Subject: Re: [PATCH v2] SUNRPC: check current nsproxy before set of node name on client creation
Posted by Myklebust, Trond on Mon, 10 Sep 2012 15:27:20 GMT
View Forum Message <> Reply to Message

On Mon, 2012-09-10 at 12:43 +0400, Stanislav Kinsbursky wrote:

> > On Sat, 2012-09-08 at 08:59 +0300, Stanislav Kinsbursky wrote:

> >>> On Mon, 2012-08-13 at 08:10 -0400, Jeff Layton wrote:
> >>>> On Mon, 13 Aug 2012 15:37:31 +0400
> >>>> Stanislav Kinsbursky <skinsbursky@parallels.com> wrote:
> >>>>
> >>>> v2:
> >>>> 1) rpc_clnt_set_nodename() prototype updated.
> >>>> 2) fixed errors in comment.
> >>>>
> >>>> When child reaper exits, it can destroy mount namespace it belongs to, and if
> >>>> there are NFS mounts inside, then it will try to umount them. But in this
> >>>> point current->nsproxy is set to NULL and all namespaces will be destroyed one
> >>>> by one. I.e. we can't dereference current->nsproxy to obtain uts namespace.
> >>>>
> >>>> Signed-off-by: Stanislav Kinsbursky <skinsbursky@parallels.com>
> >>>> ---
> >>>> net/sunrpc/clnt.c | 16 +++++++++++++---
> >>>> 1 files changed, 13 insertions(+), 3 deletions(-)
> >>>>
> >>>> diff --git a/net/sunrpc/clnt.c b/net/sunrpc/clnt.c
> >>>> index 9a9676e..8fbcbc8 100644
> >>>> --- a/net/sunrpc/clnt.c
> >>>> +++ b/net/sunrpc/clnt.c
> >>>> @@ -277,8 +277,18 @@ void rpc_clients_notifier_unregister(void)
> >>>> 	return rpc_pipefs_notifier_unregister(&rpc_clients_block);
> >>>> }
> >>>>
> >>>> -static void rpc_clnt_set_nodename(struct rpc_clnt *clnt, const char *nodename)
> >>>> +static void rpc_clnt_set_nodename(struct rpc_clnt *clnt)
> >>>> {
> >>>> +	const char *nodename;
> >>>> +
> >>>> +	/*
> >>>> +	 * We have to protect against dying child reaper, which has released
> >>>> +	 * its nsproxy already and is trying to destroy mount namespace.
> >>>> +	 */
> >>>> +	if (current->nsproxy == NULL)
> >>>> +		return;
> >>>> +
> >>>> +	nodename = utsname()->nodename;

> >>>>>   clnt->cl_nodelen = strlen(nodename);
> >>>>>   if (clnt->cl_nodelen > UNX_MAXNODENAME)
> >>>>>   clnt->cl_nodelen = UNX_MAXNODENAME;
> >>>>> @@ -365,7 +375,7 @@ static struct rpc_clnt * rpc_new_client(const struct
rpc_create_args *args, stru
> >>>>>   }
> >>>>>
> >>>>>   /* save the nodename */
> >>>>> - rpc_clnt_set_nodename(clnt, utsname()->nodename);
> >>>>> + rpc_clnt_set_nodename(clnt);
> >>>>>   rpc_register_client(clnt);
> >>>>>   return clnt;
> >>>>>
> >>>>> @@ -524,7 +534,7 @@ rpc_clone_client(struct rpc_clnt *clnt)
> >>>>>   err = rpc_setup_pipedir(new, clnt->cl_program->pipe_dir_name);
> >>>>>   if (err != 0)
> >>>>>   goto out_no_path;
> >>>>> - rpc_clnt_set_nodename(new, utsname()->nodename);
> >>>>> + rpc_clnt_set_nodename(new);
> >>>>>   if (new->cl_auth)
> >>>>>   atomic_inc(&new->cl_auth->au_count);
> >>>>>   atomic_inc(&clnt->cl_count);
> >>>>>
> >>>>> --
> >>>>> To unsubscribe from this list: send the line "unsubscribe linux-nfs" in
> >>>>> the body of a message to majordomo@vger.kernel.org
> >>>>> More majordomo info at http://vger.kernel.org/majordomo-info.html
> >>>> Acked-by: Jeff Layton <jlayton@redhat.com>
> >>> OK, colour me confused (again).
> >>
> >> What color?
> >>
> >>> Why should a umount trigger an
> >>> rpc_create() or rpc_clone_client()?
> >>
> >> It calls nsm_create().
> >> Here is the trace (https://bugzilla.redhat.com/show_bug.cgi?id=830862,
> >> comment 68):
> >
> > Right, but if we're using NFSv3 lock monitoring, we know in advance that
> > we're going to need an nsm call to localhost. Why can't we just cache
> > the one that we used to start lock monitoring in the first place?
> >
>
> Do you suggest to cache the call or the client for the call?

Hi Stanislav,

Sorry, I agree that the above was unclear. My intention was to suggest that we should cache a reference to the rpc client that we used to connect to rpc.statd when initiating lock monitoring.

Basically, I'm suggesting that we should do something similar to the rpcbind rpc_client caching scheme in net/sunrpc/rpcb_clnt.c.

Cheers
  Trond

--
Trond Myklebust
Linux NFS client maintainer

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