
Subject: [RFC PATCH 0/5] SUNRPC: "RPC pipefs per network namespace" preparations

Posted by [Stanislav Kinsbursky](#) on Mon, 17 Oct 2011 12:13:09 GMT

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

Hello to everyone.

RPC pipefs file system have to work per network namespace context is required prior to any NFS modifications.

This is a way how to do it. I'll really appreciate for any comments.

There are several statements about how to make RPC pipefs working per network namespace context.

Here they are:

- 1) RPC pipefs should be mounted per network namespace context.
- 2) RPC pipefs superblock should holds network namespace while active.
- 3) RPC pipefs lookup and readdir should be performed in network namespace context it was mounted. IOW, user-space process, working in another network namespace context, should see RPC pipefs dentries from network namespace context this mount-point was created (like it was done for sysfs).

These statement leads to some restrictions which we must follow during implementation. Here are they:

- 1) RPC pipefs mount can't be performed in kernel context since new super block will holds networks namespace reference and it's impossible to recognize, when and how we have to release this mount point. IOW `rpc_get_mount()` and `rpc_put_mount()` have to be removed.
- 2) RPC pipefs should provide some new helpers to lookup directory dentry for those modules which creates pipes, because without RPC pipefs mount point general lookup can't be performed.
- 3) These methods must guarantee, that pipefs superblock will be active during pipes creation and destruction.

So, here is the idea of making RPC pipefs works per network namespace context:

- 1) RPC pipefs superblock should holds network namespace context while active.
- 2) RPC pipefs should send notification events on superblock creation and destruction.
- 3) RPC pipefs should provide "lookup dentry by name" method for notification subscribers.
- 4) RPC pipefs should place superblock reference on current network namespace context on creation and remove it on destruction.
- 5) RPC pipefs should provide safe "lookup dentry by name" method for per-net operations, which guarantees, that superblock is active, while per-net-operations are performing.
- 6) Client and cache directories creation and destruction should be performed also on superblock creation and destruction notification events. Note: generic creation (like now) can fail (if no superblock is not created yet).
- 7) Pipes creation and destruction should be performed on superblock creation and destruction events. Also pipes operations should be performed during

per-net operation and in this case they could fail (due to the same reason as in statement above).

This patch-set implements first 5 points and thus doesn't affects current RPC pipefs logic.

The only problem about I'm not sure how to solve properly yet, is auth gss pipes creations operations. Hoping for some help with it.

The following series consists of:

Stanislav Kinsbursky (5):

- SUNRPC: hold current network namespace while pipefs superblock is active
- SUNRPC: send notification events on pipefs sb creation and destruction
- SUNRPC: pipefs dentry lookup helper introduced
- SUNRPC: put pipefs superblock link on network namespace
- SUNRPC: pipefs per-net operations helper introduced

```
include/linux/sunrpc/rpc_pipe_fs.h | 16 ++++++
net/sunrpc/netns.h                 | 3 +
net/sunrpc/rpc_pipe.c               | 103 ++++++
net/sunrpc/sunrpc_syms.c           | 1
4 files changed, 122 insertions(+), 1 deletions(-)
```

--

Signature

Subject: Re: [RFC PATCH 0/5] SUNRPC: "RPC pipefs per network namespace" preparations

Posted by [Stanislav Kinsbursky](#) on Thu, 20 Oct 2011 11:06:46 GMT

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

Guys, please, spend some of your expensive time to review this patch-set briefly.

This is not for commit, but just an idea representation.

I really need some opinions about it, since all my further work around RPC pipefs depends on it.

IOW I need to now, does anyone has something against this idea.

Trond, please, respond, does this idea suits you in general or not?

> Hello to everyone.

> RPC pipefs file system have to work per network namespace context is required

> prior to any NFS modifications.

> This is a way how to do it. I'll really appreciate for any comments.

>

> There are several statements about how to make RPC pipefs working per network namespace context.

> Here they are:

> 1) RPC pipefs should be mounted per network namespace context.

> 2) RPC pipefs superblock should holds network namespace while active.

> 3) RPC pipefs lookup and readdir should be performed in network namespace context it was mounted. IOW, user-space process, working in another network namespace context, should see RPC pipefs dentries from network namespace context this mount-point was created (like it was done for sysfs).

>

> These statement leads to some restrictions which we must follow during implementation. Here are they:

> 1) RPC pipefs mount can't be performed in kernel context since new super block will holds networks namespace reference and it's impossible to recognize, when and how we have to release this mount point. IOW `rpc_get_mount()` and `rpc_put_mount()` have to be removed.

> 2) RPC pipefs should provide some new helpers to lookup directory dentry for those modules which creates pipes, because without RPC pipefs mount point general lookup can't be performed.

> 3) These methods must guarantee, that pipefs superblock will be active during pipes creation and destruction.

