Subject: [PATCH] [NETNS49] Add struct net to flush_cache in fib_rules_ops
Posted by den on Thu, 18 Oct 2007 10:00:33 GMT
View Forum Message <> Reply to Message

commit d5daed837d7b0d2492a4226eea248cc7d92bcf81
Author: Denis V. Lunev <den@openvz.org>
Date: Thu Oct 18 12:41:20 2007 +0400

    Add struct net to flush_cache in fib_rules_ops

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

```
diff --git a/include/net/fib_rules.h b/include/net/fib_rules.h
index ba8caa9..163c920 100644
--- a/include/net/fib_rules.h
+++ b/include/net/fib_rules.h
@@ -61,7 +61,8 @@ struct fib_rules_ops

  /* Called after modifications to the rules set, must flush
   * the route cache if one exists. */
- void   (*flush_cache)(struct fib_rules_ops *ops);
+ void   (*flush_cache)(struct net *net,
+       struct fib_rules_ops *ops);

  int    nlgroup;
  const struct nla_policy *policy;
diff --git a/net/core/fib_rules.c b/net/core/fib_rules.c
index 407d4bf..220bee6 100644
--- a/net/core/fib_rules.c
+++ b/net/core/fib_rules.c
@@ -43,10 +43,10 @@ static void rules_ops_put(struct fib_rules_ops *ops)
   module_put(ops->owner);
 }

-static void flush_route_cache(struct fib_rules_ops *ops)
+static void flush_route_cache(struct net *net, struct fib_rules_ops *ops)
 {
  if (ops->flush_cache)
-  ops->flush_cache(ops);
+  ops->flush_cache(net, ops);
 }

 int fib_rules_register(struct net *net, struct fib_rules_ops *ops)
@@ -323,7 +323,7 @@ static int fib_nl_newrule(struct sk_buff *skb, struct nlmsghdr* nlh, void
*arg)
   list_add_rcu(&rule->list, ops->rules_list);

  notify_rule_change(net, RTM_NEWRULE, rule, ops, nlh, NETLINK_CB(skb).pid);
```

```
- flush_route_cache(ops);
+ flush_route_cache(net, ops);
  rules_ops_put(ops);
  return 0;

@@ -415,7 +415,7 @@ static int fib_nl_delrule(struct sk_buff *skb, struct nlmsghdr* nlh, void
*arg)
   notify_rule_change(net, RTM_DELRULE, rule, ops, nlh,
        NETLINK_CB(skb).pid);
   fib_rule_put(rule);
-  flush_route_cache(ops);
+  flush_route_cache(net, ops);
   rules_ops_put(ops);
   return 0;
  }
diff --git a/net/decnet/dn_rules.c b/net/decnet/dn_rules.c
index 9f7a206..2850b66 100644
--- a/net/decnet/dn_rules.c
+++ b/net/decnet/dn_rules.c
@@ -243,7 +243,7 @@ static u32 dn_fib_rule_default_pref(struct fib_rules_ops *ops)
  return 0;
 }

-static void dn_fib_rule_flush_cache(struct fib_rules_ops *ops)
+static void dn_fib_rule_flush_cache(struct net *net, struct fib_rules_ops *ops)
 {
  dn_rt_cache_flush(-1);
 }
diff --git a/net/ipv4/fib_rules.c b/net/ipv4/fib_rules.c
index 6bf5b33..c93b278 100644
--- a/net/ipv4/fib_rules.c
+++ b/net/ipv4/fib_rules.c
@@ -279,7 +279,7 @@ static size_t fib4_rule_nlmsg_payload(struct fib_rule *rule)
        + nla_total_size(4); /* flow */
 }

-static void fib4_rule_flush_cache(struct fib_rules_ops *ops)
+static void fib4_rule_flush_cache(struct net *net, struct fib_rules_ops *ops)
 {
  rt_cache_flush(-1);
 }
```