

---

Subject: Re: [PATCH 9/9] userns: check user namespace for task-&gt;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](mailto: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>

---