
Subject: [PATCH 2/12 net-2.6.25] [IPV4]: Consolidate fib_select_default.

Posted by [den](#) on Tue, 22 Jan 2008 15:59:19 GMT

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

The difference in the implementation of the fib_select_default when CONFIG_IP_MULTIPLE_TABLES is (not) defined looks negligible. Consolidate it and place into fib_frontend.c.

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

```
include/net/ip_fib.h | 10 +-----
net/ipv4/fib_frontend.c | 14 ++++++
net/ipv4/fib_rules.c | 10 -----
3 files changed, 15 insertions(+), 19 deletions(-)
```

diff --git a/include/net/ip_fib.h b/include/net/ip_fib.h

index be70b33..39f944a 100644

--- a/include/net/ip_fib.h

+++ b/include/net/ip_fib.h

```
@@ -193,14 +193,6 @@ static inline int fib_lookup(struct net *net, const struct flowi *flp,
    return -ENETUNREACH;
}
```

```
-static inline void fib_select_default(const struct flowi *flp,
```

```
-    struct fib_result *res)
```

```
-{
```

```
- struct fib_table *table = fib_get_table(&init_net, RT_TABLE_MAIN);
```

```
- if (FIB_RES_GW(*res) && FIB_RES_NH(*res).nh_scope == RT_SCOPE_LINK)
```

```
- table->tb_select_default(table, flp, res);
```

```
-}
```

```
-
```

```
#else /* CONFIG_IP_MULTIPLE_TABLES */
```

```
extern int __net_init fib4_rules_init(struct net *net);
```

```
extern void __net_exit fib4_rules_exit(struct net *net);
```

```
@@ -213,7 +205,6 @@ extern int fib_lookup(struct net *n, struct flowi *flp, struct fib_result *res);
```

```
extern struct fib_table *fib_new_table(struct net *net, u32 id);
```

```
extern struct fib_table *fib_get_table(struct net *net, u32 id);
```

```
-extern void fib_select_default(const struct flowi *flp, struct fib_result *res);
```

```
#endif /* CONFIG_IP_MULTIPLE_TABLES */
```

```
@@ -222,6 +213,7 @@ extern const struct nla_policy rtm_ipv4_policy[];
```

```
extern void ip_fib_init(void);
```

```
extern int fib_validate_source(__be32 src, __be32 dst, u8 tos, int oif,
```

```
    struct net_device *dev, __be32 *spec_dst, u32 *itag);
```

```
+extern void fib_select_default(const struct flowi *flp, struct fib_result *res);
```

```

/* Exported by fib_semantics.c */
extern int ip_fib_check_default(__be32 gw, struct net_device *dev);
diff --git a/net/ipv4/fib_frontend.c b/net/ipv4/fib_frontend.c
index 6761639..13bf01d 100644
--- a/net/ipv4/fib_frontend.c
+++ b/net/ipv4/fib_frontend.c
@@ -116,6 +116,20 @@ struct fib_table *fib_get_table(struct net *net, u32 id)
}
#endif /* CONFIG_IP_MULTIPLE_TABLES */

+void fib_select_default(const struct flowi *flp, struct fib_result *res)
+{
+ struct fib_table *tb;
+ int table = RT_TABLE_MAIN;
+ #ifdef CONFIG_IP_MULTIPLE_TABLES
+ if (res->r == NULL || res->r->action != FR_ACT_TO_TBL)
+ return;
+ table = res->r->table;
+ #endif
+ tb = fib_get_table(&init_net, table);
+ if (FIB_RES_GW(*res) && FIB_RES_NH(*res).nh_scope == RT_SCOPE_LINK)
+ tb->tb_select_default(tb, flp, res);
+}
+
static void fib_flush(struct net *net)
{
int flushed = 0;
diff --git a/net/ipv4/fib_rules.c b/net/ipv4/fib_rules.c
index 1effb4a..19274d0 100644
--- a/net/ipv4/fib_rules.c
+++ b/net/ipv4/fib_rules.c
@@ -102,16 +102,6 @@ errout:
}

-void fib_select_default(const struct flowi *flp, struct fib_result *res)
-{
- if (res->r && res->r->action == FR_ACT_TO_TBL &&
- FIB_RES_GW(*res) && FIB_RES_NH(*res).nh_scope == RT_SCOPE_LINK) {
- struct fib_table *tb;
- if ((tb = fib_get_table(&init_net, res->r->table)) != NULL)
- tb->tb_select_default(tb, flp, res);
- }
-}
-
static int fib4_rule_match(struct fib_rule *rule, struct flowi *fl, int flags)
{
struct fib4_rule *r = (struct fib4_rule *) rule;

```

--
1.5.3.rc5
