
Subject: [PATCH net-2.6.25 4/19] [NETNS] Add namespace to API for routing /proc entries creation.

Posted by [den](#) on Wed, 09 Jan 2008 17:59:19 GMT

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

This adds netns parameter to fib_proc_init/exit and replaces __init specifier with __net_init. After this, we will not yet have these proc files show info from the specific namespace - this will be done when these tables become namespaced.

Acked-by: Benjamin Thery <benjamin.thery@bull.net>

Acked-by: Daniel Lezcano <dlezcano@fr.ibm.com>

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

```
include/net/ip_fib.h | 4 +--
net/ipv4/af_inet.c | 4 +--
net/ipv4/fib_hash.c | 8 ++++----
net/ipv4/fib_trie.c | 21 ++++++++-----
4 files changed, 19 insertions(+), 18 deletions(-)
```

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

index d70b9b4..f74cbb2 100644

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

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

```
@@ -253,8 +253,8 @@ static inline void fib_res_put(struct fib_result *res)
}
```

```
#ifdef CONFIG_PROC_FS
-extern int fib_proc_init(void);
-extern void fib_proc_exit(void);
+extern int __net_init fib_proc_init(struct net *net);
+extern void __net_exit fib_proc_exit(struct net *net);
#endif
```

```
#endif /* _NET_FIB_H */
```

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

index 03633b7..ea9fd3d 100644

--- a/net/ipv4/af_inet.c

+++ b/net/ipv4/af_inet.c

```
@@ -1470,14 +1470,14 @@ static int __init ipv4_proc_init(void)
    goto out_tcp;
    if (udp4_proc_init())
        goto out_udp;
- if (fib_proc_init())
+ if (fib_proc_init(&init_net))
    goto out_fib;
    if (ip_misc_proc_init())
        goto out_misc;
```

```

out:
    return rc;
out_misc:
- fib_proc_exit();
+ fib_proc_exit(&init_net);
out_fib:
    udp4_proc_exit();
out_udp:
diff --git a/net/ipv4/fib_hash.c b/net/ipv4/fib_hash.c
index 481de47..f5476e2 100644
--- a/net/ipv4/fib_hash.c
+++ b/net/ipv4/fib_hash.c
@@ -1038,15 +1038,15 @@ static const struct file_operations fib_seq_fops = {
    .release = seq_release_private,
};

-int __init fib_proc_init(void)
+int __net_init fib_proc_init(struct net *net)
{
- if (!proc_net_fops_create(&init_net, "route", S_IRUGO, &fib_seq_fops))
+ if (!proc_net_fops_create(net, "route", S_IRUGO, &fib_seq_fops))
    return -ENOMEM;
    return 0;
}

-void __init fib_proc_exit(void)
+void __net_exit fib_proc_exit(struct net *net)
{
- proc_net_remove(&init_net, "route");
+ proc_net_remove(net, "route");
}
#endif /* CONFIG_PROC_FS */
diff --git a/net/ipv4/fib_trie.c b/net/ipv4/fib_trie.c
index 46c3b01..3c7f668 100644
--- a/net/ipv4/fib_trie.c
+++ b/net/ipv4/fib_trie.c
@@ -2505,32 +2505,32 @@ static const struct file_operations fib_route_fops = {
    .release = seq_release_private,
};

-int __init fib_proc_init(void)
+int __net_init fib_proc_init(struct net *net)
{
- if (!proc_net_fops_create(&init_net, "fib_trie", S_IRUGO, &fib_trie_fops))
+ if (!proc_net_fops_create(net, "fib_trie", S_IRUGO, &fib_trie_fops))
    goto out1;

- if (!proc_net_fops_create(&init_net, "fib_triestat", S_IRUGO, &fib_triestat_fops))

```

```

+ if (!proc_net_fops_create(net, "fib_triestat", S_IRUGO,
+   &fib_triestat_fops))
    goto out2;

- if (!proc_net_fops_create(&init_net, "route", S_IRUGO, &fib_route_fops))
+ if (!proc_net_fops_create(net, "route", S_IRUGO, &fib_route_fops))
    goto out3;

    return 0;

out3:
- proc_net_remove(&init_net, "fib_triestat");
+ proc_net_remove(net, "fib_triestat");
out2:
- proc_net_remove(&init_net, "fib_tribe");
+ proc_net_remove(net, "fib_tribe");
out1:
    return -ENOMEM;
}

-void __init fib_proc_exit(void)
+void __net_exit fib_proc_exit(struct net *net)
{
- proc_net_remove(&init_net, "fib_tribe");
- proc_net_remove(&init_net, "fib_triestat");
- proc_net_remove(&init_net, "route");
+ proc_net_remove(net, "fib_tribe");
+ proc_net_remove(net, "fib_triestat");
+ proc_net_remove(net, "route");
}

#endif /* CONFIG_PROC_FS */
--
1.5.3.rc5

```
