Subject: [patch 7/9] unprivileged mounts: allow unprivileged fuse mounts
Posted by Miklos Szeredi on Tue, 08 Jan 2008 11:35:09 GMT
View Forum Message <> Reply to Message

From: Miklos Szeredi <mszeredi@suse.cz>

Use FS_SAFE for "fuse" fs type, but not for "fuseblk".

FUSE was designed from the beginning to be safe for unprivileged users. This has also been verified in practice over many years. In addition unprivileged mounts require the parent mount to be owned by the user, which is more strict than the current userspace policy.

This will enable future installations to remove the suid-root fusermount utility.

Don't require the "user_id=" and "group_id=" options for unprivileged mounts, but if they are present, verify them for sanity.

Disallow the "allow_other" option for unprivileged mounts.

Signed-off-by: Miklos Szeredi <mszeredi@suse.cz>

Index: linux/fs/fuse/inode.c
```
===================================================================
--- linux.orig/fs/fuse/inode.c 2008-01-03 17:13:13.000000000 +0100
+++ linux/fs/fuse/inode.c 2008-01-03 21:28:01.000000000 +0100
@@ -357,6 +357,19 @@ static int parse_fuse_opt(char *opt, str
  d->max_read = ~0;
  d->blksize = 512;

+ /*
+  * For unprivileged mounts use current uid/gid.  Still allow
+  * "user_id" and "group_id" options for compatibility, but
+  * only if they match these values.
+  */
+ if (!capable(CAP_SYS_ADMIN)) {
+  d->user_id = current->uid;
+  d->user_id_present = 1;
+  d->group_id = current->gid;
+  d->group_id_present = 1;
+
+ }
+
  while ((p = strsep(&opt, ",")) != NULL) {
   int token;
   int value;
```

```
@@ -385,6 +398,8 @@ static int parse_fuse_opt(char *opt, str
   case OPT_USER_ID:
    if (match_int(&args[0], &value))
     return 0;
+   if (d->user_id_present && d->user_id != value)
+    return 0;
    d->user_id = value;
    d->user_id_present = 1;
    break;
@@ -392,6 +407,8 @@ static int parse_fuse_opt(char *opt, str
   case OPT_GROUP_ID:
    if (match_int(&args[0], &value))
     return 0;
+   if (d->group_id_present && d->group_id != value)
+    return 0;
    d->group_id = value;
    d->group_id_present = 1;
    break;
@@ -596,6 +613,10 @@ static int fuse_fill_super(struct super_
  if (!parse_fuse_opt((char *) data, &d, is_bdev))
   return -EINVAL;

+ /* This is a privileged option */
+ if ((d.flags & FUSE_ALLOW_OTHER) && !capable(CAP_SYS_ADMIN))
+  return -EPERM;
+
  if (is_bdev) {
 #ifdef CONFIG_BLOCK
   if (!sb_set_blocksize(sb, d.blksize))
@@ -696,9 +717,9 @@ static int fuse_get_sb(struct file_syste
 static struct file_system_type fuse_fs_type = {
  .owner  = THIS_MODULE,
  .name  = "fuse",
- .fs_flags = FS_HAS_SUBTYPE,
  .get_sb  = fuse_get_sb,
  .kill_sb = kill_anon_super,
+ .fs_flags = FS_HAS_SUBTYPE | FS_SAFE,
 };

 #ifdef CONFIG_BLOCK

--
```

Subject: Re: [patch 7/9] unprivileged mounts: allow unprivileged fuse mounts
Posted by Pavel Machek on Tue, 08 Jan 2008 21:46:26 GMT
View Forum Message <> Reply to Message

On Tue 2008-01-08 12:35:09, Miklos Szeredi wrote:
> From: Miklos Szeredi <mszeredi@suse.cz>
>
> Use FS_SAFE for "fuse" fs type, but not for "fuseblk".
>
> FUSE was designed from the beginning to be safe for unprivileged users. This
> has also been verified in practice over many years. In addition unprivileged

Eh? So 'kill -9 no longer works' and 'suspend no longer works' is not
considered important enough to even mention?

'updatedb no longer works' is not a problem?

Are you ready to offer shell account for bugtraq people to see how
long it survives?
	Pavel
--
(english) http://www.livejournal.com/~pavelmachek
(cesky, pictures) http://atrey.karlin.mff.cuni.cz/~pavel/picture/horses/blog.html

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

Subject: Re: [patch 7/9] unprivileged mounts: allow unprivileged fuse mounts
Posted by Miklos Szeredi on Tue, 08 Jan 2008 22:42:20 GMT
View Forum Message <> Reply to Message

> On Tue 2008-01-08 12:35:09, Miklos Szeredi wrote:
> > From: Miklos Szeredi <mszeredi@suse.cz>
> >
> > Use FS_SAFE for "fuse" fs type, but not for "fuseblk".
> >
> > FUSE was designed from the beginning to be safe for unprivileged users. This
> > has also been verified in practice over many years. In addition unprivileged
>
> Eh? So 'kill -9 no longer works' and 'suspend no longer works' is not
> considered important enough to even mention?

