Subject: [RFC][PATCH 0/2] user namespace [try #2]
Posted by Cedric Le Goater on Mon, 28 Aug 2006 14:56:30 GMT
View Forum Message <> Reply to Message

Hi all,

Here's a second version. It's very close from the first one and takes into account some discussions we had with kirill on the topic during OLS. 2 patches follow, the first introduces the user namespace core and the last enables to use it with unshare.

Changes [try #2]

 - removed struct user_namespace* argument from find_user()
 - added a root_user per user namespace

execns() syscall is back in the attic for the moment. I'm still maintaining it and we'll see if it's of any use when we address the user space API of the full conainer. soon, I hope !

This user namespace patchset does not try to address all the issues that were raised by the previous thread on the topic, like user mapping per namespace, per mount, etc. It tries to solve some simple issues with the current implementation of containers in mind. It should be especially useful the existing solutions and lay ground basic objects.

thanks for your comments,

C.

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

Subject: Re: [RFC][PATCH 0/2] user namespace [try #2]
Posted by Cedric Le Goater on Wed, 30 Aug 2006 20:27:16 GMT
View Forum Message <> Reply to Message

Cedric Le Goater wrote:
> Hi all,
>
> Here's a second version. It's very close from the first one and takes into
> account some discussions we had with kirill on the topic during OLS. 2
> patches follow, the first introduces the user namespace core and the last
> enables to use it with unshare.
>
> Changes [try #2]

>
> - removed struct user_namespace* argument from find_user()
> - added a root_user per user namespace
>
> execns() syscall is back in the attic for the moment. I'm still maintaining
> it and we'll see if it's of any use when we address the user space API of
> the full conainer. soon, I hope !
>
> This user namespace patchset does not try to address all the issues that
> were raised by the previous thread on the topic, like user mapping per
> namespace, per mount, etc. It tries to solve some simple issues with the
> current implementation of containers in mind. It should be especially
> useful the existing solutions and lay ground basic objects.
>
> thanks for your comments,

I didn't get much comments on that one. is everybody happy with it ? can we
merge ask andrew to merge in -mm ?

thanks,

C.

____

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

Subject: Re: [RFC][PATCH 0/2] user namespace [try #2]
Posted by serue on Thu, 31 Aug 2006 03:55:16 GMT

View Forum Message <> Reply to Message

Quoting Cedric Le Goater (clg@fr.ibm.com):
> Cedric Le Goater wrote:
> > Hi all,
> >
> > Here's a second version. It's very close from the first one and takes into
> > account some discussions we had with kirill on the topic during OLS. 2
> > patches follow, the first introduces the user namespace core and the last
> > enables to use it with unshare.
> >
> > Changes [try #2]
> >
> > - removed struct user_namespace* argument from find_user()
> > - added a root_user per user namespace
> >
> > execns() syscall is back in the attic for the moment. I'm still maintaining
> > it and we'll see if it's of any use when we address the user space API of

> > the full conainer. soon, I hope !
> >
> > This user namespace patchset does not try to address all the issues that
> > were raised by the previous thread on the topic, like user mapping per
> > namespace, per mount, etc. It tries to solve some simple issues with the
> > current implementation of containers in mind. It should be especially
> > useful the existing solutions and lay ground basic objects.
> >
> > thanks for your comments,
>
> I didn't get much comments on that one. is everybody happy with it ? can we
> merge ask andrew to merge in -mm ?
>
> thanks,

Ideally we could collect Acked-by: or Signed-off-by: from Eric, Kir or
Kirill, and Herbert or Sam, to show we are all in agreement.

Or a NACK :)

-serge

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

Subject: Re: [RFC][PATCH 0/2] user namespace [try #2]
Posted by ebiederm on Thu, 31 Aug 2006 15:17:57 GMT

View Forum Message <> Reply to Message

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

> Quoting Cedric Le Goater (clg@fr.ibm.com):
>> Cedric Le Goater wrote:
>> > Hi all,
>> >
>> > Here's a second version. It's very close from the first one and takes into
>> > account some discussions we had with kirill on the topic during OLS. 2
>> > patches follow, the first introduces the user namespace core and the last
>> > enables to use it with unshare.
>> >
>> > Changes [try #2]
>> >
>> > - removed struct user_namespace* argument from find_user()
>> > - added a root_user per user namespace
>> >
>> > execns() syscall is back in the attic for the moment. I'm still maintaining

