
Subject: [PATCH 2.6.25 1/3] ipv4: no need pass pointer to a default into fib_detect_death

Posted by [den](#) on Fri, 07 Dec 2007 16:08:37 GMT

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

ipv4: no need pass pointer to a default into fib_detect_death

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

Acked-by: Alexey Kuznetsov <kuznet@ms2.inr.ac.ru>

net/ipv4/fib_hash.c | 4 ++--

net/ipv4/fib_lookup.h | 2 +-

net/ipv4/fib_semantics.c | 6 +++---

net/ipv4/fib_trie.c | 4 ++--

4 files changed, 8 insertions(+), 8 deletions(-)

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

index 30ff657..76bb7fd 100644

--- a/net/ipv4/fib_hash.c

+++ b/net/ipv4/fib_hash.c

@@ -314,7 +314,7 @@ fn_hash_select_default(struct fib_table *tb, const struct flowi *flp, struct fib

if (next_fi != res->fi)

break;

} else if (!fib_detect_death(fi, order, &last_resort,

- &last_idx, &fn_hash_last_dflt)) {

+ &last_idx, fn_hash_last_dflt)) {

if (res->fi)

fib_info_put(res->fi);

res->fi = fi;

@@ -332,7 +332,7 @@ fn_hash_select_default(struct fib_table *tb, const struct flowi *flp, struct fib

goto out;

}

- if (!fib_detect_death(fi, order, &last_resort, &last_idx, &fn_hash_last_dflt)) {

+ if (!fib_detect_death(fi, order, &last_resort, &last_idx, fn_hash_last_dflt)) {

if (res->fi)

fib_info_put(res->fi);

res->fi = fi;

diff --git a/net/ipv4/fib_lookup.h b/net/ipv4/fib_lookup.h

index eef9eec..6c9dd42 100644

--- a/net/ipv4/fib_lookup.h

+++ b/net/ipv4/fib_lookup.h

@@ -36,6 +36,6 @@ extern struct fib_alias *fib_find_alias(struct list_head *fah,
u8 tos, u32 prio);

extern int fib_detect_death(struct fib_info *fi, int order,
struct fib_info **last_resort,

```

-     int *last_idx, int *dflt);
+     int *last_idx, int dflt);

#endif /* _FIB_LOOKUP_H */
diff --git a/net/ipv4/fib_semantics.c b/net/ipv4/fib_semantics.c
index ec9b0dd..bbd4a24 100644
--- a/net/ipv4/fib_semantics.c
+++ b/net/ipv4/fib_semantics.c
@@ -346,7 +346,7 @@ struct fib_alias *fib_find_alias(struct list_head *fah, u8 tos, u32 prio)
}

int fib_detect_death(struct fib_info *fi, int order,
-     struct fib_info **last_resort, int *last_idx, int *dflt)
+     struct fib_info **last_resort, int *last_idx, int dflt)
{
    struct neighbour *n;
    int state = NUD_NONE;
@@ -358,10 +358,10 @@ int fib_detect_death(struct fib_info *fi, int order,
}
if (state==NUD_REACHABLE)
    return 0;
- if ((state&NUD_VALID) && order != *dflt)
+ if ((state&NUD_VALID) && order != dflt)
    return 0;
if ((state&NUD_VALID) ||
-     (*last_idx<0 && order > *dflt)) {
+     (*last_idx<0 && order > dflt)) {
    *last_resort = fi;
    *last_idx = order;
}
diff --git a/net/ipv4/fib_trie.c b/net/ipv4/fib_trie.c
index 6385cca..914a0d2 100644
--- a/net/ipv4/fib_trie.c
+++ b/net/ipv4/fib_trie.c
@@ -1827,7 +1827,7 @@ fn_trie_select_default(struct fib_table *tb, const struct flowi *flp, struct
fib
    if (next_fi != res->fi)
        break;
    } else if (!fib_detect_death(fi, order, &last_resort,
-     &last_idx, &trie_last_dflt)) {
+     &last_idx, trie_last_dflt)) {
    if (res->fi)
        fib_info_put(res->fi);
    res->fi = fi;
@@ -1843,7 +1843,7 @@ fn_trie_select_default(struct fib_table *tb, const struct flowi *flp, struct
fib
    goto out;
}

```

```
- if (!fib_detect_death(fi, order, &last_resort, &last_idx, &trie_last_dflt)) {  
+ if (!fib_detect_death(fi, order, &last_resort, &last_idx, trie_last_dflt)) {  
    if (res->fi)  
        fib_info_put(res->fi);  
    res->fi = fi;  
--  
1.5.3.rc5
```
