Subject: Re: + user-ns-implement-user-ns-unshare-remove-config_user_ns.patch added to -mm tree
Posted by ebiederm on Thu, 25 Jan 2007 19:38:31 GMT
View Forum Message <> Reply to Message

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

>> As it sits right now using the user namespace instead of being an
>> enhancement of security as it should feels like security loophole
>> 101.
>
> That's a bit of a callous exaggeration, don't you think? It takes what
> used to be one big pool and partitions it, offering isolation between
> partitions with one clearly defined exception. Given that there used to
> be no partitions at all, this still nets added isolation over the
> original, no loopholes.

Except from what I could tell from my quick review the partition is far from complete. That concerns me. Especially the way this is expected to be used it to setup a user who appears to be root in his partition.

The fact that for any non-filesystem based permission check he appears to be root to the rest of the system disturbs me.

Now maybe I'm blind but that is my current perception of the situation and until the partition is complete I don't want people thinking it is.

The little minor details about your exception are important but not really what concerned me at this point.

>> For the stable namespaces I'm fine with removing the config options
>> but the user namespace is not at all complete.
>
> I'm fine with that, it certainly is still very new.
>
> I was liking Cedric's patch because with the CONFIG_USER_NS option,
> we just end up with all the more corner cases to test, so Cedric's
> patch, considering how little ends up actually CONFIG'd out, actually
> seems an improvement.

As long as we retain the ability to compile out the ability to actually create a second user namespace I am happy. I have no problem with running all of the code with just the initial user namespace present.

My apologies for not being able to have the security conversation yet and not having done a more thorough review.

The truth is that coming anywhere near the user namespace I find scary because of all it's security implications. Almost by definition any bug there is a security issue.

If we limit ourselves to exactly allowing overlapping uid and gids by making the comparisons for equality include their corresponding namespace, and we look carefully to ensure we get every such comparison. I feel fairly comfortable in saying we have simply added a slightly more sophisticated form of naming users but the current security model remains. If we deviate from that one iota I figure we need to completely think through the new rules very carefully as we have changed the rules and sometimes innocuous changes combined to allow unexpected loop holes.

I think it is healthy to be paranoid about security issues, isn't it?

So in summary my only real complaint with removing CONFIG_USER_NS is that it appears to me that the code is incomplete and has not been closely scrutinized. As such making it available to end users without even a warning when that is the case appears irresponsible. Especially as much of the code that is sitting in Andrews tree is merged into the production kernel, when the window opens.

Eric

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

Subject: Re: + user-ns-implement-user-ns-unshare-remove-config_user_ns.patch added to -mm tree
Posted by serue on Thu, 25 Jan 2007 20:32:06 GMT

View Forum Message <> Reply to Message

Quoting Eric W. Biederman (ebiederm@xmission.com):
> "Serge E. Hallyn" <serue@us.ibm.com> writes:
> So in summary my only real complaint with removing CONFIG_USER_NS is
> that it appears to me that the code is incomplete and has not been
> closely scrutinized. As such making it available to end users without

Valid complaint.

> even a warning when that is the case appears irresponsible.
> Especially as much of the code that is sitting in Andrews tree is

> merged into the production kernel, when the window opens.

An experimental marker like Cedric introduced does seem a good idea.

It's just too bad that it complicates the testing quite a bit.
I'm still not sure whether just running ltp on a CONFIG_USER_NS=n
kernel suffices or whether custom testcases are needed.

-serge

Subject: Re: + user-ns-implement-user-ns-unshare-remove-config_user_ns.patch added to -mm tree
Posted by serue on Thu, 25 Jan 2007 20:46:35 GMT

View Forum Message <> Reply to Message

Quoting Eric W. Biederman (ebiederm@xmission.com):
> "Serge E. Hallyn" <serue@us.ibm.com> writes:
>
>
> >> As it sits right now using the user namespace instead of being an
> >> enhancement of security as it should feels like security loophole
> >> 101.
> >
> > That's a bit of a callous exaggeration, don't you think? It takes what
> > used to be one big pool and partitions it, offering isolation between
> > partitions with one clearly defined exception. Given that there used to
> > be no partitions at all, this still nets added isolation over the
> > original, no loopholes.
>
> Except from what I could tell from my quick review the partition is far
> from complete. That concerns me. Especially the way this is expected
> to be used it to setup a user who appears to be root in his partition.
>
> The fact that for any non-filesystem based permission check he appears
> to be root to the rest of the system disturbs me.

Allow me to re-ask a fundamental question: do we want the uid namespace
to stick to turning uid checks into (uid,ns) checks? or do we want the
uid namespaces to try to protect against root in other namespaces?

If we go with the first, we can always enforce protection against root
in other namespaces using LSMs. SELinux users have what they need, and
others can use a trivial new LSM.

Eric? Herbert? Kirill? Cedric? Anyone with an opinion?

-serge

Subject: Re: + user-ns-implement-user-ns-unshare-remove-config_user_ns.patch added to -mm tree
Posted by ebiederm on Fri, 26 Jan 2007 06:48:06 GMT

View Forum Message <> Reply to Message

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

> Allow me to re-ask a fundamental question: do we want the uid namespace
> to stick to turning uid checks into (uid,ns) checks? or do we want the
> uid namespaces to try to protect against root in other namespaces?