>> > it and we'll see if it's of any use when we address the user space API of
>> > the full conainer. soon, I hope !
>> >
>> > This user namespace patchset does not try to address all the issues that
>> > were raised by the previous thread on the topic, like user mapping per
>> > namespace, per mount, etc. It tries to solve some simple issues with the
>> > current implementation of containers in mind. It should be especially
>> > useful the existing solutions and lay ground basic objects.
>> >
>> > thanks for your comments,
>>
>> I didn't get much comments on that one. is everybody happy with it ? can we
>> merge ask andrew to merge in -mm ?
>>
>> thanks,
>
> Ideally we could collect Acked-by: or Signed-off-by: from Eric, Kir or
> Kirill, and Herbert or Sam, to show we are all in agreement.
>
> Or a NACK :)

Ok for the collection

Nacked-by: Eric Biederman

My gut feel is that this is terribly incomplete, and doesn't
come with enough description to tell me why it could possibly be complete.

I don't think this addresses any of my primary objections from last
round.

This doesn't change the kernel to make uid comparisons as (uid_ns, uid)
tuples or explain why that isn't necessary. It doesn't touch keys.
it doesn't explain why we are not introducing possibly subtle security problems.

Cedric sorry for not saying so earlier, but I thought that the incompleteness
was obvious.

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

Subject: Re: [RFC][PATCH 0/2] user namespace [try #2]

Posted by serue on Thu, 31 Aug 2006 15:59:22 GMT

View Forum Message <> Reply to Message

Quoting Eric W. Biederman (ebiederm@xmission.com):
> "Serge E. Hallyn" <serue@us.ibm.com> writes:
>
> > Quoting Cedric Le Goater (clg@fr.ibm.com):
> >> Cedric Le Goater wrote:
> >> > Hi all,
> >> >
> >> > Here's a second version. It's very close from the first one and takes into
> >> > account some discussions we had with kirill on the topic during OLS. 2
> >> > patches follow, the first introduces the user namespace core and the last
> >> > enables to use it with unshare.
> >> >
> >> > Changes [try #2]
> >> >
> >> > - removed struct user_namespace* argument from find_user()
> >> > - added a root_user per user namespace
> >> >
> >> > execns() syscall is back in the attic for the moment. I'm still maintaining
> >> > it and we'll see if it's of any use when we address the user space API of
> >> > the full conainer. soon, I hope !
> >> >
> >> > This user namespace patchset does not try to address all the issues that
> >> > were raised by the previous thread on the topic, like user mapping per
> >> > namespace, per mount, etc. It tries to solve some simple issues with the
> >> > current implementation of containers in mind. It should be especially
> >> > useful the existing solutions and lay ground basic objects.
> >> >
> >> > thanks for your comments,
> >>
> >> I didn't get much comments on that one. is everybody happy with it ? can we
> >> merge ask andrew to merge in -mm ?
> >>
> >> thanks,
> >
> > Ideally we could collect Acked-by: or Signed-off-by: from Eric, Kir or
> > Kirill, and Herbert or Sam, to show we are all in agreement.
> >
> > Or a NACK :)
>
> Ok for the collection
> Nacked-by: Eric Biederman

Thanks :)

> My gut feel is that this is terribly incomplete, and doesn't
> come with enough description to tell me why it could possibly be complete.

>
> I don't think this addresses any of my primary objections from last
> round.
>
> This doesn't change the kernel to make uid comparisons as (uid_ns, uid)
> tuples or explain why that isn't necessary. It doesn't touch keys.
> it doesn't explain why we are not introducing possibly subtle security problems.
>
> Cedric sorry for not saying so earlier, but I thought that the incompleteness
> was obvious.

(ignoring keys and other uid actions for now)

Here's a stab at semantics for how to handle file access. Should be
pretty simple to implement, but i won't get a chance to implement this
week.

At mount, by default the vfsmount is tagged with a uid_ns.
A new -o uid_ns=<pid> option instead tags the vfsmount with the uid_ns
 belonging to pid <pid>. Since any process in a descendent pid
 namespace should still have a valid pid in the ancestor
 pidspaces, this should work fine.
At vfs_permission, if current->nsproxy->uid_ns != file->f_vfsmnt->uid_ns,
 1. If file is owned by root, then read permission is granted
 2. If file is owned by non-root, no permission is granted
(regardless of process uid)

Does this sound reasonable?

