
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_cache;

>

> 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

```
