
Subject: [PATCH] nfs lockd reclaimer: Convert to kthread API

Posted by [ebiederm](#) on Thu, 19 Apr 2007 07:58:59 GMT

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

From: Eric W. Biederman <ebiederm@xmission.com>

Start the reclaimer thread using kthread_run instead of a combination of kernel_thread and daemonize.

The small amount of signal handling code is also removed as it makes no sense and is a maintenance problem to handle signals in kernel threads.

Cc: Neil Brown <neilb@suse.de>

Cc: Trond Myklebust <trond.myklebust@fys.uio.no>

Signed-off-by: Eric W. Biederman <ebiederm@xmission.com>

fs/lockd/clntlock.c | 8 ++-----

1 files changed, 2 insertions(+), 6 deletions(-)

diff --git a/fs/lockd/clntlock.c b/fs/lockd/clntlock.c

index f4d45d4..83591f6 100644

--- a/fs/lockd/clntlock.c

+++ b/fs/lockd/clntlock.c

@@ -9,6 +9,7 @@

#include <linux/module.h>

#include <linux/types.h>

#include <linux/time.h>

+#include <linux/kthread.h>

#include <linux/nfs_fs.h>

#include <linux/sunrpc/clnt.h>

#include <linux/sunrpc/svc.h>

@@ -153,7 +154,7 @@ nlmclnt_recovery(struct nlm_host *host)

if (!host->h_reclaiming++) {

nlm_get_host(host);

__module_get(THIS_MODULE);

- if (kernel_thread(reclaimer, host, CLONE_KERNEL) < 0)

+ if (IS_ERR(kthread_run(reclaimer, host, "%s-reclaim", host->h_name)))

module_put(THIS_MODULE);

}

}

@@ -166,9 +167,6 @@ reclaimer(void *ptr)

struct file_lock *fl, *next;

u32 nsmstate;

- daemonize("%s-reclaim", host->h_name);

- allow_signal(SIGKILL);

-

down_write(&host->h_rwsem);

```
/* This one ensures that our parent doesn't terminate while the
@@ -193,8 +191,6 @@ restart:
    list_del_init(&fl->fl_u.nfs_fl.list);

/* Why are we leaking memory here? --okir */
- if (signalled())
- continue;
if (nlmcnt_reclaim(host, fl) != 0)
    continue;
list_add_tail(&fl->fl_u.nfs_fl.list, &host->h_granted);
--
1.5.0.g53756
```

Containers mailing list
Containers@lists.linux-foundation.org
<https://lists.linux-foundation.org/mailman/listinfo/containers>