I assume the list of other things we'll need to consider includes
 signals between user namespaces
 keystore
 sys_setpriority and the like
I might argue that all of these should be sufficiently protected
by proper setup by userspace. Can you explain why that is not
the case?

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

Subject: Re: [RFC][PATCH 0/2] user namespace [try #2]
Posted by ebiederm on Thu, 31 Aug 2006 17:02:35 GMT

"Serge E. Hallyn" <serue@us.ibm.com> writes:
>
> Thanks :)
>
>> My gut feel is that this is terribly incomplete, and doesn't
>> come with enough description to tell me why it could possibly be complete.
>>
>> I don't think this addresses any of my primary objections from last
>> round.
>>
>> This doesn't change the kernel to make uid comparisons as (uid_ns, uid)
>> tuples or explain why that isn't necessary. It doesn't touch keys.
>> it doesn't explain why we are not introducing possibly subtle security
> problems.
>>
>> Cedric sorry for not saying so earlier, but I thought that the incompleteness
>> was obvious.
>
> (ignoring keys and other uid actions for now)
>
> Here's a stab at semantics for how to handle file access. Should be
> pretty simple to implement, but i won't get a chance to implement this
> week.
>
> At mount, by default the vfsmount is tagged with a uid_ns.
> A new -o uid_ns=<pid> option instead tags the vfsmount with the uid_ns
> belonging to pid <pid>. Since any process in a descendent pid
> namespace should still have a valid pid in the ancestor
> pidspaces, this should work fine.
> At vfs_permission, if current->nsproxy->uid_ns != file->f_vfsmnt->uid_ns,
> 1. If file is owned by root, then read permission is granted
> 2. If file is owned by non-root, no permission is granted
> (regardless of process uid)
>
> Does this sound reasonable?

Possibly, special casing the owner like that is odd.

I would cap the permission granted with at most the other permissions.
The permissions you have described would currently give me the ability
to read /etc/shadow.

Heck I would be happy initially to say
if current->nsproxy->uid_ns != file->f_vfsmnt->uid_ns, then you get no
permissions at all. As a first pass.

> I assume the list of other things we'll need to consider includes

> signals between user namespaces
> keystore
> sys_setpriority and the like
> I might argue that all of these should be sufficiently protected
> by proper setup by userspace. Can you explain why that is not
> the case?

The keystore is simple to explain. It is global ids. So we need
a separate set of global ids, so one container does not conflict
with another.

Signals between user namespaces are also simple to explain. If I can
see your process in my ps listing, I don't have CAP_KILL, and I'm not
you. I shouldn't be able to terminate your processes. See check_kill_permission.

At this point I don't see how proper namespace can protect you in either
of the above two cases.

As long as it requires root privileges to create a new user namespace
I do agree (under the principle of allowing people to shoot themselves
in the foot) that we can require the user to setup properly.

The only other rule we need to look towards for a minimal
implementation is that you can always add more capability but
you can't take something away once someone has come to depend on it.
Just look at sys_sysctl. It only has one user and we can't remove it.

So as long as what the user namespace allows is initially very
draconian. (When in doubt say no). We can probably start with
something fairly simple.

Eric

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

Subject: Re: [RFC][PATCH 0/2] user namespace [try #2]
Posted by Cedric Le Goater on Fri, 01 Sep 2006 08:32:22 GMT

View Forum Message <> Reply to Message

Eric W. Biederman wrote:

[...]

> Cedric sorry for not saying so earlier, but I thought that the incompleteness
> was obvious.

No worries :) But let's see how obvious is that incompleteness before we
add more work to it.

thanks,

C.

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

Subject: Re: [RFC][PATCH 0/2] user namespace [try #2]
Posted by ebiederm on Fri, 01 Sep 2006 09:33:52 GMT

View Forum Message <> Reply to Message

Cedric Le Goater <clg@fr.ibm.com> writes:

> Eric W. Biederman wrote:
>
> [...]
>
>> Cedric sorry for not saying so earlier, but I thought that the incompleteness
>> was obvious.
>
> No worries :) But let's see how obvious is that incompleteness before we
> add more work to it.

Sorry I don't understand what you mean.
Are you suggesting not fixing bugs because everyone cannot see them?

