Subject: Re: [PATCH 2/3] SUNRPC: check rpcbind clients usage counter before decrement
Posted by Stanislav Kinsbursky on Fri, 27 Apr 2012 14:08:13 GMT
View Forum Message <> Reply to Message

On 27.04.2012 17:55, J. Bruce Fields wrote:
> On Wed, Apr 25, 2012 at 05:37:49PM +0400, Stanislav Kinsbursky wrote:
>> Registering service with svc_bind() can fail. In this case service will be
>> destroyed and during destruction it will try to unregister itself from rpcbind.
>> In this case unregister have to be skipped.
>
> Isn't this a preexisting bug, in which case perhaps Trond should at it
> to his list of bugs to submit now?
>

Not it's not.

Previously, in case of bind operations failure, rpcb_put_local() wasn't called,
but service data was destroyed instead:

```
- if (svc_uses_rpcbind(serv)) {
-  if (svc_rpcb_setup(serv, current->nsproxy->net_ns) < 0) {
-   kfree(serv->sv_pools);
-   kfree(serv);
-   return NULL;
-  }
-  if (!serv->sv_shutdown)
-   serv->sv_shutdown = svc_rpcb_cleanup;
- }
```

Now (with svc_bind() introduction) service can be created, but svc_bind() could
fail. And thus svc_destroy() have to be called. Which will call rpcb_put_local().

And the problem here that svc_bind() can fail before incrementing of rpcb usage
counter.

```
> --b.
>
>>
>> Signed-off-by: Stanislav Kinsbursky<skinsbursky@parallels.com>
>>
>> ---
>>   net/sunrpc/rpcb_clnt.c |   12 +++++++-----
>>   1 files changed, 7 insertions(+), 5 deletions(-)
>>
>> diff --git a/net/sunrpc/rpcb_clnt.c b/net/sunrpc/rpcb_clnt.c
>> index 78ac39f..4c38b33 100644
>> --- a/net/sunrpc/rpcb_clnt.c
>> +++ b/net/sunrpc/rpcb_clnt.c
>> @@ -180,14 +180,16 @@ void rpcb_put_local(struct net *net)
>>    struct sunrpc_net *sn = net_generic(net, sunrpc_net_id);
>>    struct rpc_clnt *clnt = sn->rpcb_local_clnt;
>>    struct rpc_clnt *clnt4 = sn->rpcb_local_clnt4;
>> - int shutdown;
>> + int shutdown = 0;
>>
>>    spin_lock(&sn->rpcb_clnt_lock);
>> - if (--sn->rpcb_users == 0) {
>> -  sn->rpcb_local_clnt = NULL;
>> -  sn->rpcb_local_clnt4 = NULL;
>> + if (sn->rpcb_users) {
>> +  if (--sn->rpcb_users == 0) {
>> +   sn->rpcb_local_clnt = NULL;
>> +   sn->rpcb_local_clnt4 = NULL;
>> +  }
>> +  shutdown = !sn->rpcb_users;
>>    }
>> - shutdown = !sn->rpcb_users;
>>    spin_unlock(&sn->rpcb_clnt_lock);
>>
>>    if (shutdown) {
>>
```

--
Best regards,
Stanislav Kinsbursky