

---

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

Posted by [Trond Myklebust](#) on Tue, 25 Oct 2011 11:16:09 GMT

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

---

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...

```

> +}
> +

```

--

Trond Myklebust  
Linux NFS client maintainer

NetApp  
Trond.Myklebust@netapp.com  
www.netapp.com

---