Eric

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

Subject: Re: [RFC][PATCH 0/2] user namespace [try #2]
Posted by Cedric Le Goater on Fri, 01 Sep 2006 13:03:00 GMT

View Forum Message <> Reply to Message

Eric W. Biederman wrote:
> Cedric Le Goater <clg@fr.ibm.com> writes:
>
>> Eric W. Biederman wrote:

>>
>> [...]
>>
>>> Cedric sorry for not saying so earlier, but I thought that the incompleteness
>>> was obvious.
>> No worries :) But let's see how obvious is that incompleteness before we
>> add more work to it.
>
> Sorry I don't understand what you mean.
> Are you suggesting not fixing bugs because everyone cannot see them?

I'm only suggesting to wait for the opinion of the vserver and openvz guys.

If it's already useful for someone, good. if not, let's work on it when we
have time. if they are bugs, let's fix them.

just being pragmatic on what to do because I'd like to spend more time on
the process namespace which really is a must have to do anything else.

cheers,

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

Subject: Re: [RFC][PATCH 0/2] user namespace [try #2]
Posted by ebiederm on Fri, 01 Sep 2006 18:22:50 GMT
View Forum Message <> Reply to Message

Cedric Le Goater <clg@fr.ibm.com> writes:

> Eric W. Biederman wrote:
>> Cedric Le Goater <clg@fr.ibm.com> writes:
>>
>>> Eric W. Biederman wrote:
>>>
>>> [...]
>>>
>>>> Cedric sorry for not saying so earlier, but I thought that the
> incompleteness
>>>> was obvious.
>>> No worries :) But let's see how obvious is that incompleteness before we
>>> add more work to it.
>>
>> Sorry I don't understand what you mean.

>> Are you suggesting not fixing bugs because everyone cannot see them?
>
> I'm only suggesting to wait for the opinion of the vserver and openvz guys.
>
> If it's already useful for someone, good. if not, let's work on it when we
> have time. if they are bugs, let's fix them.
>
> just being pragmatic on what to do because I'd like to spend more time on
> the process namespace which really is a must have to do anything else.

As an ordering suggest that sounds fine. That is the other reason I barely looked
at it. :(

Eric

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

Subject: Re: [RFC][PATCH 0/2] user namespace [try #2]
Posted by Herbert Poetzl on Mon, 04 Sep 2006 20:40:02 GMT

View Forum Message <> Reply to Message

On Fri, Sep 01, 2006 at 03:03:00PM +0200, Cedric Le Goater wrote:
> Eric W. Biederman wrote:
> > Cedric Le Goater <clg@fr.ibm.com> writes:
> >
> >> Eric W. Biederman wrote:
> >>
> >> [...]
> >>
> >>> Cedric sorry for not saying so earlier, but I thought that the
> >>> incompleteness was obvious.
> >> No worries :) But let's see how obvious is that incompleteness
> >> before we add more work to it.
> >
> > Sorry I don't understand what you mean.
> > Are you suggesting not fixing bugs because everyone cannot see them?
>
> I'm only suggesting to wait for the opinion of the vserver and openvz
> guys.

didn't get the patches, I'm now in the process of
looking them up in the mailing list archives ...

so I will comment on them shortly ...

> If it's already useful for someone, good. if not, let's work on it
> when we have time. if they are bugs, let's fix them.
>
> just being pragmatic on what to do because I'd like to spend more time
> on the process namespace which really is a must have to do anything
> else.

best,
Herbert

> cheers,
>
> C.
> ____
> 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

Subject: Re: [RFC][PATCH 0/2] user namespace [try #2]
Posted by ebiederm on Thu, 07 Sep 2006 19:23:12 GMT

View Forum Message <> Reply to Message

Herbert Poetzl <herbert@13thfloor.at> writes:

> On Thu, Sep 07, 2006 at 12:18:14PM -0600, Eric W. Biederman wrote:
>> Kirill Korotaev <dev@sw.ru> writes:
>>
>> > yes, these patches are usable for OpenVZ AS IS, so I'm not sure
>> > why we can't do step by step and commit. However I posted some comments on
>> > patches...
>> >
>> > Eric do you have some STRONG objections (maybe I just missed it somewhere)?
>>
>> - We do not handle interactions between processes in different uid
>>   namespaces and still have the normal uid equality checks.
>> - I am willing to be convinced that this is a nuclear missile the user
>>   is allowed to shoot themselves in the foot with if someone can show me
>>   how to use the current version safely.
>>
>> A lot of this scares me silly as when ever you touch the primary
>> identifier in the security checks you must be very very very careful.
>> My gut feeling is that I'm nowhere near paranoid enough and the rest
>> of you aren't even paranoid.

