

---

**Subject:** [PATCH 0/3] procfs wrt pid namespace cleanups  
**Posted by** [Daniel Lezcano](#) **on** Mon, 14 Feb 2011 19:16:00 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

This patchset is a cleanup and a preparation to unshare the pid namespace. These prerequisite prepare the next Eric's patchset to give a file descriptor to a namespace and join an existing namespace.

The initial authors of this patchset are Eric Biederman and Oleg Nesterov.

Changelog:

02/14/11 - daniel.lezcano@free.fr  
\* patch 2/3 : fix return ENOMEM and put\_pid\_ns on error  
\* removed buggy patch #4 from the initial patchset

01/31/11 - daniel.lezcano@free.fr:  
\* patch 1/4 : wrapped test in a function  
\* patch 2/4 : handle proc\_pid\_ns\_prepare\_proc error  
\* patch 2/4 : put parent pid\_ns on error  
\* other patches : refreshed against linux-next

---

Containers mailing list  
[Containers@lists.linux-foundation.org](mailto:Containers@lists.linux-foundation.org)  
<https://lists.linux-foundation.org/mailman/listinfo/containers>

---

---

**Subject:** [PATCH 3/3] procfs: kill the global proc\_mnt variable  
**Posted by** [Daniel Lezcano](#) **on** Mon, 14 Feb 2011 19:16:03 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

From: Oleg Nesterov <[oleg@redhat.com](mailto:oleg@redhat.com)>

After the previous cleanup in proc\_get\_sb() the global proc\_mnt has no reasons to exists, kill it.

Signed-off-by: Oleg Nesterov <[oleg@redhat.com](mailto:oleg@redhat.com)>  
Signed-off-by: Eric W. Biederman <[ebiederm@xmission.com](mailto:ebiederm@xmission.com)>  
Signed-off-by: Daniel Lezcano <[daniel.lezcano@free.fr](mailto:daniel.lezcano@free.fr)>

---

fs/proc/inode.c | 2 --  
fs/proc/internal.h | 1 -  
fs/proc/root.c | 7 +++++--  
3 files changed, 4 insertions(+), 6 deletions(-)

```

diff --git a/fs/proc/inode.c b/fs/proc/inode.c
index 176ce4c..ee0f802 100644
--- a/fs/proc/inode.c
+++ b/fs/proc/inode.c
@@ -42,8 +42,6 @@ static void proc_evict_inode(struct inode *inode)
    sysctl_head_put(PROC_I(inode)->sysctl);
}

-struct vfsmount *proc_mnt;
-
 static struct kmem_cache * proc_inode_cachep;

 static struct inode *proc_alloc_inode(struct super_block *sb)
diff --git a/fs/proc/internal.h b/fs/proc/internal.h
index 9ad561d..c03e8d3 100644
--- a/fs/proc/internal.h
+++ b/fs/proc/internal.h
@@ -107,7 +107,6 @@ static inline struct proc_dir_entry *pde_get(struct proc_dir_entry *pde)
}
void pde_put(struct proc_dir_entry *pde);

-extern struct vfsmount *proc_mnt;
int proc_fill_super(struct super_block *);
struct inode *proc_get_inode(struct super_block *, struct proc_dir_entry *);

diff --git a/fs/proc/root.c b/fs/proc/root.c
index e5e2bfa..a9000e9 100644
--- a/fs/proc/root.c
+++ b/fs/proc/root.c
@@ -90,19 +90,20 @@ static struct file_system_type proc_fs_type = {

void __init proc_root_init(void)
{
+ struct vfsmount *mnt;
int err;

proc_init_inodecache();
err = register_filesystem(&proc_fs_type);
if (err)
return;
- proc_mnt = kern_mount_data(&proc_fs_type, &init_pid_ns);
- if (IS_ERR(proc_mnt)) {
+ mnt = kern_mount_data(&proc_fs_type, &init_pid_ns);
+ if (IS_ERR(mnt)) {
unregister_filesystem(&proc_fs_type);
return;
}

```

```
- init_pid_ns.proc_mnt = proc_mnt;
+ init_pid_ns.proc_mnt = mnt;
proc_symlink("mounts", NULL, "self/mounts");

proc_net_init();
--
```

1.7.1

---

Containers mailing list

Containers@lists.linux-foundation.org

<https://lists.linux-foundation.org/mailman/listinfo/containers>

---

---

Subject: Re: [PATCH 0/3] procfs wrt pid namespace cleanups

Posted by [Oleg Nesterov](#) on Mon, 14 Feb 2011 19:47:15 GMT

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

---

On 02/14, Daniel Lezcano wrote:

>  
> \* removed buggy patch #4 from the initial patchset

Hmm. probably it was me who confused you... I am sorry if this is true.

