
Subject: [RFC][PATCH 1/4] seed /proc root inode in proc_fill_super()
Posted by [Dave Hansen](#) on Fri, 26 Jan 2007 22:42:09 GMT
[View Forum Message](#) <> [Reply to Message](#)

Here's a set of patches which I believe allows multiple instances of /proc to exist. We want these because it allows each pid_namespace to have a completely different view of /proc.

/proc currently has some special code to make sure that the root directory gets set up correctly. It uses the global proc_mnt variable in order to find its way to the root inode.

In order to try to get multiple /proc mounts, I'm attempting to kill this variable. But, I noticed that we have all of the data to fill in the inode's pid in proc_fill_super(), where it takes a wee bit more work in proc_get_sb().

Any reason not to just do it in proc_fill_super()?

```
lxc-dave/fs/proc/inode.c | 9 ++++++++
lxc-dave/fs/proc/root.c | 11 -----
2 files changed, 9 insertions(+), 11 deletions(-)
```

```
diff -puN fs/proc/inode.c~A0-seed-proc-differently fs/proc/inode.c
--- lxc/fs/proc/inode.c~A0-seed-proc-differently 2007-01-26 14:29:16.000000000 -0800
+++ lxc-dave/fs/proc/inode.c 2007-01-26 14:29:16.000000000 -0800
@@ -183,6 +183,7 @@ out_mod:
```

```
int proc_fill_super(struct super_block *s, void *data, int silent)
{
+ struct proc_inode *ei;
  struct inode * root_inode;

  s->s_flags |= MS_NODIRATIME | MS_NOSUID | MS_NOEXEC;
@@ -200,6 +201,14 @@ int proc_fill_super(struct super_block *
  s->s_root = d_alloc_root(root_inode);
  if (!s->s_root)
    goto out_no_root;
+ /* Seed the root directory with a pid so it doesn't need
+ * to be special in base.c. I would do this earlier but
+ * the only task alive when /proc is mounted the first time
```

```

+ * is the init_task and it doesn't have any pids.
+ */
+ ei = PROC_I(root_inode);
+ if (!ei->pid)
+ ei->pid = find_get_pid(1);
+ return 0;

```

out_no_root:

```

diff -puN fs/proc/root.c~A0-seed-proc-differently fs/proc/root.c
--- lxc/fs/proc/root.c~A0-seed-proc-differently 2007-01-26 14:29:16.000000000 -0800
+++ lxc-dave/fs/proc/root.c 2007-01-26 14:29:16.000000000 -0800
@@ -30,17 +30,6 @@ struct proc_dir_entry *proc_sys_root;
static int proc_get_sb(struct file_system_type *fs_type,
int flags, const char *dev_name, void *data, struct vfsmount *mnt)
{
- if (proc_mnt) {
- /* Seed the root directory with a pid so it doesn't need
- * to be special in base.c. I would do this earlier but
- * the only task alive when /proc is mounted the first time
- * is the init_task and it doesn't have any pids.
- */
- struct proc_inode *ei;
- ei = PROC_I(proc_mnt->mnt_sb->s_root->d_inode);
- if (!ei->pid)
- ei->pid = find_get_pid(1);
- }
+ return get_sb_single(fs_type, flags, data, proc_fill_super, mnt);
+ }

```

Containers mailing list
Containers@lists.osdl.org
<https://lists.osdl.org/mailman/listinfo/containers>