No. Because in practice they don't seem to matter. Also because
there's no way in which fuse could be done differently to address
these issues.

The 'kill -9' thing is basically due to VFS level locking not being interruptible. It could be changed, but I'm not sure it's worth it.

For the suspend issue, there are also no easy solutions.

> 'updatedb no longer works' is not a problem?

I haven't seen any problems with updatedb, and haven't had any bug reports about it either.

> Are you ready to offer shell account for bugtraq people to see how
> long it survives?

Bugtraq people are free to install fuse on their machines and take it apart.

AFAIR there were two security vulnerabilities in fuse's history, one of them an information leak in the kernel module, and the other one an mtab corruption issue in the fusermount utility. I don't think this is such a bad track record.

Miklos

Subject: Re: [patch 7/9] unprivileged mounts: allow unprivileged fuse mounts
Posted by Pavel Machek on Tue, 08 Jan 2008 22:58:20 GMT
View Forum Message <> Reply to Message

On Tue 2008-01-08 23:42:20, Miklos Szeredi wrote:
> > On Tue 2008-01-08 12:35:09, Miklos Szeredi wrote:
> > > From: Miklos Szeredi <mszeredi@suse.cz>
> > >
> > > Use FS_SAFE for "fuse" fs type, but not for "fuseblk".
> > >
> > > FUSE was designed from the beginning to be safe for unprivileged users. This
> > > has also been verified in practice over many years. In addition unprivileged
> >
> > Eh? So 'kill -9 no longer works' and 'suspend no longer works' is not
> > considered important enough to even mention?
>
> No. Because in practice they don't seem to matter. Also because
> there's no way in which fuse could be done differently to address
> these issues.
>

> The 'kill -9' thing is basically due to VFS level locking not being
> interruptible. It could be changed, but I'm not sure it's worth it.

Well, I believe it should be changed. "You need to mount /sys, then
echo X to Y before kill -9 works" does not look nice... I agree it is
not easy.

> > 'updatedb no longer works' is not a problem?
>
> I haven't seen any problems with updatedb, and haven't had any bug
> reports about it either.

Ok, I don't know much about FUSE. In current version, if user creates
infinite maze and mounts it under ~, updatedb just does not enter it?

> AFAIR there were two security vulnerabilities in fuse's history, one
> of them an information leak in the kernel module, and the other one an
> mtab corruption issue in the fusermount utility. I don't think this
> is such a bad track record.

Not bad indeed. But I'd consider 'kill -9 not working' to be DoS
vulnerability... and I'm woried about problems fuse + user mounts
expose in other parts of system.
        Pavel
--
(english) http://www.livejournal.com/~pavelmachek
(cesky, pictures) http://atrey.karlin.mff.cuni.cz/~pavel/picture/horses/blog.html
_____
Containers mailing list
Containers@lists.linux-foundation.org
https://lists.linux-foundation.org/mailman/listinfo/containers

---

Subject: Re: [patch 7/9] unprivileged mounts: allow unprivileged fuse mounts
Posted by Nigel Cunningham on Tue, 08 Jan 2008 23:56:28 GMT
View Forum Message <> Reply to Message

Hi.

Miklos Szeredi wrote:
>> On Tue 2008-01-08 12:35:09, Miklos Szeredi wrote:
>>> From: Miklos Szeredi <mszeredi@suse.cz>
>>>
>>> Use FS_SAFE for "fuse" fs type, but not for "fuseblk".
>>>
>>> FUSE was designed from the beginning to be safe for unprivileged users. This
>>> has also been verified in practice over many years. In addition unprivileged
>> Eh? So 'kill -9 no longer works' and 'suspend no longer works' is not

>> considered important enough to even mention?
>
> No. Because in practice they don't seem to matter. Also because
> there's no way in which fuse could be done differently to address
> these issues.

Could you clarify, please? I hope I'm getting the wrong end of the stick
- it sounds to me like you and Pavel are saying that this patch breaks
suspending to ram (and hibernating?) but you want to push it anyway
because you haven't been able to produce an instance, don't think
suspending or hibernating matter and couldn't fix fuse anyway?

> The 'kill -9' thing is basically due to VFS level locking not being
> interruptible. It could be changed, but I'm not sure it's worth it.
>
> For the suspend issue, there are also no easy solutions.

What are the non-easy solutions?

