
Subject: [PATCH v3, RESEND 02/16] sunrpc: introduce init_rpc_pipefs
Posted by [Kirill A. Shutemov](#) on Tue, 08 Feb 2011 18:41:53 GMT
[View Forum Message](#) <> [Reply to Message](#)

Introduce global variable init_rpc_pipefs and use it instead of
rpc_get_mount()/rpc_put_mount().

Signed-off-by: Kirill A. Shutemov <kas@openvz.org>
Reviewed-by: Rob Landley <rlandley@parallels.com>

fs/nfs/cache_lib.c | 6 +++---
include/linux/sunrpc/rpc_pipe_fs.h | 4 ++--
net/sunrpc/clnt.c | 10 ++++-----
net/sunrpc/rpc_pipe.c | 21 ++++++-----
4 files changed, 14 insertions(+), 27 deletions(-)

diff --git a/fs/nfs/cache_lib.c b/fs/nfs/cache_lib.c
index 8469031..dd7ca5f 100644

--- a/fs/nfs/cache_lib.c
+++ b/fs/nfs/cache_lib.c
@@ -117,7 +117,7 @@ int nfs_cache_register(struct cache_detail *cd)
 struct vfsmount *mnt;
 int ret;

- mnt = rpc_get_mount();
+ mnt = mntget(init_rpc_pipefs);
 if (IS_ERR(mnt))
 return PTR_ERR(mnt);
 ret = vfs_path_lookup(mnt->mnt_root, mnt, "/cache", 0, &nd);
@@ -129,13 +129,13 @@ int nfs_cache_register(struct cache_detail *cd)
 if (!ret)
 return ret;
err:
- rpc_put_mount();
+ mntput(mnt);
 return ret;
}

void nfs_cache_unregister(struct cache_detail *cd)
{
 sunrpc_cache_unregister_pipefs(cd);
- rpc_put_mount();
+ mntput(init_rpc_pipefs);
}

diff --git a/include/linux/sunrpc/rpc_pipe_fs.h b/include/linux/sunrpc/rpc_pipe_fs.h
index cf14db9..b09bfa5 100644
--- a/include/linux/sunrpc/rpc_pipe_fs.h

```

+++ b/include/linux/sunrpc/rpc_pipe_fs.h
@@ -44,6 +44,8 @@ RPC_I(struct inode *inode)
    return container_of(inode, struct rpc_inode, vfs_inode);
}

+extern struct vfsmount *init_rpc_pipefs;
+
extern int rpc_queue_upcall(struct inode *, struct rpc_pipe_msg *);

struct rpc_clnt;
@@ -60,8 +62,6 @@ extern void rpc_remove_cache_dir(struct dentry *);
extern struct dentry *rpc_mkpipe(struct dentry *, const char *, void *,
    const struct rpc_pipe_ops *, int flags);
extern int rpc_unlink(struct dentry *);
-extern struct vfsmount *rpc_get_mount(void);
-extern void rpc_put_mount(void);
extern int register_rpc_pipefs(void);
extern void unregister_rpc_pipefs(void);

diff --git a/net/sunrpc/clnt.c b/net/sunrpc/clnt.c
index 57d344c..f3812d0 100644
--- a/net/sunrpc/clnt.c
+++ b/net/sunrpc/clnt.c
@@ -112,9 +112,7 @@ rpc_setup_pipedir(struct rpc_clnt *clnt, char *dir_name)
    if (dir_name == NULL)
        return 0;

-    path.mnt = rpc_get_mount();
-    if (IS_ERR(path.mnt))
-        return PTR_ERR(path.mnt);
+    path.mnt = mntget(init_rpc_pipefs);
    error = vfs_path_lookup(path.mnt->mnt_root, path.mnt, dir_name, 0, &nd);
    if (error)
        goto err;
@@ -140,7 +138,7 @@ rpc_setup_pipedir(struct rpc_clnt *clnt, char *dir_name)
err_path_put:
    path_put(&nd.path);
err:
-    rpc_put_mount();
+    mntput(path.mnt);
    return error;
}

@@ -251,7 +249,7 @@ static struct rpc_clnt * rpc_new_client(const struct rpc_create_args *args,
stru
out_no_auth:
    if (!IS_ERR(clnt->cl_path.dentry)) {
        rpc_remove_client_dir(clnt->cl_path.dentry);

```

```

- rpc_put_mount();
+ mntput(clnt->cl_path.mnt);
}
out_no_path:
kfree(clnt->cl_principal);
@@ -472,7 +470,7 @@ rpc_free_client(struct rpc_clnt *clnt)
    clnt->cl_protname, clnt->cl_server);
    if (!IS_ERR(clnt->cl_path.dentry)) {
        rpc_remove_client_dir(clnt->cl_path.dentry);
- rpc_put_mount();
+ mntput(clnt->cl_path.mnt);
    }
    if (clnt->cl_parent != clnt) {
        rpc_release_client(clnt->cl_parent);
diff --git a/net/sunrpc/rpc_pipe.c b/net/sunrpc/rpc_pipe.c
index 9ab9355..484c9a3 100644
--- a/net/sunrpc/rpc_pipe.c
+++ b/net/sunrpc/rpc_pipe.c
@@ -28,7 +28,8 @@
#include <linux/sunrpc/rpc_pipe_fs.h>
#include <linux/sunrpc/cache.h>

-static struct vfsmount *rpc_mnt __read_mostly;
+struct vfsmount *init_rpc_pipefs __read_mostly;
+EXPORT_SYMBOL_GPL(init_rpc_pipefs);

static struct file_system_type rpc_pipe_fs_type;

@@ -420,18 +421,6 @@ struct rpc_filelist {
    umode_t mode;
};

-struct vfsmount *rpc_get_mount(void)
-{
- return mntget(rpc_mnt);
-}
-EXPORT_SYMBOL_GPL(rpc_get_mount);
-
-void rpc_put_mount(void)
-{
- mntput(rpc_mnt);
-}
-EXPORT_SYMBOL_GPL(rpc_put_mount);
-
static int rpc_delete_dentry(const struct dentry *dentry)
{
    return 1;
}
@@ -1068,9 +1057,9 @@ int register_rpc_pipefs(void)

```

```
if (err)
    goto destroy_cache;

- rpc_mnt = kern_mount(&rpc_pipe_fs_type);
- if (IS_ERR(rpc_mnt)) {
-     err = PTR_ERR(rpc_mnt);
+ init_rpc_pipefs = kern_mount(&rpc_pipe_fs_type);
+ if (IS_ERR(init_rpc_pipefs)) {
+     err = PTR_ERR(init_rpc_pipefs);
+     goto unregister_fs;
+ }
```

--

1.7.4

Containers mailing list
Containers@lists.linux-foundation.org
<https://lists.linux-foundation.org/mailman/listinfo/containers>
