
Subject: Re: [PATCH v6 1/8] SUNRPC: introduce helpers for reference counted
rpcbind clients

Posted by [Trond Myklebust](#) on Tue, 25 Oct 2011 12:45:50 GMT

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

On Tue, 2011-10-25 at 16:41 +0400, Stanislav Kinsbursky wrote:

> > On Tue, 2011-10-25 at 14:16 +0300, Stanislav Kinsbursky wrote:

> >> v6:

> >> 1) added write memory barrier to rpcb_set_local to make sure, that rpcbind

> >> clients become valid before rpcb_users assignment

> >> 2) explicitly set rpcb_users to 1 instead of incrementing it (looks clearer from

> >> my pow).

> >>

> >> v5: fixed races with rpcb_users in rpcb_get_local()

> >>

> >> This helpers will be used for dynamical creation and destruction of rpcbind

> >> clients.

> >> Variable rpcb_users is actually a counter of lauched RPC services. If rpcbind

> >> clients has been created already, then we just increase rpcb_users.

> >>

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

> >>

> >> ---

> >> net/sunrpc/rpcb_clnt.c | 54 ++++++
> >> 1 files changed, 54 insertions(+), 0 deletions(-)

> >>

> >> diff --git a/net/sunrpc/rpcb_clnt.c b/net/sunrpc/rpcb_clnt.c

> >> index e45d2fb..9fcdb42 100644

> >> --- a/net/sunrpc/rpcb_clnt.c

> >> +++ b/net/sunrpc/rpcb_clnt.c

> >> @@ -114,6 +114,9 @@ static struct rpc_program rpcb_program;

> >> static struct rpc_clnt * rpcb_local_clnt;

> >> static struct rpc_clnt * rpcb_local_clnt4;

> >>

> >> +DEFINE_SPINLOCK(rpcb_clnt_lock);

> >> +unsigned int rpcb_users;

> >> +

> >> struct rpcbind_args {

> >> struct rpc_xprt * r_xprt;

> >>

> >> @@ -161,6 +164,57 @@ static void rpcb_map_release(void *data)

> >> kfree(map);

> >> }

> >>

> >> +static int rpcb_get_local(void)

> >> +{

> >> + int cnt;

```

>>> +
>>> + spin_lock(&rpcb_clnt_lock);
>>> + if (rpcb_users)
>>> +   rpcb_users++;
>>> +   cnt = rpcb_users;
>>> + spin_unlock(&rpcb_clnt_lock);
>>> +
>>> + return cnt;
>>> +}
>>> +
>>> +void rpcb_put_local(void)
>>> +{
>>> + struct rpc_clnt *clnt = rpcb_local_clnt;
>>> + struct rpc_clnt *clnt4 = rpcb_local_clnt4;
>>> + int shutdown;
>>> +
>>> + spin_lock(&rpcb_clnt_lock);
>>> + if (--rpcb_users == 0) {
>>> +   rpcb_local_clnt = NULL;
>>> +   rpcb_local_clnt4 = NULL;
>>> + }
>>> + shutdown = !rpcb_users;
>>> + spin_unlock(&rpcb_clnt_lock);
>>> +
>>> + if (shutdown) {
>>> +   /*
>>> +    * cleanup_rpcb_clnt - remove xprtsock's sysctls, unregister
>>> +    */
>>> +   if (clnt4)
>>> +     rpc_shutdown_client(clnt4);
>>> +   if (clnt)
>>> +     rpc_shutdown_client(clnt);
>>> + }
>>> + return;
>>
>> I'm removing this before applying...
>>
> Sorry, but I don't understand what exactly you are removing, and why?

```

The empty 'return' at the end of a void function: it is 100% redundant...

--

Trond Myklebust
Linux NFS client maintainer

NetApp
Trond.Myklebust@netapp.com

