

---

Subject: Containers don't handle keys, but should they?  
Posted by [David Howells](#) on Fri, 14 Mar 2008 11:37:59 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Am I right in thinking that a UID in one container is not necessarily equivalent to the numerically equivalent UID in another container?

If that's the case then the key management code will need changing as it assumes all keys belonging to one numeric UID eat out of the same quota and the numeric UIDs are used in security checks.

Furthermore, processes in one container can access keys created by a process in another container by ID. Is this desirable or not?

David

---

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

---

---

Subject: Re: Containers don't handle keys, but should they?  
Posted by [Kirill Korotaev](#) on Fri, 14 Mar 2008 11:44:38 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

yes. If I understand correct key management requires containerization (i.e. "virtualization") as well other subsystems like IPC dealing with IDs.

Processes from one container should not be able to access keys from another container.

David Howells wrote:

> Am I right in thinking that a UID in one container is not necessarily  
> equivalent to the numerically equivalent UID in another container?  
>  
> If that's the case then the key management code will need changing as it  
> assumes all keys belonging to one numeric UID eat out of the same quota and  
> the numeric UIDs are used in security checks.  
>  
> Furthermore, processes in one container can access keys created by a process  
> in another container by ID. Is this desirable or not?  
>  
> David  
>  
> \_\_\_\_\_  
> Containers mailing list  
> [Containers@lists.linux-foundation.org](mailto:Containers@lists.linux-foundation.org)  
> <https://lists.linux-foundation.org/mailman/listinfo/containers>  
>

---

---

Subject: Re: Containers don't handle keys, but should they?

Posted by [serue](#) on Fri, 14 Mar 2008 14:54:47 GMT

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

---

Quoting David Howells (dhowells@redhat.com):

>  
> Am I right in thinking that a UID in one container is not necessarily  
> equivalent to the numerically equivalent UID in another container?  
>  
> If that's the case then the key management code will need changing as it  
> assumes all keys belonging to one numeric UID eat out of the same quota and  
> the numeric UIDs are used in security checks.  
>  
> Furthermore, processes in one container can access keys created by a process  
> in another container by ID. Is this desirable or not?  
>  
> David

Yes, the confusion comes from using the word 'container' which doesn't really exist. The user namespaces (CLONE\_NEWUSER) are what provide separate of uids. We want uid 5 in one user namespace to have completely separate set of keys from uid 5 in another user namespace.

This isn't yet a crucial thing to get right as the user namespaces are only partially implemented, but it's certainly a good thing to be looking at and fix when convenient to do so. It looks like maybe just adding a struct user\_namespace \* to a struct key should suffice.

-serge

---

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

---

---

Subject: Re: Containers don't handle keys, but should they?

Posted by [David Howells](#) on Fri, 14 Mar 2008 15:49:20 GMT

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

---

Serge E. Hallyn <serue@us.ibm.com> wrote:

> It looks like maybe just adding a struct user\_namespace \* to a struct key

> should suffice.

That's not quite sufficient. The per-UID key\_user structs also need to be differentiated. Unfortunately, I can't just merge it into user\_struct as I then end up with a reference loop user\_struct -> uid\_keyring -> user\_struct.

Rooting the key\_user trees in user\_namespace will probably do the trick.

A couple of questions:

- (1) A process may inherit a session keyring over clone(). Should this be discarded if CLONE\_NEWUSER is set? Or would I need to copy it?
- (2) In a recent patch, I've given the root user its own quota limits. Is UID 0 always the root user in any container? Or would it make more sense just to scrap the per-root quota limits?

David

---

Containers mailing list

Containers@lists.linux-foundation.org

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

---

---

Subject: Re: Containers don't handle keys, but should they?

Posted by [serue](#) on Fri, 14 Mar 2008 16:17:11 GMT

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

---

Quoting David Howells (dhowells@redhat.com):

> Serge E. Hallyn <serue@us.ibm.com> wrote:

>

> > It looks like maybe just adding a struct user\_namespace \* to a struct key

> > should suffice.

>

> That's not quite sufficient. The per-UID key\_user structs also need to be

> differentiated. Unfortunately, I can't just merge it into user\_struct as I

> then end up with a reference loop user\_struct -> uid\_keyring -> user\_struct.

>

> Rooting the key\_user trees in user\_namespace will probably do the trick.

>

> A couple of questions:

>

> (1) A process may inherit a session keyring over clone(). Should this be

> discarded if CLONE\_NEWUSER is set? Or would I need to copy it?

Someone else may have stronger feelings about this. Personally so long as a container setup program has a way of discarding the keyring manually I think that's fine.

- > (2) In a recent patch, I've given the root user its own quota limits. Is UID
- > 0 always the root user in any container? Or would it make more sense
- > just to scrap the per-root quota limits?

Yeah uid 0 may not have a bunch of privileges, but it is still the root user.

thanks,  
-serge

---

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

---