Subject: Re: [PATCH v2] SUNRPC: check current nsproxy before set of node name on client creation
Posted by Stanislav Kinsbursky on Sat, 08 Sep 2012 05:59:53 GMT
View Forum Message <> Reply to Message

> 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):

CR2: 0000000000000008
Process mysqld

Call Trace:
? __schedule+0x3c7
nsm_create+0x8b
nsm_mon_unmon+0x64
nlm_destroy_host_locked+0x6b
nlmclnt_release_host+0x88
nlmclnt_done+0x1a
nfs_destroy_server+0x24
nfs_free_server+0xce

nfs_kill_super+0x34
deactivate_locked_super+0x57
deactivate_super+0x4e
mntput_no_expire+0xcc
mntput+0x26
release_mounts+0x77
put_mnt_ns+0x78
free_nsproxy+0x1f
switch_task_namespaces+0x50
exit_task_namespaces+0x10
do_exit+0x456
do_group_exit+0x3f
sys_exit_group+0x17
system_call_fastpath+0x16
RIP rpc_create+0x401 [sunrpc]
Kernel panic - not syncing

>