Subject: Re: [patch 0/8] unprivileged mount syscall
Posted by akpm on Fri, 06 Apr 2007 23:02:38 GMT
View Forum Message <> Reply to Message

On Wed, 04 Apr 2007 20:30:12 +0200 Miklos Szeredi <miklos@szeredi.hu> wrote:

> This patchset adds support for keeping mount ownership information in
> the kernel, and allow unprivileged mount(2) and umount(2) in certain
> cases.

No replies, huh?

My knowledge of the code which you're touching is not strong, and my spare reviewing capacity is not high. And this work does need close review by people who are familar with the code which you're changing.

So could I suggest that you go for a dig through the git history, identify some individuals who look like they know this code, then do a resend, cc'ing those people? Please also cc linux-kernel on that resend.

> This can be useful for the following reasons:
>
> - mount(8) can store ownership ("user=XY" option) in the kernel
> instead, or in addition to storing it in /etc/mtab. For example if
> private namespaces are used with mount propagations /etc/mtab
> becomes unworkable, but using /proc/mounts works fine
>
> - fuse won't need a special suid-root mount/umount utility. Plain
> umount(8) can easily be made to work with unprivileged fuse mounts
>
> - users can use bind mounts without having to pre-configure them in
> /etc/fstab
>
> All this is done in a secure way, and unprivileged bind and fuse
> mounts are disabled by default and can be enabled through sysctl or
> /proc/sys.
>
> One thing that is missing from this series is the ability to restrict
> user mounts to private namespaces. The reason is that private
> namespaces have still not gained the momentum and support needed for
> painless user experience. So such a feature would not yet get enough
> attention and testing. However adding such an optional restriction
> can be done with minimal changes in the future, once private
> namespaces have matured.

I suspect the people who developed and maintain nsproxy would disagree ;)

Please also cc containers@lists.osdl.org.

> An earlier version of these patches have been discussed here:
>
> http://lkml.org/lkml/2005/5/3/64
>
> --

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

---

Subject: Re: [patch 0/8] unprivileged mount syscall
Posted by hpa on Fri, 06 Apr 2007 23:16:36 GMT
View Forum Message <> Reply to Message

>>
>> - users can use bind mounts without having to pre-configure them in
>> /etc/fstab
>>

This is by far the biggest concern I see. I think the security
implication of allowing anyone to do bind mounts are poorly understood.

 -hpa

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

---

Subject: Re: [patch 0/8] unprivileged mount syscall
Posted by Jan Engelhardt on Fri, 06 Apr 2007 23:55:44 GMT
View Forum Message <> Reply to Message

On Apr 6 2007 16:16, H. Peter Anvin wrote:
>> >
>> > - users can use bind mounts without having to pre-configure them in
>> > /etc/fstab
>> >
>
> This is by far the biggest concern I see. I think the security implication of
> allowing anyone to do bind mounts are poorly understood.

$ whoami
miklos
$ mount --bind / ~/down_under

later that day:
userdel -r miklos

So both the source (/) and target (~/down_under) directory must be owned
by the user before --bind may succeed.

There may be other implications hpa might want to fill us in.

Regards,
Jan
--

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

---

Subject: Re: [patch 0/8] unprivileged mount syscall
Posted by Miklos Szeredi on Sat, 07 Apr 2007 06:41:48 GMT
View Forum Message <> Reply to Message

> > This patchset adds support for keeping mount ownership information in
> > the kernel, and allow unprivileged mount(2) and umount(2) in certain
> > cases.
>
> No replies, huh?

All we need is a comment from Andrew, and the replies come flooding in ;)

> My knowledge of the code which you're touching is not strong, and my spare
> reviewing capacity is not high. And this work does need close review by
> people who are familar with the code which you're changing.
>
> So could I suggest that you go for a dig through the git history, identify
> some individuals who look like they know this code, then do a resend,
> cc'ing those people? Please also cc linux-kernel on that resend.

OK.

