
Subject: Re: [PATCH 9/9] userns: check user namespace for task->file uid equivalence checks

Posted by [akpm](#) on Fri, 18 Feb 2011 23:59:35 GMT

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

On Thu, 17 Feb 2011 15:04:07 +0000

"Serge E. Hallyn" <serge@hallyn.com> wrote:

> Cheat for now and say all files belong to init_user_ns. Next
> step will be to let superblocks belong to a user_ns, and derive
> inode_userns(inode) from inode->i_sb->s_user_ns. Finally we'll
> introduce more flexible arrangements.

>

>

> ...

>

> +

> +/*

> + * return 1 if current either has CAP_FOWNER to the

> + * file, or owns the file.

> + */

> +int is_owner_or_cap(const struct inode *inode)

> +{

> + struct user_namespace *ns = inode_userns(inode);

> +

> + if (current_user_ns() == ns && current_fsuid() == inode->i_uid)

> + return 1;

> + if (ns_capable(ns, CAP_FOWNER))

> + return 1;

> + return 0;

> +}

bool?

> +EXPORT_SYMBOL(is_owner_or_cap);

There's a fairly well adhered to convention that global symbols (and often static symbols) have a prefix which identifies the subsystem to which they belong. This patchset rather scorns that convention.

Most of these identifiers are pretty obviously from the capability subsystem, but still...

>

> ...

>

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