>

> So, here is the idea of making RPC pipefs works per network namespace context:

> 1) RPC pipefs superblock should holds network namespace context while active.

> 2) RPC pipefs should send notification events on superblock creation and destruction.

> 3) RPC pipefs should provide "lookup dentry by name" method for notification subscribers.

> 4) RPC pipefs should place superblock reference on current network namespace context on creation and remove it on destruction.

> 5) RPC pipefs should provide safe "lookup dentry by name" method for per-net operations, which guarantees, that superblock is active, while per-net-operations are performing.

> 6) Client and cache directories creation and destruction should be performed also on superblock creation and destruction notification events. Note: generic creation (like now) can fail (if no superblock is not created yet).

> 7) Pipes creation and destruction should be performed on superblock creation and destruction events. Also pipes operations should be performed during per-net operation and in this case they could fail (due to the same reason as in statement above).

>

> This patch-set implements first 5 points and thus doesn't affects current RPC pipefs logic.

>

> The only problem about I'm not sure how to solve properly yet, is auth gss pipes creations operations. Hoping for some help with it.

>
>
> The following series consists of:
>
> ---
>
> Stanislav Kinsbursky (5):
> SUNRPC: hold current network namespace while pipefs superblock is active
> SUNRPC: send notification events on pipefs sb creation and destruction
> SUNRPC: pipefs dentry lookup helper introduced
> SUNRPC: put pipefs superblock link on network namespace
> SUNRPC: pipefs per-net operations helper introduced
>
>
> include/linux/sunrpc/rpc_pipe_fs.h | 16 ++++++
> net/sunrpc/netns.h | 3 +
> net/sunrpc/rpc_pipe.c | 103 +++
> net/sunrpc/sunrpc_syms.c | 1
> 4 files changed, 122 insertions(+), 1 deletions(-)
>

--
Best regards,
Stanislav Kinsbursky

Subject: Re: [RFC PATCH 0/5] SUNRPC: "RPC pipefs per network namespace"; preparations
Posted by [bfields](#) on Thu, 20 Oct 2011 12:32:42 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Thu, Oct 20, 2011 at 03:06:46PM +0400, Stanislav Kinsbursky wrote:
> Guys, please, spend some of your expensive time to review this patch-set briefly.

I'll try to take a look soon, but I'm travelling tomorrow through the 31st, and things will be a little hectic.

Just one quick comment:

> >The only problem about I'm not sure how to solve properly yet, is auth gss
> >pipes creations operations. Hoping for some help with it.

I suspect one reason it may be a little complicated is the upcall-version switching. The old version is deprecated, and there's no need to support the combination of the old version with the a new feature like containers. And now that it's been there a while the version-switching code already achieved its goal of avoiding a flag day.

So, one approach might be:

- move all the code for the old gss upcall and for the version switching under a new CONFIG_DEPRECATED_GSS, or similar.
- print a warning if the old stuff is used, and plan to rip it out completely in a future kernel version.
- do something that works just in the !CONFIG_DEPRECATED_GSS case.

Would that help?

--b.

Subject: Re: [RFC PATCH 0/5] SUNRPC: "RPC pipefs per network namespace" preparations

Posted by [Stanislav Kinsbursky](#) on Thu, 20 Oct 2011 12:56:02 GMT

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

> On Thu, Oct 20, 2011 at 03:06:46PM +0400, Stanislav Kinsbursky wrote:

>> Guys, please, spend some of your expensive time to review this patch-set briefly.

>

> I'll try to take a look soon, but I'm travelling tomorrow through the

> 31st, and things will be a little hectic.

>

Thanks for your time, Bruce.

> Just one quick comment:

>

>>> The only problem about I'm not sure how to solve properly yet, is auth gss

>>> pipes creations operations. Hoping for some help with it.

>

> I suspect one reason it may be a little complicated is the

> upcall-version switching. The old version is deprecated, and there's no

> need to support the combination of the old version with the a new

> feature like containers. And now that it's been there a while the

> version-switching code already achieved its goal of avoiding a flag day.

> So, one approach might be:

>

> - move all the code for the old gss upcall and for the version

> switching under a new CONFIG_DEPRECATED_GSS, or similar.

> - print a warning if the old stuff is used, and plan to rip it

> out completely in a future kernel version.

> - do something that works just in the !CONFIG_DEPRECATED_GSS

> case.

>

Thanks for this comment. I'll check the code for problem you mentioned here.
But I was actually talking about other thing.
Currently we create pipe in gss without any checks since we assume, that pipefs client dir is created already.
But with approach, represented in this patch set, pipes and dirs will be created only when pipefs was mounted from user-space. I.e. clients with gss auth may already present and some callback is required for creating gss pipes.
And also this approach assumes existence of gss auth without pipe.

> Would that help?
>
> --b.

--
Best regards,
Stanislav Kinsbursky
