Subject: [PATCH][NET-2.6.25] Move sock_valbool_flag to socket.c
Posted by Pavel Emelianov on Thu, 15 Nov 2007 09:43:51 GMT
View Forum Message <> Reply to Message

The sock_valbool_flag() helper is used in setsockopt to
set or reset some flag on the sock. This helper is required
in the net/socket.c only, so move it there.

Besides, patch two places in sys_setsockopt() that repeat
this helper functionality manually.

Since this is not a bugfix, but a trivial cleanup, I
prepared this patch against net-2.6.25, but it also
applies (with a single offset) to the latest net-2.6.

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

```
diff --git a/include/net/sock.h b/include/net/sock.h
index cfb946a..80ca671 100644
--- a/include/net/sock.h
+++ b/include/net/sock.h
@@ -1393,14 +1393,6 @@ extern int net_msg_warn;
     lock_sock(sk); \
     }

-static inline void sock_valbool_flag(struct sock *sk, int bit, int valbool)
-{
- if (valbool)
-  sock_set_flag(sk, bit);
- else
-  sock_reset_flag(sk, bit);
-}
-
 extern __u32 sysctl_wmem_max;
 extern __u32 sysctl_rmem_max;

diff --git a/net/core/sock.c b/net/core/sock.c
index 2029d09..98b243a 100644
--- a/net/core/sock.c
+++ b/net/core/sock.c
@@ -419,6 +419,14 @@ out:
  return ret;
 }

+static inline void sock_valbool_flag(struct sock *sk, int bit, int valbool)
+{
```

```
+ if (valbool)
+  sock_set_flag(sk, bit);
+ else
+  sock_reset_flag(sk, bit);
+}
+
 /*
  * This is meant for all protocols to use and covers goings on
  * at the socket level. Everything here is generic.
@@ -463,11 +471,8 @@ int sock_setsockopt(struct socket *sock, int level, int optname,
  case SO_DEBUG:
   if (val && !capable(CAP_NET_ADMIN)) {
    ret = -EACCES;
-  }
-  else if (valbool)
-   sock_set_flag(sk, SOCK_DBG);
-  else
-   sock_reset_flag(sk, SOCK_DBG);
+  } else
+   sock_valbool_flag(sk, SOCK_DBG, valbool);
   break;
  case SO_REUSEADDR:
   sk->sk_reuse = valbool;
@@ -477,10 +482,7 @@ int sock_setsockopt(struct socket *sock, int level, int optname,
   ret = -ENOPROTOOPT;
   break;
  case SO_DONTROUTE:
-  if (valbool)
-   sock_set_flag(sk, SOCK_LOCALROUTE);
-  else
-   sock_reset_flag(sk, SOCK_LOCALROUTE);
+  sock_valbool_flag(sk, SOCK_LOCALROUTE, valbool);
   break;
  case SO_BROADCAST:
   sock_valbool_flag(sk, SOCK_BROADCAST, valbool);
```