc, NULL);
#endif
}

```

## Linux-Kernel: [PATCH 2.4.22-pre1][NET] timer cleanups

```
diff -urN linux-2.4.22/net/netrom/nr_loopback.c linux-2.4.22-nvk/net/netrom/nr_loopback.c
--- linux-2.4.22/net/netrom/nr_loopback.c 2003-06-14 10:03:12.000000000 +0530
+++ linux-2.4.22-nvk/net/netrom/nr_loopback.c 2003-08-30 16:21:47.000000000 +0530
@@ -23,8 +23,10 @@
#include <net/netrom.h>
#include <linux/init.h>

+static void nr_loopback_timer(unsigned long);
+
static struct sk_buff_head loopback_queue;
-static struct timer_list loopback_timer;
+static struct timer_list loopback_timer = TIMER_INITIALIZER(nr_loopback_timer, 0, 0);

static void nr_set_loopback_timer(void);

@@ -51,26 +53,13 @@
    skb_queue_tail(&loopback_queue, skb);

    if (!nr_loopback_running())
- nr_set_loopback_timer();
+ mod_timer(&loopback_timer, jiffies + 10);
    }

    kfree_skb(skb);
    return 1;
}

-static void nr_loopback_timer(unsigned long);
-
-static void nr_set_loopback_timer(void)
-{
- del_timer(&loopback_timer);
-
- loopback_timer.data = 0;
- loopback_timer.function = &nr_loopback_timer;
- loopback_timer.expires = jiffies + 10;
-
- add_timer(&loopback_timer);
-}
-
static void nr_loopback_timer(unsigned long param)
{
    struct sk_buff *skb;
diff -urN linux-2.4.22/net/rose/rose_loopback.c linux-2.4.22-nvk/net/rose/rose_loopback.c
--- linux-2.4.22/net/rose/rose_loopback.c 2003-06-14 10:03:13.000000000 +0530
+++ linux-2.4.22-nvk/net/rose/rose_loopback.c 2003-08-30 15:41:40.000000000 +0530
@@ -22,10 +22,10 @@
#include <net/rose.h>
#include <linux/init.h>

-static struct sk_buff_head loopback_queue;
```

## Linux-Kernel: [PATCH 2.4.22-pre1][NET] timer cleanups

```

-static struct timer_list loopback_timer;
+static void rose_loopback_timer(unsigned long);

-static void rose_set_loopback_timer(void);
+static struct sk_buff_head loopback_queue;
+static struct timer_list loopback_timer = TIMER_INITIALIZER(rose_loopback_timer, 0, 0);

void rose_loopback_init(void)
{
@@ -51,24 +51,12 @@
    skb_queue_tail(&loopback_queue, skbn);

    if (!rose_loopback_running())
- rose_set_loopback_timer();
+ mod_timer(&loopback_timer, jiffies + 10);
}

return 1;
}

-static void rose_loopback_timer(unsigned long);
-
-static void rose_set_loopback_timer(void)
-{
- del_timer(&loopback_timer);
-
- loopback_timer.data = 0;
- loopback_timer.function = &rose_loopback_timer;
- loopback_timer.expires = jiffies + 10;
-
- add_timer(&loopback_timer);
-}

static void rose_loopback_timer(unsigned long param)
{
diff -urN linux-2.4.22/net/sched/sch_cbq.c linux-2.4.22-nvk/net/sched/sch_cbq.c
--- linux-2.4.22/net/sched/sch_cbq.c 2003-08-26 10:08:36.000000000 +0530
+++ linux-2.4.22-nvk/net/sched/sch_cbq.c 2003-08-30 14:56:04.000000000 +0530
@@ -554,10 +554,9 @@
    cl->penalized = sched;
    cl->cpriority = TC_CBQ_MAXPRIO;
    q->pmask |= (1<<TC_CBQ_MAXPRIO);
- if (del_timer(&q->delay_timer) &&
- (long)(q->delay_timer.expires - sched) > 0)
- q->delay_timer.expires = sched;
+ add_timer(&q->delay_timer);
+ if (timer_pending(&q->delay_timer) &&
+ time_after(q->delay_timer.expires, sched))
+ mod_timer(&q->delay_timer, sched);
    cl->delayed = 1;
    cl->xstats.overactions++;

```

## Linux-Kernel: [PATCH 2.4.22-pre1][NET] timer cleanups

```
        return;
diff -urN linux-2.4.22/net/sched/sch_sfq.c linux-2.4.22-nvk/net/sched/sch_sfq.c
--- linux-2.4.22/net/sched/sch_sfq.c 2003-08-26 10:08:36.000000000 +0530
+++ linux-2.4.22-nvk/net/sched/sch_sfq.c 2003-08-30 14:51:57.000000000 +0530
@@ -371,7 +371,6 @@
     struct sfq_sched_data *q = (struct sfq_sched_data *)sch->data;

     q->perturbation = net_random()&0x1F;
- q->perturb_timer.expires = jiffies + q->perturb_period;

     if (q->perturb_period) {
         q->perturb_timer.expires = jiffies + q->perturb_period;
@@ -396,11 +395,11 @@
         while (sch->q.qlen >= q->limit-1)
             sfq_drop(sch);

- del_timer(&q->perturb_timer);
- if (q->perturb_period) {
- q->perturb_timer.expires = jiffies + q->perturb_period;
- add_timer(&q->perturb_timer);
- }
+ if (q->perturb_period)
+ mod_timer(&q->perturb_timer, jiffies + q->perturb_period);
+ else
+ del_timer(&q->perturb_timer);
+
     sch_tree_unlock(sch);
     return 0;
}
diff -urN linux-2.4.22/net/wanrouter/af_wanpipe.c linux-2.4.22-nvk/net/wanrouter/af_wanpipe.c
--- linux-2.4.22/net/wanrouter/af_wanpipe.c 2003-06-14 10:03:13.000000000 +0530
+++ linux-2.4.22-nvk/net/wanrouter/af_wanpipe.c 2003-08-26 11:21:31.000000000 +0530
@@ -649,11 +649,8 @@
     skb_queue_tail(&sk->write_queue,skb);
     atomic_inc(&sk->protinfo.af_wanpipe->packet_sent);

- if (!test_and_set_bit(0,&sk->protinfo.af_wanpipe->timer)){
- del_timer(&sk->protinfo.af_wanpipe->tx_timer);
- sk->protinfo.af_wanpipe->tx_timer.expires=jiffies+1;
- add_timer(&sk->protinfo.af_wanpipe->tx_timer);
- }
+ if (!test_and_set_bit(0,&sk->protinfo.af_wanpipe->timer))
+ mod_timer(&sk->protinfo.af_wanpipe->tx_timer, jiffies + 1);

     return(len);

-

```

To unsubscribe from this list: send the line "unsubscribe linux-kernel" in the body of a message to majordomo@vger.kernel.org  
More majordomo info at <http://vger.kernel.org/majordomo-info.html>

Linux–Kernel: [PATCH 2.4.22–pre1][NET] timer cleanups

Please read the FAQ at <http://www.tux.org/lkml/>