> > One thing that is missing from this series is the ability to restrict
> > user mounts to private namespaces. The reason is that private
> > namespaces have still not gained the momentum and support needed for
> > painless user experience. So such a feature would not yet get enough
> > attention and testing. However adding such an optional restriction
> > can be done with minimal changes in the future, once private
> > namespaces have matured.
>

> I suspect the people who developed and maintain nsproxy would disagree ;)

Well, they better show me some working and simple-to-use userspace
code, because I've not seen anything like that related to mount
namespaces.

pam_namespace.so is one example of a non-working, but probably-not-too-
hard-to-fix one.

I'm just saying this is not yet something that Joe Blow would just
enable by ticking a box in their desktop setup wizard, and it would
all work flawlessly thereafter. There's still a _long_ way towards
that, and mostly in userspace.

Thanks,
Miklos

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

---

Subject: Re: [patch 0/8] unprivileged mount syscall
Posted by serue on Mon, 09 Apr 2007 14:38:02 GMT
View Forum Message <> Reply to Message

Quoting Miklos Szeredi (miklos@szeredi.hu):
> > > This patchset adds support for keeping mount ownership information in
> > > the kernel, and allow unprivileged mount(2) and umount(2) in certain
> > > cases.
> >
> > No replies, huh?
>
> All we need is a comment from Andrew, and the replies come flooding in ;)
>
> > My knowledge of the code which you're touching is not strong, and my spare
> > reviewing capacity is not high. And this work does need close review by
> > people who are familar with the code which you're changing.
> >
> > So could I suggest that you go for a dig through the git history, identify
> > some individuals who look like they know this code, then do a resend,
> > cc'ing those people? Please also cc linux-kernel on that resend.
>
> OK.
>
> > > One thing that is missing from this series is the ability to restrict
> > > user mounts to private namespaces. The reason is that private
> > > namespaces have still not gained the momentum and support needed for

> > > painless user experience. So such a feature would not yet get enough
> > > attention and testing. However adding such an optional restriction
> > > can be done with minimal changes in the future, once private
> > > namespaces have matured.
> >
> > I suspect the people who developed and maintain nsproxy would disagree ;)
>
> Well, they better show me some working and simple-to-use userspace
> code, because I've not seen anything like that related to mount
> namespaces.

If you mean to test/exploit them, see
http://lxc.sourceforge.net/patches/2.6.20/2.6.20-lxc8/broken-out/tests/

Compile the ns_exec.c program and do

 ns_exec -m /bin/sh

to get a shell in a new mounts namespace.

> pam_namespace.so is one example of a non-working, but probably-not-too-
> hard-to-fix one.

Non-working? I sure hope the one used for LSPP certification is
working... As is the ugly version I wrote 18 mounts ago and use on my
laptop.

> I'm just saying this is not yet something that Joe Blow would just
> enable by ticking a box in their desktop setup wizard, and it would
> all work flawlessly thereafter. There's still a _long_ way towards
> that, and mostly in userspace.

I'm not sure there's a that long a way to go, but clearly we need to be
showing users what they can do, or they'll never work their way towards
there.

For instance, as you say, a user admin gui with a checkmark and text
boxes saying 'enter new namespace on login', 'create private /tmp',
and 'create private dmcrypted /home' would be trivial right now.

-serge

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

Subject: Re: [patch 0/8] unprivileged mount syscall
Posted by Miklos Szeredi on Mon, 09 Apr 2007 16:24:10 GMT
View Forum Message <> Reply to Message

> > > > One thing that is missing from this series is the ability to restrict
> > > > user mounts to private namespaces. The reason is that private
> > > > namespaces have still not gained the momentum and support needed for
> > > > painless user experience. So such a feature would not yet get enough
> > > > attention and testing. However adding such an optional restriction
> > > > can be done with minimal changes in the future, once private
> > > > namespaces have matured.
> > >
> > > I suspect the people who developed and maintain nsproxy would disagree ;)
> >
> > Well, they better show me some working and simple-to-use userspace
> > code, because I've not seen anything like that related to mount
> > namespaces.
>
> If you mean to test/exploit them, see
> http://lxc.sourceforge.net/patches/2.6.20/2.6.20-lxc8/broken-out/tests/
>
> Compile the ns_exec.c program and do
>
> ns_exec -m /bin/sh
>
> to get a shell in a new mounts namespace.