>>
>> What I want to see is that every uid identity check becomes either
>> a struct user comparison or a uid, uid_ns tuple comparison.
>
> second that!

In addition I don't have problems with incremental progress
if we implement in such a way that we don't enable the ability
to create a new uid namespace to user space before we are certain
it is safe.

All of the code could be present and we just have a one line check
that denied requests to create a new namespace.

Eric

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

Subject: Re: [RFC][PATCH 0/2] user namespace [try #2]
Posted by Cedric Le Goater on Mon, 11 Sep 2006 08:09:22 GMT

View Forum Message <> Reply to Message

Eric W. Biederman wrote:
> Herbert Poetzl <herbert@13thfloor.at> writes:
>
>> On Thu, Sep 07, 2006 at 12:18:14PM -0600, Eric W. Biederman wrote:
>>> Kirill Korotaev <dev@sw.ru> writes:
>>>
>>>> yes, these patches are usable for OpenVZ AS IS, so I'm not sure
>>>> why we can't do step by step and commit. However I posted some comments on
>>>> patches...
>>>>
>>>> Eric do you have some STRONG objections (maybe I just missed it somewhere)?
>>> - We do not handle interactions between processes in different uid
>>>   namespaces and still have the normal uid equality checks.
>>> - I am willing to be convinced that this is a nuclear missile the user
>>>   is allowed to shoot themselves in the foot with if someone can show me
>>>   how to use the current version safely.
>>>
>>> A lot of this scares me silly as when ever you touch the primary
>>> identifier in the security checks you must be very very very careful.
>>> My gut feeling is that I'm nowhere near paranoid enough and the rest
>>> of you aren't even paranoid.
>>>
>>> What I want to see is that every uid identity check becomes either

>>> a struct user comparison or a uid, uid_ns tuple comparison.
>> second that!
>
> In addition I don't have problems with incremental progress
> if we implement in such a way that we don't enable the ability
> to create a new uid namespace to user space before we are certain
> it is safe.
>
> All of the code could be present and we just have a one line check
> that denied requests to create a new namespace.

OK. I'll see how this is possible. I guess the simplest way for the moment
is to remove the unshare() of the user_namespace.

So, shall we follow the 'grep' method for uids like we are doing for pids
and thread ? This is going to be painful but I guess there is no simple
solution ...

C.

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

Subject: Re: [RFC][PATCH 0/2] user namespace [try #2]
Posted by ebiederm on Mon, 11 Sep 2006 11:48:25 GMT

View Forum Message <> Reply to Message

Cedric Le Goater <clg@fr.ibm.com> writes:

> Eric W. Biederman wrote:
>> Herbert Poetzl <herbert@13thfloor.at> writes:
>>
>>
>> In addition I don't have problems with incremental progress
>> if we implement in such a way that we don't enable the ability
>> to create a new uid namespace to user space before we are certain
>> it is safe.
>>
>> All of the code could be present and we just have a one line check
>> that denied requests to create a new namespace.
>
> OK. I'll see how this is possible. I guess the simplest way for the moment
> is to remove the unshare() of the user_namespace.

That is largely what I was thinking. Possibly even leaving the code
there but denying all requests with the CLONE_NEWUSERNS bit set.

> So, shall we follow the 'grep' method for uids like we are doing for pids
> and thread ? This is going to be painful but I guess there is no simple
> solution ...

I can't think of a better one. Although hopefully since security
is involved those checks are in a little better shape, and a little
less distributed throughout the kernel.

Eric

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

Subject: Re: [RFC][PATCH 0/2] user namespace [try #2]
Posted by Dave Hansen on Mon, 11 Sep 2006 16:57:21 GMT

View Forum Message <> Reply to Message

On Mon, 2006-09-11 at 10:09 +0200, Cedric Le Goater wrote:
>
>
> So, shall we follow the 'grep' method for uids like we are doing for
> pids
> and thread ? This is going to be painful but I guess there is no
> simple
> solution ...

```
struct foo
{
 union
 {
  __deprecated uid_t uid;
  uid_t new_uid;
 };
}
```

It will spew warnings for all uses of foo->uid, but still compile.

-- Dave

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