I don't think that patch was buggy. What I meant, it looks like  
the preparation for the next changes  
( <http://kerneltrap.org/mailarchive/linux-kernel/2010/6/20/4585095> )  
and those changes are not correct.

In fact, 1/3 looks the same.

Anyway, this series looks correct.

Oleg.

---

Containers mailing list

Containers@lists.linux-foundation.org

<https://lists.linux-foundation.org/mailman/listinfo/containers>

---

---

Subject: Re: [PATCH 3/3] procfs: kill the global proc\_mnt variable

Posted by [Serge E. Hallyn](#) on Tue, 15 Feb 2011 18:50:02 GMT

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

---

Quoting Daniel Lezcano (daniel.lezcano@free.fr):

> From: Oleg Nesterov <oleg@redhat.com>  
>  
> After the previous cleanup in proc\_get\_sb() the global proc\_mnt has  
> no reasons to exists, kill it.  
>  
> Signed-off-by: Oleg Nesterov <oleg@redhat.com>  
> Signed-off-by: Eric W. Biederman <ebiederm@xmission.com>  
> Signed-off-by: Daniel Lezcano <daniel.lezcano@free.fr>

Acked-by: Serge E. Hallyn <serge@hallyn.com>

> ---  
> fs/proc/inode.c | 2 --  
> fs/proc/internal.h | 1 -  
> fs/proc/root.c | 7 +++++--  
> 3 files changed, 4 insertions(+), 6 deletions(-)  
>  
> diff --git a/fs/proc/inode.c b/fs/proc/inode.c  
> index 176ce4c..ee0f802 100644  
> --- a/fs/proc/inode.c  
> +++ b/fs/proc/inode.c  
> @@ -42,8 +42,6 @@ static void proc\_evict\_inode(struct inode \*inode)  
> sysctl\_head\_put(PROC\_I(inode)->sysctl);  
> }  
>  
> -struct vfsmount \*proc\_mnt;  
> -  
> static struct kmem\_cache \* proc\_inode\_cachep;  
>  
> static struct inode \*proc\_alloc\_inode(struct super\_block \*sb)  
> diff --git a/fs/proc/internal.h b/fs/proc/internal.h  
> index 9ad561d..c03e8d3 100644  
> --- a/fs/proc/internal.h  
> +++ b/fs/proc/internal.h  
> @@ -107,7 +107,6 @@ static inline struct proc\_dir\_entry \*pde\_get(struct proc\_dir\_entry \*pde)  
> }  
> void pde\_put(struct proc\_dir\_entry \*pde);  
>  
> -extern struct vfsmount \*proc\_mnt;  
> int proc\_fill\_super(struct super\_block \*);  
> struct inode \*proc\_get\_inode(struct super\_block \*, struct proc\_dir\_entry \*);  
>  
> diff --git a/fs/proc/root.c b/fs/proc/root.c  
> index e5e2bfa..a9000e9 100644  
> --- a/fs/proc/root.c  
> +++ b/fs/proc/root.c  
> @@ -90,19 +90,20 @@ static struct file\_system\_type proc\_fs\_type = {  
>

```
> void __init proc_root_init(void)
> {
> + struct vfsmount *mnt;
> int err;
>
> proc_init_inodecache();
> err = register_filesystem(&proc_fs_type);
> if (err)
> return;
> - proc_mnt = kern_mount_data(&proc_fs_type, &init_pid_ns);
> - if (IS_ERR(proc_mnt)) {
> + mnt = kern_mount_data(&proc_fs_type, &init_pid_ns);
> + if (IS_ERR(mnt)) {
>     unregister_filesystem(&proc_fs_type);
>     return;
> }
>
> - init_pid_ns.proc_mnt = proc_mnt;
> + init_pid_ns.proc_mnt = mnt;
> proc_symlink("mounts", NULL, "self/mounts");
>
> proc_net_init();
> --
> 1.7.1
>
> _____
```

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

---

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

---