
Subject: [PATCH RFC 09/13] NFSd: make nfsd4_manager allocated per network namespace context.

Posted by [Stanislav Kinsbursky](#) on Sat, 05 May 2012 17:22:37 GMT

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

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

```
fs/nfsd/netns.h    | 2 ++
fs/nfsd/nfs4state.c | 32 ++++++-----
2 files changed, 21 insertions(+), 13 deletions(-)
```

```
diff --git a/fs/nfsd/netns.h b/fs/nfsd/netns.h
```

```
index 3936563..e99767d 100644
```

```
--- a/fs/nfsd/netns.h
```

```
+++ b/fs/nfsd/netns.h
```

```
@@ -34,6 +34,8 @@ struct nfsd_net {
```

```
    struct cache_detail *idtoname_cache;
```

```
    struct cache_detail *nametoid_cache;
```

```
+
```

```
+ struct lock_manager nfsd4_manager;
```

```
};
```

```
extern int nfsd_net_id;
```

```
diff --git a/fs/nfsd/nfs4state.c b/fs/nfsd/nfs4state.c
```

```
index c33f052..c6c1be9 100644
```

```
--- a/fs/nfsd/nfs4state.c
```

```
+++ b/fs/nfsd/nfs4state.c
```

```
@@ -44,6 +44,8 @@
```

```
#include "vfs.h"
```

```
#include "current_stateid.h"
```

```
+#include "netns.h"
```

```
+
```

```
#define NFSDDBG_FACILITY      NFSDDBG_PROC
```

```
/* Globals */
```

```
@@ -3184,22 +3186,21 @@ out:
```

```
    return status;
```

```
}
```

```
-static struct lock_manager nfsd4_manager = {
```

```
-};
```

```
-
```

```
static bool grace_ended;
```

```
static void
```

```
-nfsd4_end_grace(void)
```

```

+ nfsd4_end_grace(struct net *net)
{
+ struct nfsd_net *nn = net_generic(net, nfsd_net_id);
+
  /* do nothing if grace period already ended */
  if (grace_ended)
    return;

  dprintk("NFSD: end of grace period\n");
  grace_ended = true;
- nfsd4_record_grace_done(&init_net, boot_time);
- locks_end_grace(&nfsd4_manager);
+ nfsd4_record_grace_done(net, boot_time);
+ locks_end_grace(&nn->nfsd4_manager);
  /*
   * Now that every NFSv4 client has had the chance to recover and
   * to see the (possibly new, possibly shorter) lease time, we
@@ -3222,7 +3223,7 @@ nfs4_laundromat(void)
  nfs4_lock_state();

  dprintk("NFSD: laundromat service - starting\n");
- nfsd4_end_grace();
+ nfsd4_end_grace(&init_net);
  INIT_LIST_HEAD(&reaplist);
  spin_lock(&client_lock);
  list_for_each_safe(pos, next, &client_lru) {
@@ -4743,6 +4744,8 @@ set_max_delegations(void)
int
nfs4_state_start(void)
{
+ struct net *net = &init_net;
+ struct nfsd_net *nn = net_generic(net, nfsd_net_id);
  int ret;

  /*
@@ -4752,10 +4755,10 @@ nfs4_state_start(void)
   * to that instead and then do most of the rest of this on a per-net
   * basis.
   */
- get_net(&init_net);
- nfsd4_client_tracking_init(&init_net);
+ get_net(net);
+ nfsd4_client_tracking_init(net);
  boot_time = get_seconds();
- locks_start_grace(&nfsd4_manager);
+ locks_start_grace(&nn->nfsd4_manager);
  grace_ended = false;
  printk(KERN_INFO "NFSD: starting %ld-second grace period\n",

```

```

        nfsd4_grace);
@@ -4778,8 +4781,8 @@ nfs4_state_start(void)
out_free_laundry:
    destroy_workqueue(laundry_wq);
out_recovery:
- nfsd4_client_tracking_exit(&init_net);
- put_net(&init_net);
+ nfsd4_client_tracking_exit(net);
+ put_net(net);
    return ret;
}

@@ -4820,9 +4823,12 @@ __nfs4_state_shutdown(void)
void
nfs4_state_shutdown(void)
{
+ struct net *net = &init_net;
+ struct nfsd_net *nn = net_generic(net, nfsd_net_id);
+
    cancel_delayed_work_sync(&laundromat_work);
    destroy_workqueue(laundry_wq);
- locks_end_grace(&nfsd4_manager);
+ locks_end_grace(&nn->nfsd4_manager);
    nfs4_lock_state();
    __nfs4_state_shutdown();
    nfs4_unlock_state();

```
