
Subject: [PATCH 1/4] Introduce the sk_detach_filter() call
Posted by [Pavel Emelianov](#) on Wed, 17 Oct 2007 09:47:31 GMT
[View Forum Message](#) <[Reply to Message](#)

Filter is attached in a separate function, so do the same for filter detaching.

This also removes one variable sock_setsockopt().

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

```
diff --git a/include/linux/filter.h b/include/linux/filter.h
index 91b2e3b..ddfa037 100644
--- a/include/linux/filter.h
+++ b/include/linux/filter.h
@@ -146,6 +146,7 @@ struct sock;

extern unsigned int sk_run_filter(struct sk_buff *skb, struct sock_filter *filter, intflen);
extern int sk_attach_filter(struct sock_fprog *fprog, struct sock *sk);
+extern int sk_detach_filter(struct sock *sk);
extern int sk_chk_filter(struct sock_filter *filter, intflen);
#endif /* __KERNEL__ */
```

```
diff --git a/net/core/filter.c b/net/core/filter.c
index bd903aa..fd60758 100644
--- a/net/core/filter.c
+++ b/net/core/filter.c
@@ -433,5 +433,21 @@ int sk_attach_filter(struct sock_fprog *fprog, struct sock *sk)
    return err;
}

+int sk_detach_filter(struct sock *sk)
+{
+    int ret = -ENOENT;
+    struct sock_filter *filter;
+
+    rCU_read_lock_bh();
+    filter = rCU_dereference(sk->sk_filter);
+    if (filter) {
+        rCU_assign_pointer(sk->sk_filter, NULL);
+        sk_filter_release(sk, filter);
+        ret = 0;
+    }
+    rCU_read_unlock_bh();
+    return ret;
+}
```

```

+
EXPORT_SYMBOL(sk_chk_filter);
EXPORT_SYMBOL(sk_run_filter);
diff --git a/net/core/sock.c b/net/core/sock.c
index d45ecdc..0710138 100644
--- a/net/core/sock.c
+++ b/net/core/sock.c
@@ -428,7 +428,6 @@ int sock_setsockopt(struct socket *sock, int level, int optname,
    char __user *optval, int optlen)
{
    struct sock *sk=sock->sk;
-   struct sk_filter *filter;
    int val;
    int valbool;
    struct linger ling;
@@ -652,16 +651,7 @@ set_rcvbuf:
    break;

    case SO_DETACH_FILTER:
-   rCU_read_lock_bh();
-   filter = rCU_dereference(sk->sk_filter);
-   if (filter) {
-   rCU_assign_pointer(sk->sk_filter, NULL);
-   sk_filter_release(sk, filter);
-   rCU_read_unlock_bh();
-   break;
-   }
-   rCU_read_unlock_bh();
-   ret = -ENONET;
+   ret = sk_detach_filter(sk);
    break;

    case SO_PASSSEC:
--
```

1.5.3.4
