

> On Tue, Jul 03, 2012 at 04:46:41PM +0400, Stanislav Kinsbursky wrote:
>> v3:
>> 1) Rebased for 3.6 kernel.
>>
>> v2:
>> 1) Set global nfsd_serv pointer to NULL only if no running threads left.
>>
>> Otherwise we will get NULL pointer dereference on last thread exit in
>> nfsd_last_thread().
>> This patch also introduces nfsd_destroy() helper for per-net NFSd shutdown.
>
> That last step should really be done in a separate patch, then the
> bugfix done after that.
>
> But I'm confused: I'm not seeing the NULL dereference.
> nfsd_last_thread doesn't use nfsd_serv as far as I can tell.
>

Sorry for the bad comment, Bruce. It takes time even for me to reconstruct the problem in my mind.
The reason for this patch was an error path in nfs_svc().

```
int
nfsd_svc(unsigned short port, int nrsvcs)
{
<--- snip --->

out_destroy:
    if (nfsd_serv->sv_nrthreads == 1)
        svc_shutdown_net(nfsd_serv, net); <=== nfsd_serv = NULL
    svc_destroy(nfsd_serv);    <=== NULL ptr
out:
    mutex_unlock(&nfsd_mutex);
    return error;
}
```

I'll prepare patch series with proper comments. Thanks.

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

```

>> ---
>> fs/nfsd/nfsctl.c | 8 ++-----
>> fs/nfsd/nfsd.h | 11 ++++++++
>> fs/nfsd/nfssvc.c | 24 ++++++-----
>> 3 files changed, 21 insertions(+), 22 deletions(-)
>>
>> diff --git a/fs/nfsd/nfsctl.c b/fs/nfsd/nfsctl.c
>> index c55298e..fa49cff 100644
>> --- a/fs/nfsd/nfsctl.c
>> +++ b/fs/nfsd/nfsctl.c
>> @@ -673,9 +673,7 @@ static ssize_t __write_ports_addfd(char *buf)
>>
>> err = svc_addsock(nfsd_serv, fd, buf, SIMPLE_TRANSACTION_LIMIT);
>> if (err < 0) {
>> - if (nfsd_serv->sv_nrthreads == 1)
>> - svc_shutdown_net(nfsd_serv, net);
>> - svc_destroy(nfsd_serv);
>> + nfsd_destroy(net);
>> return err;
>> }
>>
>> @@ -744,9 +742,7 @@ out_close:
>> svc_xprt_put(xprt);
>> }
>> out_err:
>> - if (nfsd_serv->sv_nrthreads == 1)
>> - svc_shutdown_net(nfsd_serv, net);
>> - svc_destroy(nfsd_serv);
>> + nfsd_destroy(net);
>> return err;
>> }
>>
>> diff --git a/fs/nfsd/nfsd.h b/fs/nfsd/nfsd.h
>> index 1671429..1336a65 100644
>> --- a/fs/nfsd/nfsd.h
>> +++ b/fs/nfsd/nfsd.h
>> @@ -73,6 +73,17 @@ int nfsd_nrpools(void);
>> int nfsd_get_nrthreads(int n, int *);
>> int nfsd_set_nrthreads(int n, int *);
>>
>> +static inline void nfsd_destroy(struct net *net)
>> +{
>> + int destroy = (nfsd_serv->sv_nrthreads == 1);
>> +
>> + if (destroy)
>> + svc_shutdown_net(nfsd_serv, net);
>> + svc_destroy(nfsd_serv);
>> + if (destroy)

```

```

>> + nfsd_serv = NULL;
>> +}
>> +
>> #if defined(CONFIG_NFSD_V2_ACL) || defined(CONFIG_NFSD_V3_ACL)
>> #ifdef CONFIG_NFSD_V2_ACL
>> extern struct svc_version nfsd_acl_version2;
>> diff --git a/fs/nfsd/nfssvc.c b/fs/nfsd/nfssvc.c
>> index ee709fc..240473c 100644
>> --- a/fs/nfsd/nfssvc.c
>> +++ b/fs/nfsd/nfssvc.c
>> @@ -254,8 +254,6 @@ static void nfsd_shutdown(void)
>>
>> static void nfsd_last_thread(struct svc_serv *serv, struct net *net)
>> {
>> - /* When last nfsd thread exits we need to do some clean-up */
>> - nfsd_serv = NULL;
>> - nfsd_shutdown();
>>
>> svc_rpcb_cleanup(serv, net);
>> @@ -332,6 +330,7 @@ static int nfsd_get_default_max_blksize(void)
>> int nfsd_create_serv(void)
>> {
>> int error;
>> + struct net *net = current->nsproxy->net_ns;
>>
>> WARN_ON(!mutex_is_locked(&nfsd_mutex));
>> if (nfsd_serv) {
>> @@ -346,7 +345,7 @@ int nfsd_create_serv(void)
>> if (nfsd_serv == NULL)
>> return -ENOMEM;
>>
>> - error = svc_bind(nfsd_serv, current->nsproxy->net_ns);
>> + error = svc_bind(nfsd_serv, net);
>> if (error < 0) {
>> svc_destroy(nfsd_serv);
>> return error;
>> @@ -427,11 +426,7 @@ int nfsd_set_nrthreads(int n, int *nthreads)
>> if (err)
>> break;
>> }
>> -
>> - if (nfsd_serv->sv_nrthreads == 1)
>> - svc_shutdown_net(nfsd_serv, net);
>> - svc_destroy(nfsd_serv);
>> -
>> + nfsd_destroy(net);
>> return err;
>> }

```

```

>>
>> @@ -478,9 +473,7 @@ out_shutdown:
>> if (error < 0 && !nfsd_up_before)
>>   nfsd_shutdown();
>> out_destroy:
>> - if (nfsd_serv->sv_nrthreads == 1)
>> - svc_shutdown_net(nfsd_serv, net);
>> - svc_destroy(nfsd_serv); /* Release server */
>> + nfsd_destroy(net); /* Release server */
>> out:
>>   mutex_unlock(&nfsd_mutex);
>>   return error;
>> @@ -563,12 +556,13 @@ nfsd(void *vrqstp)
>>   nfsdstats.th_cnt--;
>>
>> out:
>> - if (rqstp->rq_server->sv_nrthreads == 1)
>> - svc_shutdown_net(rqstp->rq_server, &init_net);
>> + rqstp->rq_server = NULL;
>>
>>   /* Release the thread */
>>   svc_exit_thread(rqstp);
>>
>> + nfsd_destroy(&init_net);
>> +
>>   /* Release module */
>>   mutex_unlock(&nfsd_mutex);
>>   module_put_and_exit(0);
>> @@ -682,9 +676,7 @@ int nfsd_pool_stats_release(struct inode *inode, struct file *file)
>>
>>   mutex_lock(&nfsd_mutex);
>>   /* this function really, really should have been called svc_put() */
>> - if (nfsd_serv->sv_nrthreads == 1)
>> - svc_shutdown_net(nfsd_serv, net);
>> - svc_destroy(nfsd_serv);
>> + nfsd_destroy(net);
>>   mutex_unlock(&nfsd_mutex);
>>   return ret;
>> }
>>

```

--

Best regards,
Stanislav Kinsbursky