
Subject: [PATCH net-2.6.25 3/3] Uninline the inet_twsk_put function
Posted by [Pavel Emelianov](#) on Wed, 19 Dec 2007 10:56:13 GMT
[View Forum Message](#) <> [Reply to Message](#)

This one is not that big, but is widely used: saves 1200 bytes
from net/ipv4/built-in.o

```
add/remove: 1/0 grow/shrink: 1/12 up/down: 97/-1300 (-1203)
function          old    new  delta
inet_twsk_put      -     87   +87
__inet_lookup_listener    274   284   +10
tcp_sacktag_write_queue    2255  2254   -1
tcp_time_wait      482   411   -71
__inet_check_established    796   722   -74
tcp_v4_err         973   898   -75
__inet_twsk_kill    230   154   -76
inet_twsk_deschedule    180   103   -77
tcp_v4_do_rcv      462   384   -78
inet_hash_connect    686   607   -79
inet_twdr_do_twkill_work    236   150   -86
inet_twdr_twcald_tick    395   307   -88
tcp_v4_rcv         1744  1480  -264
tcp_timewait_state_process    975   644  -331
```

Export it for ipv6 module.

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

```
---
include/net/inet_timewait_sock.h | 14 +-----
net/ipv4/inet_timewait_sock.c   | 15 ++++++
2 files changed, 16 insertions(+), 13 deletions(-)
```

```
diff --git a/include/net/inet_timewait_sock.h b/include/net/inet_timewait_sock.h
index abaff05..67e9250 100644
--- a/include/net/inet_timewait_sock.h
+++ b/include/net/inet_timewait_sock.h
@@ -193,19 +193,7 @@ static inline __be32 inet_rcv_saddr(const struct sock *sk)
     inet_sk(sk)->rcv_saddr : inet_twsk(sk)->tw_rcv_saddr;
 }

-static inline void inet_twsk_put(struct inet_timewait_sock *tw)
-{
- if (atomic_dec_and_test(&tw->tw_refcnt)) {
- struct module *owner = tw->tw_prot->owner;
- twsk_destructor((struct sock *)tw);
-#ifdef SOCK_REFCNT_DEBUG
- printk(KERN_DEBUG "%s timewait_sock %p released\n",
```

```

-      tw->tw_prot->name, tw);
-#endif
-   kmem_cache_free(tw->tw_prot->twsk_prot->twsk_slab, tw);
-   module_put(owner);
- }
-}
+extern void inet_twsk_put(struct inet_timewait_sock *tw);

extern struct inet_timewait_sock *inet_twsk_alloc(const struct sock *sk,
          const int state);
diff --git a/net/ipv4/inet_timewait_sock.c b/net/ipv4/inet_timewait_sock.c
index d43e787..1b7db42 100644
--- a/net/ipv4/inet_timewait_sock.c
+++ b/net/ipv4/inet_timewait_sock.c
@@ -48,6 +48,21 @@ static void __inet_twsk_kill(struct inet_timewait_sock *tw,
    inet_twsk_put(tw);
}

+void inet_twsk_put(struct inet_timewait_sock *tw)
+{
+ if (atomic_dec_and_test(&tw->tw_refcnt)) {
+  struct module *owner = tw->tw_prot->owner;
+  twsk_destructor((struct sock *)tw);
+ #ifdef SOCK_REFCNT_DEBUG
+  printk(KERN_DEBUG "%s timewait_sock %p released\n",
+    tw->tw_prot->name, tw);
+ #endif
+  kmem_cache_free(tw->tw_prot->twsk_prot->twsk_slab, tw);
+  module_put(owner);
+ }
+}
+EXPORT_SYMBOL_GPL(inet_twsk_put);
+
+/*
+ * Enter the time wait state. This is called with locally disabled BH.
+ * Essentially we whip up a timewait bucket, copy the relevant info into it
+ */
--
1.5.3.4

```