Regards,

Nigel

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

---

Subject: Re: [patch 7/9] unprivileged mounts: allow unprivileged fuse mounts
Posted by Miklos Szeredi on Wed, 09 Jan 2008 08:47:31 GMT
View Forum Message <> Reply to Message

> >> On Tue 2008-01-08 12:35:09, Miklos Szeredi wrote:
> >>> From: Miklos Szeredi <mszeredi@suse.cz>
> >>>
> >>> Use FS_SAFE for "fuse" fs type, but not for "fuseblk".
> >>>
> >>> FUSE was designed from the beginning to be safe for unprivileged users. This
> >>> has also been verified in practice over many years. In addition unprivileged
> >> Eh? So 'kill -9 no longer works' and 'suspend no longer works' is not
> >> considered important enough to even mention?
> >
> > No. Because in practice they don't seem to matter. Also because
> > there's no way in which fuse could be done differently to address
> > these issues.
>
> Could you clarify, please? I hope I'm getting the wrong end of the stick
> - it sounds to me like you and Pavel are saying that this patch breaks

> suspending to ram (and hibernating?) but you want to push it anyway
> because you haven't been able to produce an instance, don't think
> suspending or hibernating matter and couldn't fix fuse anyway?

This patch has nothing to do with suspend or hibernate. What this patchset does, is help get rid of fusermount, a suid-root mount helper. It also opens up new possibilities, which are not fuse related.

Fuse has bad interactions with the freezer, theoretically. In practice, I remember just one bug report (that sparked off this whole "do we need freezer, or don't we" flamefest), that actually got fixed fairly quickly, ...maybe. Rafael probably remembers better.

> > The 'kill -9' thing is basically due to VFS level locking not being
> > interruptible. It could be changed, but I'm not sure it's worth it.
> >
> > For the suspend issue, there are also no easy solutions.
>
> What are the non-easy solutions?

The ability to freeze tasks in uninterruptible sleep, or more generally at any preempt point (except when drivers are poking hardware).

I know this doesn't play well with userspace hibernate, and I don't think it can be resolved without going the kexec way.

Miklos

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

---

Subject: Re: [patch 7/9] unprivileged mounts: allow unprivileged fuse mounts
Posted by Miklos Szeredi on Wed, 09 Jan 2008 09:11:06 GMT
View Forum Message <> Reply to Message

> > > 'updatedb no longer works' is not a problem?
> >
> > I haven't seen any problems with updatedb, and haven't had any bug
> > reports about it either.
>
> Ok, I don't know much about FUSE. In current version, if user creates
> infinite maze and mounts it under ~, updatedb just does not enter it?

It doesn't. See Documentation/filesystems/fuse.txt

> > AFAIR there were two security vulnerabilities in fuse's history, one
> > of them an information leak in the kernel module, and the other one an
> > mtab corruption issue in the fusermount utility. I don't think this
> > is such a bad track record.
>
> Not bad indeed. But I'd consider 'kill -9 not working' to be DoS
> vulnerability...

The worst that can happen is that a sysadmin doesn't read the docs
(likely) before enabling fuse on a multiuser system, and is surprised
by a user doing funny things. And _then_ has to go read the docs, or
google for some info. This is basically how things normally work, and
I don't consider it a DoS.

> and I'm woried about problems fuse + user mounts expose in other
> parts of system.

I'm worried too, and I'm not saying that enabling unprivileged fuse
mounts is completely risk free. Nothing is, and nobody is forced to
do it.

Miklos

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

---

Subject: Re: [patch 7/9] unprivileged mounts: allow unprivileged fuse mounts
Posted by Szabolcs Szakacsits on Wed, 09 Jan 2008 09:19:04 GMT
View Forum Message <> Reply to Message

Hi,

On Wed, 9 Jan 2008, Nigel Cunningham wrote:
> On Tue 2008-01-08 12:35:09, Miklos Szeredi wrote:
> >
> > For the suspend issue, there are also no easy solutions.
>
> What are the non-easy solutions?

A practical point of view I've seen only fuse rootfs mounts to be a
problem. I remember Ubuntu patches for this (WUBI and some other distros
install NTFS root). But this probably also depends on the used suspend
implementation.

Personally I've never had fuse related suspend problem with ordinary mounts

during heavy use under development, nor NTFS user problem was tracked down
to it in the last one and half year.

Regards,
    Szaka

--
NTFS-3G: http://ntfs-3g.org

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

---

Subject: Re: [patch 7/9] unprivileged mounts: allow unprivileged fuse mounts
Posted by Nigel Cunningham on Wed, 09 Jan 2008 09:29:24 GMT
View Forum Message <> Reply to Message

Hi.

