Subject: [PATCH] Consolidate checking for tcp orphan count being too big
Posted by xemul on Fri, 04 May 2007 11:05:18 GMT
View Forum Message <> Reply to Message

tcp_out_of_resources() and tcp_close() perform the
same checking of number of orphan sockets. Move this
code into common place.

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

```
diff --git a/include/net/tcp.h b/include/net/tcp.h
index e22b4f0..a8af9ae 100644
--- a/include/net/tcp.h
+++ b/include/net/tcp.h
@@ -254,6 +254,12 @@ static inline int between(__u32 seq1, __
  return seq3 - seq2 >= seq1 - seq2;
 }

+static inline int tcp_too_many_orphans(struct sock *sk, int num)
+{
+ return (num > sysctl_tcp_max_orphans) ||
+  (sk->sk_wmem_queued > SOCK_MIN_SNDBUF &&
+   atomic_read(&tcp_memory_allocated) > sysctl_tcp_mem[2]);
+}

 extern struct proto tcp_prot;

diff --git a/net/ipv4/tcp.c b/net/ipv4/tcp.c
index 8b124ea..8fbe550 100644
--- a/net/ipv4/tcp.c
+++ b/net/ipv4/tcp.c
@@ -1675,9 +1675,8 @@ adjudge_to_death:
  }
  if (sk->sk_state != TCP_CLOSE) {
   sk_stream_mem_reclaim(sk);
-  if (atomic_read(sk->sk_prot->orphan_count) > sysctl_tcp_max_orphans ||
-      (sk->sk_wmem_queued > SOCK_MIN_SNDBUF &&
-       atomic_read(&tcp_memory_allocated) > sysctl_tcp_mem[2])) {
+  if (tcp_too_many_orphans(sk,
+    atomic_read(sk->sk_prot->orphan_count))) {
    if (net_ratelimit())
     printk(KERN_INFO "TCP: too many of orphaned "
            "sockets\n");
diff --git a/net/ipv4/tcp_timer.c b/net/ipv4/tcp_timer.c
index 2ca97b2..e613401 100644
--- a/net/ipv4/tcp_timer.c
```

```
+++ b/net/ipv4/tcp_timer.c
@@ -78,9 +78,7 @@ static int tcp_out_of_resources(struct s
  if (sk->sk_err_soft)
   orphans <<= 1;

- if (orphans >= sysctl_tcp_max_orphans ||
-     (sk->sk_wmem_queued > SOCK_MIN_SNDBUF &&
-      atomic_read(&tcp_memory_allocated) > sysctl_tcp_mem[2])) {
+ if (tcp_too_many_orphans(sk, orphans)) {
   if (net_ratelimit())
    printk(KERN_INFO "Out of socket memory\n");
```