
Subject: [PATCH v2 2/4] NFS: release per-net clients lock before calling PipeFS dentries creation

Posted by [Stanislav Kinsbursky](#) on Mon, 27 Feb 2012 15:51:04 GMT

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

Lockdep is sad otherwise, because inode mutex is taken on PipeFS dentry creation, which can be called on mount notification, where this per-net client lock is taken on clients list walk.

Signed-off-by: Stanislav Kinsbursky <skinsbursky@parallels.com>

```
fs/nfs/client.c | 2 +-
fs/nfs/idmap.c | 8 ++++++--
2 files changed, 7 insertions(+), 3 deletions(-)
```

```
diff --git a/fs/nfs/client.c b/fs/nfs/client.c
```

```
index 8563585..6aeb6b3 100644
```

```
--- a/fs/nfs/client.c
```

```
+++ b/fs/nfs/client.c
```

```
@@ -538,7 +538,7 @@ nfs_get_client(const struct nfs_client_initdata *cl_init,
```

```
/* install a new client and return with it unready */
```

```
install_client:
```

```
    clp = new;
```

```
- list_add(&clp->cl_share_link, &nn->nfs_client_list);
```

```
+ list_add_tail(&clp->cl_share_link, &nn->nfs_client_list);
```

```
    spin_unlock(&nn->nfs_client_lock);
```

```
    error = cl_init->rpc_ops->init_client(clp, timeparms, ip_addr,
```

```
diff --git a/fs/nfs/idmap.c b/fs/nfs/idmap.c
```

```
index b5c6d8e..8a9e7a4 100644
```

```
--- a/fs/nfs/idmap.c
```

```
+++ b/fs/nfs/idmap.c
```

```
@@ -558,16 +558,20 @@ static int rpc_pipefs_event(struct notifier_block *nb, unsigned long event,
```

```
{
```

```
    struct super_block *sb = ptr;
```

```
    struct nfs_net *nn = net_generic(sb->s_fs_info, nfs_net_id);
```

```
- struct nfs_client *clp;
```

```
+ struct nfs_client *clp, *tmp;
```

```
    int error = 0;
```

```
    spin_lock(&nn->nfs_client_lock);
```

```
- list_for_each_entry(clp, &nn->nfs_client_list, cl_share_link) {
```

```
+ list_for_each_entry_safe(clp, tmp, &nn->nfs_client_list, cl_share_link) {
```

```
    if (clp->rpc_ops != &nfs_v4_clientops)
```

```
        continue;
```

```
+ atomic_inc(&clp->cl_count);
```

```
+ spin_unlock(&nn->nfs_client_lock);
  error = __rpc_pipefs_event(clp, event, sb);
+ nfs_put_client(clp);
  if (error)
    break;
+ spin_lock(&nn->nfs_client_lock);
  }
  spin_unlock(&nn->nfs_client_lock);
  return error;
```
