
Subject: Re: namespace and nsproxy syscalls
Posted by [serue](#) on Tue, 26 Sep 2006 13:03:40 GMT
[View Forum Message](#) <> [Reply to Message](#)

Quoting Cedric Le Goater (clg@fr.ibm.com):

Looks ok, except for maintainability the following change might make sense:

```
> +asmlinkage long sys_unshare_ns(unsigned long unshare_ns_flags)
> +{
> + int err = 0;
> + struct nsproxy *new_nsproxy = NULL, *old_nsproxy = NULL;
> + struct fs_struct *fs, *new_fs = NULL;
> + struct mnt_namespace *mnt, *new_mnt = NULL;
> + struct uts_namespace *uts, *new_uts = NULL;
> + struct ipc_namespace *ipc, *new_ipc = NULL;
> + unsigned long unshare_flags = 0;
> +
> + /* Return -EINVAL for all unsupported flags */
> + err = -EINVAL;
> + if (unshare_ns_flags & ~(UNSHARE_NS_MNT|UNSHARE_NS_UTS|UNSHARE_NS_IPC|
> +   UNSHARE_NS_USER|UNSHARE_NS_NET|
> +   UNSHARE_NS_PID))
> + goto bad_unshare_ns_out;
```

Get rid of this explicit check against unshare_ns_flags, and

```
> + /* convert unshare_ns flags to clone flags */
> + if (unshare_ns_flags & UNSHARE_NS_MNT)
> +   unshare_flags |= CLONE_NEWNS|CLONE_FS;
> + if (unshare_ns_flags & UNSHARE_NS_UTS)
> +   unshare_flags |= CLONE_NEWUTS;
> + if (unshare_ns_flags & UNSHARE_NS_IPC)
> +   unshare_flags |= CLONE_NEWIPC;
```

Just check here if unshare_flags is still 0, i.e.:

```
if (unshare_flags == 0)
    goto bad_unshare_ns_out;

> + if ((err = unshare_fs(unshare_flags, &new_fs)))
> + goto bad_unshare_ns_out;
> + if ((err = unshare_mnt_namespace(unshare_flags, &new_mnt, new_fs)))
> + goto bad_unshare_ns_cleanup_fs;
> + if ((err = unshare_utsname(unshare_flags, &new_uts)))
> + goto bad_unshare_ns_cleanup_mnt;
> + if ((err = unshare_ipcs(unshare_flags, &new_ipc)))
```

```

> + goto bad_unshare_ns_cleanup_uts;
> +
> + if (new_mnt || new_uts || new_ipc) {
> +   old_nsproxy = current->nsproxy;
> +   new_nsproxy = dup_namespaces(old_nsproxy);
> +   if (!new_nsproxy) {
> +     err = -ENOMEM;
> +     goto bad_unshare_ns_cleanup_ipc;
> +   }
> + }
> +
> + if (new_fs || new_mnt || new_uts || new_ipc) {
> +
> +   task_lock(current);
> +
> +   if (new_nsproxy) {
> +     current->nsproxy = new_nsproxy;
> +     new_nsproxy = old_nsproxy;
> +   }
> +
> +   if (new_fs) {
> +     fs = current->fs;
> +     current->fs = new_fs;
> +     new_fs = fs;
> +   }
> +
> +   if (new_mnt) {
> +     mnt = current->nsproxy->mnt_ns;
> +     current->nsproxy->mnt_ns = new_mnt;
> +     new_mnt = mnt;
> +   }
> +
> +   if (new_uts) {
> +     uts = current->nsproxy->uts_ns;
> +     current->nsproxy->uts_ns = new_uts;
> +     new_uts = uts;
> +   }
> +
> +   if (new_ipc) {
> +     ipc = current->nsproxy->ipc_ns;
> +     current->nsproxy->ipc_ns = new_ipc;
> +     new_ipc = ipc;
> +   }
> +
> +   task_unlock(current);
> + }
> +
> + if (new_nsproxy)

```

```
> + put_nsproxy(new_nsproxy);
> +
> +bad_unshare_ns_cleanup_ipc:
> + if (new_ipc)
> + put_ipc_ns(new_ipc);
> +
> +bad_unshare_ns_cleanup_uts:
> + if (new_uts)
> + put_uts_ns(new_uts);
> +
> +bad_unshare_ns_cleanup_mnt:
> + if (new_mnt)
> + put_mnt_ns(new_mnt);
> +
> +bad_unshare_ns_cleanup_fs:
> + if (new_fs)
> + put_fs_struct(new_fs);
> +
> +bad_unshare_ns_out:
> + return err;
> +}
>


---


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

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