Subject: [PATCH 1/2][IPV4] Consolidate the ip cork destruction in ip_output.c
Posted by Pavel Emelianov on Fri, 02 Nov 2007 16:04:06 GMT
View Forum Message <> Reply to Message

The ip_push_pending_frames and ip_flush_pending_frames do the
same things to flush the sock's cork. Move this into a separate
function and save ~80 bytes from the .text

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

```
diff --git a/net/ipv4/ip_output.c b/net/ipv4/ip_output.c
index e5f7dc2..fd99fbd 100644
--- a/net/ipv4/ip_output.c
+++ b/net/ipv4/ip_output.c
@@ -1183,6 +1183,17 @@ error:
  return err;
 }

+static void ip_cork_release(struct inet_sock *inet)
+{
+ inet->cork.flags &= ~IPCORK_OPT;
+ kfree(inet->cork.opt);
+ inet->cork.opt = NULL;
+ if (inet->cork.rt) {
+  ip_rt_put(inet->cork.rt);
+  inet->cork.rt = NULL;
+ }
+}
+
 /*
  * Combined all pending IP fragments on the socket as one IP datagram
  * and push them out.
@@ -1276,13 +1287,7 @@ int ip_push_pending_frames(struct sock *sk)
  }

 out:
- inet->cork.flags &= ~IPCORK_OPT;
- kfree(inet->cork.opt);
- inet->cork.opt = NULL;
- if (inet->cork.rt) {
-  ip_rt_put(inet->cork.rt);
-  inet->cork.rt = NULL;
- }
+ ip_cork_release(inet);
  return err;
```

```
 error:
@@ -1295,19 +1300,12 @@ error:
  */
 void ip_flush_pending_frames(struct sock *sk)
 {
- struct inet_sock *inet = inet_sk(sk);
  struct sk_buff *skb;

  while ((skb = __skb_dequeue_tail(&sk->sk_write_queue)) != NULL)
   kfree_skb(skb);

- inet->cork.flags &= ~IPCORK_OPT;
- kfree(inet->cork.opt);
- inet->cork.opt = NULL;
- if (inet->cork.rt) {
-  ip_rt_put(inet->cork.rt);
-  inet->cork.rt = NULL;
- }
+ ip_cork_release(inet_sk(sk));
 }
```