Subject: [PATCH net-2.6.25 1/3][ROUTE] Collect proc-related functions together
Posted by Pavel Emelianov on Wed, 05 Dec 2007 17:04:16 GMT
View Forum Message <> Reply to Message

The net/ipv4/route.c file declares some entries for proc
to dump some routing info. The reading functions are
scattered over this file - collect them together.

Besides, remove a useless IP_RT_ACCT_CPU macro.

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

```
diff --git a/net/ipv4/route.c b/net/ipv4/route.c
index 134cab5..cc5b730 100644
--- a/net/ipv4/route.c
+++ b/net/ipv4/route.c
@@ -481,6 +481,45 @@ static const struct file_operations rt_cpu_seq_fops = {
  .release = seq_release,
 };

+#ifdef CONFIG_NET_CLS_ROUTE
+static int ip_rt_acct_read(char *buffer, char **start, off_t offset,
+      int length, int *eof, void *data)
+{
+ unsigned int i;
+
+ if ((offset & 3) || (length & 3))
+  return -EIO;
+
+ if (offset >= sizeof(struct ip_rt_acct) * 256) {
+  *eof = 1;
+  return 0;
+ }
+
+ if (offset + length >= sizeof(struct ip_rt_acct) * 256) {
+  length = sizeof(struct ip_rt_acct) * 256 - offset;
+  *eof = 1;
+ }
+
+ offset /= sizeof(u32);
+
+ if (length > 0) {
+  u32 *dst = (u32 *) buffer;
+
+  *start = buffer;
+  memset(dst, 0, length);
```

```
+
+  for_each_possible_cpu(i) {
+   unsigned int j;
+   u32 *src;
+
+   src = ((u32 *) per_cpu_ptr(ip_rt_acct, i)) + offset;
+   for (j = 0; j < length/4; j++)
+    dst[j] += src[j];
+  }
+ }
+ return length;
+}
+#endif
 #endif /* CONFIG_PROC_FS */

 static __inline__ void rt_free(struct rtable *rt)
@@ -2895,48 +2934,6 @@ ctl_table ipv4_route_table[] = {

 #ifdef CONFIG_NET_CLS_ROUTE
 struct ip_rt_acct *ip_rt_acct __read_mostly;
-
-/* IP route accounting ptr for this logical cpu number. */
-#define IP_RT_ACCT_CPU(cpu) (per_cpu_ptr(ip_rt_acct, cpu))
-
-#ifdef CONFIG_PROC_FS
-static int ip_rt_acct_read(char *buffer, char **start, off_t offset,
-      int length, int *eof, void *data)
-{
- unsigned int i;
-
- if ((offset & 3) || (length & 3))
-  return -EIO;
-
- if (offset >= sizeof(struct ip_rt_acct) * 256) {
-  *eof = 1;
-  return 0;
- }
-
- if (offset + length >= sizeof(struct ip_rt_acct) * 256) {
-  length = sizeof(struct ip_rt_acct) * 256 - offset;
-  *eof = 1;
- }
-
- offset /= sizeof(u32);
-
- if (length > 0) {
-  u32 *dst = (u32 *) buffer;
-
```

```
-  *start = buffer;
-  memset(dst, 0, length);
-
-  for_each_possible_cpu(i) {
-   unsigned int j;
-   u32 *src = ((u32 *) IP_RT_ACCT_CPU(i)) + offset;
-
-   for (j = 0; j < length/4; j++)
-    dst[j] += src[j];
-  }
- }
- return length;
-}
-#endif /* CONFIG_PROC_FS */
 #endif /* CONFIG_NET_CLS_ROUTE */

 static __initdata unsigned long rhash_entries;
--
1.5.3.4
```