
Subject: [PATCH net-next 9/9] netns: selective flush of rt_cache

Posted by [den](#) on Fri, 04 Jul 2008 13:17:12 GMT

[View Forum Message](#) <> [Reply to Message](#)

dst cache is marked as expired on the per/namespace basis by previous path. Right now we have to implement selective cache shrinking. This procedure has been ported from older OpenVz codebase.

Signed-off-by: Denis V. Lunev <den@openvz.org>

net/ipv4/route.c | 31 ++++++
1 files changed, 30 insertions(+), 1 deletions(-)

diff --git a/net/ipv4/route.c b/net/ipv4/route.c

index 67c3ed7..113cd25 100644

--- a/net/ipv4/route.c

+++ b/net/ipv4/route.c

@@ -699,6 +699,7 @@ static void rt_do_flush(int process_context)

{
 unsigned int i;
 struct rtable *rth, *next;
+ struct rtable * tail;

for (i = 0; i <= rt_hash_mask; i++) {
 if (process_context && need_resched())

@@ -708,11 +709,39 @@ static void rt_do_flush(int process_context)
 continue;

spin_lock_bh(rt_hash_lock_addr(i));

+#ifdef CONFIG_NET_NS

+ {
+ struct rtable ** prev, * p;
+
+ rth = rt_hash_table[i].chain;
+
+ /* defer releasing the head of the list after spin_unlock */
+ for (tail = rth; tail; tail = tail->u.dst.rt_next)
+ if (!rt_is_expired(tail))
+ break;
+ if (rth != tail)
+ rt_hash_table[i].chain = tail;
+
+ /* call rt_free on entries after the tail requiring flush */
+ prev = &rt_hash_table[i].chain;
+ for (p = *prev; p; p = next) {
+ next = p->u.dst.rt_next;
+ if (!rt_is_expired(p)) {
+ prev = &p->u.dst.rt_next;

```
+ } else {
+   *prev = next;
+   rt_free(p);
+ }
+ }
+ }
+#else
+   rth = rt_hash_table[i].chain;
+   rt_hash_table[i].chain = NULL;
+   tail = NULL;
+#endif
+   spin_unlock_bh(rt_hash_lock_addr(i));

- for (; rth; rth = next) {
+ for (; rth != tail; rth = next) {
+   next = rth->u.dst.rt_next;
+   rt_free(rth);
+ }
--
1.5.3.rc5
```

Containers mailing list
Containers@lists.linux-foundation.org
<https://lists.linux-foundation.org/mailman/listinfo/containers>