I am fairly certain we want to at least make the checks (uid, ns) checks.
That gives a minimal level of protection against root in other namespaces,
as the lesser root does not match the (uid, ns) check for the system root.

Exactly how capabilities play into this I'm not quite certain, but something
important to understand. Especially for suid root executables.

> If we go with the first, we can always enforce protection against root
> in other namespaces using LSMs. SELinux users have what they need, and
> others can use a trivial new LSM.

What is the hole you see with root in other namespaces that needs an
LSM, the only hole I know of currently is the incomplete state of the
(uid/gid, ns) checks.

Not that I don't think an LSM couldn't improve the situation.
Although if I have to deal with the LSM insanity much more I'm going
to lobby for changing the concept it to an interapplication firewall,
and get all of the stupid code into the kernel.

Eric

Subject: Re: + user-ns-implement-user-ns-unshare-remove-config_user_ns.patch added to -mm tree
Posted by Cedric Le Goater on Fri, 26 Jan 2007 13:53:55 GMT

View Forum Message <> Reply to Message

Serge E. Hallyn wrote:
> Quoting Eric W. Biederman (ebiederm@xmission.com):
>> "Serge E. Hallyn" <serue@us.ibm.com> writes:
>> So in summary my only real complaint with removing CONFIG_USER_NS is
>> that it appears to me that the code is incomplete and has not been
>> closely scrutinized. As such making it available to end users without
>
> Valid complaint.
>
>> even a warning when that is the case appears irresponsible.
>> Especially as much of the code that is sitting in Andrews tree is
>> merged into the production kernel, when the window opens.
>
> An experimental marker like Cedric introduced does seem a good idea.

Current -mm contains a fix from Andrew which forces user namespace
to Y by default. I'll wait for the next -mm to rework the CONFIG_USER_NS
if the patchset survives andrew's indigestion :(

Sorry about that.

C.

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

Subject: Re: + user-ns-implement-user-ns-unshare-remove-config_user_ns.patch added to -mm tree
Posted by Cedric Le Goater on Fri, 26 Jan 2007 14:40:31 GMT

View Forum Message <> Reply to Message

Eric W. Biederman wrote:
> "Serge E. Hallyn" <serue@us.ibm.com> writes:
>
>> Allow me to re-ask a fundamental question: do we want the uid namespace
>> to stick to turning uid checks into (uid,ns) checks? or do we want the
>> uid namespaces to try to protect against root in other namespaces?
>
> I am fairly certain we want to at least make the checks (uid, ns) checks.
> That gives a minimal level of protection against root in other namespaces,
> as the lesser root does not match the (uid, ns) check for the system root.

I agree that we need the (uid, ns) checks. I would say that these are the next
steps to complete the user namespace feature and remove its experimental
status. But I'm glad that the kernel supports the initial framework, it
should make it easier to fill in the gap. IMO.

Note that we did the opposite with the pid namespace, clean up the kernel
before doing the framework, but the framework is also much more complex to
get right.

C.

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

Subject: Re: + user-ns-implement-user-ns-unshare-remove-config_user_ns.patch added to -mm tree
Posted by serue on Fri, 26 Jan 2007 16:37:56 GMT

View Forum Message <> Reply to Message

Quoting Eric W. Biederman (ebiederm@xmission.com):
> "Serge E. Hallyn" <serue@us.ibm.com> writes:
>
> > Allow me to re-ask a fundamental question: do we want the uid namespace
> > to stick to turning uid checks into (uid,ns) checks? or do we want the
> > uid namespaces to try to protect against root in other namespaces?
>
> I am fairly certain we want to at least make the checks (uid, ns) checks.

Well we do that, the question is whether we want to stick to just that.

> That gives a minimal level of protection against root in other namespaces,

Not really.

> as the lesser root does not match the (uid, ns) check for the system root.

But it will pass capable(CAP_DAC_OVERRIDE) and such checks.

> Exactly how capabilities play into this I'm not quite certain, but something
> important to understand. Especially for suid root executables.
>
> > If we go with the first, we can always enforce protection against root
> > in other namespaces using LSMs. SELinux users have what they need, and
> > others can use a trivial new LSM.

>
> What is the hole you see with root in other namespaces that needs an
> LSM, the only hole I know of currently is the incomplete state of the
> (uid/gid, ns) checks.

The hole is that most permission checks are of the form

```
 if (uid1 == uid2 || capable(CAP_DAC_OVERRIDE))
  allow permission;
```

The root user in a vserver needs CAP_DAC_OVERRIDE within his own userns,
so we can't just take that away.

My patch is one way to handle that for files. Another, arguably
cleaner approach, would be to not handle the problem with user
namespaces, but do so with security modules. But that does feel
like a leak across user namespaces.

> Not that I don't think an LSM couldn't improve the situation.
> Although if I have to deal with the LSM insanity much more I'm going
> to lobby for changing the concept it to an interapplication firewall,

A whatzit?

> and get all of the stupid code into the kernel.

It sounds like you're having an interesting problem - would love to
see it explained on the lsm or selinux list.

-serge
____
Containers mailing list
Containers@lists.osdl.org
https://lists.osdl.org/mailman/listinfo/containers