Subject: [PATCH 2/4] Move the filter releasing into a separate call
Posted by Pavel Emelianov on Wed, 17 Oct 2007 09:49:38 GMT
View Forum Message <> Reply to Message

This is done merely as a preparation for the fix.

The sk_filter_uncharge() unaccounts the filter memory and calls
the sk_filter_release(), which in turn decrements the refcount
anf frees the filter.

The latter function will be required separately.

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

```
diff --git a/include/net/sock.h b/include/net/sock.h
index 453c79d..b9cfe12 100644
--- a/include/net/sock.h
+++ b/include/net/sock.h
@@ -922,14 +922,18 @@ static inline void sk_filter_rcu_free(struct rcu_head *rcu)
  * Remove a filter from a socket and release its resources.
  */

-static inline void sk_filter_release(struct sock *sk, struct sk_filter *fp)
+static inline void sk_filter_release(struct sk_filter *fp)
+{
+ if (atomic_dec_and_test(&fp->refcnt))
+  call_rcu_bh(&fp->rcu, sk_filter_rcu_free);
+}
+
+static inline void sk_filter_uncharge(struct sock *sk, struct sk_filter *fp)
 {
  unsigned int size = sk_filter_len(fp);

  atomic_sub(size, &sk->sk_omem_alloc);
-
- if (atomic_dec_and_test(&fp->refcnt))
-  call_rcu_bh(&fp->rcu, sk_filter_rcu_free);
+ sk_filter_release(fp);
 }

 static inline void sk_filter_charge(struct sock *sk, struct sk_filter *fp)
diff --git a/net/core/filter.c b/net/core/filter.c
index fd60758..2be1830 100644
--- a/net/core/filter.c
+++ b/net/core/filter.c
@@ -429,7 +429,7 @@ int sk_attach_filter(struct sock_fprog *fprog, struct sock *sk)
```

```
  }

  if (fp)
-  sk_filter_release(sk, fp);
+  sk_filter_uncharge(sk, fp);
  return err;
 }

@@ -442,7 +442,7 @@ int sk_detach_filter(struct sock *sk)
  filter = rcu_dereference(sk->sk_filter);
  if (filter) {
   rcu_assign_pointer(sk->sk_filter, NULL);
-  sk_filter_release(sk, filter);
+  sk_filter_uncharge(sk, filter);
   ret = 0;
  }
  rcu_read_unlock_bh();
diff --git a/net/core/sock.c b/net/core/sock.c
index 0710138..d292b41 100644
--- a/net/core/sock.c
+++ b/net/core/sock.c
@@ -915,7 +915,7 @@ void sk_free(struct sock *sk)

  filter = rcu_dereference(sk->sk_filter);
  if (filter) {
-  sk_filter_release(sk, filter);
+  sk_filter_uncharge(sk, filter);
   rcu_assign_pointer(sk->sk_filter, NULL);
  }

--
1.5.3.4
```