

---

Subject: [PATCH] lockd: fix races in per-net NSM client handling  
Posted by [Stanislav Kinsbursky](#) on Wed, 24 Oct 2012 10:18:10 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

This patch fixes two problems:

- 1) Removes races on NSM creation.
- 2) Fixes silly misprint on NSM client destruction (usage counter was checked for non-zero value instead of zero).

Signed-off-by: Stanislav Kinsbursky <[skinsbursky@parallels.com](mailto:skinsbursky@parallels.com)>

---

```
fs/lockd/mon.c | 35 ++++++-----  
1 files changed, 23 insertions(+), 12 deletions(-)
```

```
diff --git a/fs/lockd/mon.c b/fs/lockd/mon.c
```

```
index e4fb3ba..e3e59f6 100644
```

```
--- a/fs/lockd/mon.c
```

```
+++ b/fs/lockd/mon.c
```

```
@@ -85,30 +85,41 @@ static struct rpc_clnt *nsm_create(struct net *net)  
    return rpc_create(&args);  
}
```

```
-static struct rpc_clnt *nsm_client_get(struct net *net)  
+static struct rpc_clnt *nsm_get_client(struct net *net)
```

```
{  
- static DEFINE_MUTEX(nsm_create_mutex);  
- struct rpc_clnt *clnt;  
+ struct rpc_clnt *clnt = NULL;  
    struct lockd_net *ln = net_generic(net, lockd_net_id);
```

```
    spin_lock(&ln->nsm_clnt_lock);  
    if (ln->nsm_users) {  
        ln->nsm_users++;  
        clnt = ln->nsm_clnt;  
-    spin_unlock(&ln->nsm_clnt_lock);  
-    goto out;  
    }
```

```
    spin_unlock(&ln->nsm_clnt_lock);  
+    return clnt;
```

```
+}
```

```
+
```

```
+static struct rpc_clnt *nsm_client_get(struct net *net)
```

```
+{  
+ static DEFINE_MUTEX(nsm_create_mutex);  
+ struct rpc_clnt *clnt;  
+ struct lockd_net *ln = net_generic(net, lockd_net_id);  
+  
+ clnt = nsm_get_client(net);
```

```

+ if (clnt)
+ return clnt;

    mutex_lock(&nsm_create_mutex);
- clnt = nsm_create(net);
- if (!IS_ERR(clnt)) {
- ln->nsm_clnt = clnt;
- smp_wmb();
- ln->nsm_users = 1;
+ clnt = nsm_get_client(net);
+ if (clnt == NULL) {
+ clnt = nsm_create(net);
+ if (!IS_ERR(clnt)) {
+ ln->nsm_clnt = clnt;
+ smp_wmb();
+ ln->nsm_users = 1;
+ }
}
    mutex_unlock(&nsm_create_mutex);
-out:
    return clnt;
}

```

@@ -120,7 +131,7 @@ static void nsm\_client\_put(struct net \*net)

```

    spin_lock(&ln->nsm_clnt_lock);
    if (ln->nsm_users) {
- if (--ln->nsm_users)
+ if (--ln->nsm_users == 0)
        ln->nsm_clnt = NULL;
        shutdown = !ln->nsm_users;
    }

```