
Subject: [PATCH RFC 07/13] LockD: manage grace period per network namespace
Posted by [Stanislav Kinsbursky](#) on Sat, 05 May 2012 17:22:22 GMT
[View Forum Message](#) <> [Reply to Message](#)

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

```
fs/lockd/netns.h | 2 ++
fs/lockd/svc.c   | 17 ++++++++-----
2 files changed, 13 insertions(+), 6 deletions(-)
```

```
diff --git a/fs/lockd/netns.h b/fs/lockd/netns.h
index 44c8f0b..94653ae 100644
```

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

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

```
@@ -7,6 +7,8 @@ struct lockd_net {
    unsigned int nlmsvc_users;
    unsigned long next_gc;
    unsigned long nrhosts;
```

```
+
```

```
+ struct delayed_work grace_period_end;
};
```

```
extern int lockd_net_id;
```

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

```
index 80938fd..70c4177 100644
```

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

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

```
@@ -95,21 +95,22 @@ static void grace_ender(struct work_struct *not_used)
    locks_end_grace(&lockd_manager);
}
```

```
-static DECLARE_DELAYED_WORK(grace_period_end, grace_ender);
```

```
-
```

```
static void set_grace_period(void)
```

```
{
```

```
    unsigned long grace_period = get_lockd_grace_period();
```

```
+ struct lockd_net *ln = net_generic(&init_net, lockd_net_id);
```

```
    locks_start_grace(&lockd_manager);
```

```
- cancel_delayed_work_sync(&grace_period_end);
```

```
- schedule_delayed_work(&grace_period_end, grace_period);
```

```
+ cancel_delayed_work_sync(&ln->grace_period_end);
```

```
+ schedule_delayed_work(&ln->grace_period_end, grace_period);
```

```
}
```

```
static void restart_grace(void)
```

```
{
```

```
    if (nlmsvc_ops) {
```

```

- cancel_delayed_work_sync(&grace_period_end);
+ struct lockd_net *ln = net_generic(&init_net, lockd_net_id);
+
+ cancel_delayed_work_sync(&ln->grace_period_end);
  locks_end_grace(&lockd_manager);
  nlmsvc_invalidate_all();
  set_grace_period();
@@ -124,6 +125,7 @@ lockd(void *vrqstp)
{
  int err = 0, preverr = 0;
  struct svc_rqst *rqstp = vrqstp;
+ struct lockd_net *ln = net_generic(&init_net, lockd_net_id);

  /* try_to_freeze() is called from svc_rcv() */
  set_freezable();
@@ -184,7 +186,7 @@ lockd(void *vrqstp)
  svc_process(rqstp);
}
flush_signals(current);
- cancel_delayed_work_sync(&grace_period_end);
+ cancel_delayed_work_sync(&ln->grace_period_end);
  locks_end_grace(&lockd_manager);
  if (nlmsvc_ops)
    nlmsvc_invalidate_all();
@@ -589,6 +591,9 @@ module_param(nlm_max_connections, uint, 0644);

static int lockd_init_net(struct net *net)
{
+ struct lockd_net *ln = net_generic(net, lockd_net_id);
+
+ INIT_DELAYED_WORK(&ln->grace_period_end, grace_ender);
  return 0;
}

```