Cool, thanks. This is a very nice utility for testing, but for the
end user rather useless:

- user starts up a private namespace in a shell, mounts something

- then opens app from menu, tries to access mount, but the mount is
  not there

- user unhappy

BTW, looking at -mm unshare() on namespace is not privileged any more.
Why is that? Or rather, what's the reason, that clone() is privileged
and unshare() is not?

> > pam_namespace.so is one example of a non-working, but probably-not-too-
> > hard-to-fix one.
>
> Non-working? I sure hope the one used for LSPP certification is
> working... As is the ugly version I wrote 18 mounts ago and use on my
> laptop.

The one in pam-0.99.6.3-29.1 in opensuse-10.2 is totally broken. Are

you interested in the details? I can reproduce it, but forgot to note
down the details of the brokenness.

> > I'm just saying this is not yet something that Joe Blow would just
> > enable by ticking a box in their desktop setup wizard, and it would
> > all work flawlessly thereafter. There's still a _long_ way towards
> > that, and mostly in userspace.
>
> I'm not sure there's a that long a way to go, but clearly we need to be
> showing users what they can do, or they'll never work their way towards
> there.

There _is_ a long way to go. Random things that spring to my mind:

- using /etc/mtab is broken with private namespaces, using
  /proc/mounts is missing various functionality, that /etc/mtab has,
  for example the "user" option, which this patchset adds

- need to set up mount propagation from global namespace to private
  ones, mount(8) does not yet have options to configure propagation

- user namespace setup: what if user has multiple sessions?

  1) namespaces are shared? That's tricky because the session needs to
  be a child of a namespace server, not of login. I'm not sure PAM
  can handle this

  2) or mounts are copied on login? That's not possible currently,
  as there's no way to send a mount between namespaces. Also it's
  tricky to make sure that new mounts are also shared

> For instance, as you say, a user admin gui with a checkmark and text
> boxes saying 'enter new namespace on login', 'create private /tmp',
> and 'create private dmcrypted /home' would be trivial right now.

Trivial modulo the above slightly non-trivial exemptions ;)

Miklos

Subject: Re: [patch 0/8] unprivileged mount syscall
Posted by Ian Kent on Tue, 10 Apr 2007 08:52:05 GMT
View Forum Message <> Reply to Message

On Fri, 2007-04-06 at 16:16 -0700, H. Peter Anvin wrote:
> >>
> >> - users can use bind mounts without having to pre-configure them in
> >>   /etc/fstab
> >>
>
> This is by far the biggest concern I see. I think the security
> implication of allowing anyone to do bind mounts are poorly understood.

And especially so since there is no way for a filesystem module to veto
such requests.

Ian

Subject: Re: [patch 0/8] unprivileged mount syscall
Posted by Miklos Szeredi on Wed, 11 Apr 2007 10:48:34 GMT
View Forum Message <> Reply to Message

> > >>
> > >> - users can use bind mounts without having to pre-configure them in
> > >>   /etc/fstab
> > >>
> >
> > This is by far the biggest concern I see. I think the security
> > implication of allowing anyone to do bind mounts are poorly understood.
>
> And especially so since there is no way for a filesystem module to veto
> such requests.

The filesystem can't veto initial mounts based on destination either.
I don't think it's up to the filesystem to police bind/move mounts in
any way.

Miklos

Subject: Re: [patch 0/8] unprivileged mount syscall
Posted by Ian Kent on Wed, 11 Apr 2007 13:48:30 GMT
View Forum Message <> Reply to Message

On Wed, 2007-04-11 at 12:48 +0200, Miklos Szeredi wrote:
> > > >>
> > > >> - users can use bind mounts without having to pre-configure them in
> > > >>   /etc/fstab
> > > >>
> > > >
> > > > This is by far the biggest concern I see. I think the security
> > > > implication of allowing anyone to do bind mounts are poorly understood.
> > >
> > > And especially so since there is no way for a filesystem module to veto
> > > such requests.
> >
> > The filesystem can't veto initial mounts based on destination either.
> > I don't think it's up to the filesystem to police bind/move mounts in
> > any way.

But if a filesystem can't or the developer thinks that it shouldn't for
some reason, support bind/move mounts then there should be a way for the
filesystem to tell the kernel that.

Surely a filesystem is in a good position to be able to decide if a
mount request "for it" should be allowed to continue based on it's "own
situation and capabilities".

Ian