Miklos Szeredi wrote:
>>>> On Tue 2008-01-08 12:35:09, Miklos Szeredi wrote:
>>>>> From: Miklos Szeredi <mszeredi@suse.cz>
>>>>>
>>>>> Use FS_SAFE for "fuse" fs type, but not for "fuseblk".
>>>>>
>>>>> FUSE was designed from the beginning to be safe for unprivileged users. This
>>>>> has also been verified in practice over many years. In addition unprivileged
>>>> Eh? So 'kill -9 no longer works' and 'suspend no longer works' is not
>>>> considered important enough to even mention?
>>> No. Because in practice they don't seem to matter. Also because
>>> there's no way in which fuse could be done differently to address
>>> these issues.
>> Could you clarify, please? I hope I'm getting the wrong end of the stick
>> - it sounds to me like you and Pavel are saying that this patch breaks
>> suspending to ram (and hibernating?) but you want to push it anyway
>> because you haven't been able to produce an instance, don't think
>> suspending or hibernating matter and couldn't fix fuse anyway?
>
> This patch has nothing to do with suspend or hibernate. What this
> patchset does, is help get rid of fusermount, a suid-root mount
> helper. It also opens up new possibilities, which are not fuse
> related.

That's what I thought. So what was Pavel talking about with "kill -9 no
longer works" and "suspend no longer works" above? I couldn't understand
it from the context.

> Fuse has bad interactions with the freezer, theoretically. In
> practice, I remember just one bug report (that sparked off this whole
> "do we need freezer, or don't we" flamefest), that actually got fixed
> fairly quickly, ...maybe. Rafael probably remembers better.

I think they just gave up and considered it unsolvable. I'm not sure it is.

>>> The 'kill -9' thing is basically due to VFS level locking not being
>>> interruptible. It could be changed, but I'm not sure it's worth it.
>>>
>>> For the suspend issue, there are also no easy solutions.
>> What are the non-easy solutions?
>
> The ability to freeze tasks in uninterruptible sleep, or more
> generally at any preempt point (except when drivers are poking
> hardware).

Couldn't some sort of scheduler based solution deal with the
uninterruptible sleeping case?

> I know this doesn't play well with userspace hibernate, and I don't
> think it can be resolved without going the kexec way.

I can see the desirability of kexec when it comes to avoiding the
freezer, but comes with its own problems too - having the original
context usable is handy, not having to set aside a large amount of space
for a second kernel is also desirable and there are still greater issues
of transferring information backwards and forwards between the two kernels.

Regards,

Nigel

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

---

Subject: Re: [patch 7/9] unprivileged mounts: allow unprivileged fuse mounts
Posted by Pavel Machek on Wed, 09 Jan 2008 11:12:42 GMT
View Forum Message <> Reply to Message

On Wed 2008-01-09 09:47:31, Miklos Szeredi wrote:
> > >> On Tue 2008-01-08 12:35:09, Miklos Szeredi wrote:
> > >>> From: Miklos Szeredi <mszeredi@suse.cz>
> > >>>
> > >>> Use FS_SAFE for "fuse" fs type, but not for "fuseblk".
> > >>>

> > >>> FUSE was designed from the beginning to be safe for unprivileged users. This
> > >>> has also been verified in practice over many years. In addition unprivileged
> > >> Eh? So 'kill -9 no longer works' and 'suspend no longer works' is not
> > >> considered important enough to even mention?
> > >
> > > No. Because in practice they don't seem to matter. Also because
> > > there's no way in which fuse could be done differently to address
> > > these issues.
> >
> > Could you clarify, please? I hope I'm getting the wrong end of the stick
> > - it sounds to me like you and Pavel are saying that this patch breaks
> > suspending to ram (and hibernating?) but you want to push it anyway
> > because you haven't been able to produce an instance, don't think
> > suspending or hibernating matter and couldn't fix fuse anyway?
>
> This patch has nothing to do with suspend or hibernate. What this
> patchset does, is help get rid of fusermount, a suid-root mount
> helper. It also opens up new possibilities, which are not fuse
> related.
>
> Fuse has bad interactions with the freezer, theoretically. In
> practice, I remember just one bug report (that sparked off this whole
> "do we need freezer, or don't we" flamefest), that actually got fixed
> fairly quickly, ...maybe. Rafael probably remembers better.

In practice, if the "unpriviledged fuse" gets enabled, any user can
prevent suspend/hibernation from working.
Pavel
--
(english) http://www.livejournal.com/~pavelmachek
(cesky, pictures) http://atrey.karlin.mff.cuni.cz/~pavel/picture/horses/blog.html

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

---

Subject: Re: [patch 7/9] unprivileged mounts: allow unprivileged fuse mounts
Posted by Pavel Machek on Wed, 09 Jan 2008 11:33:25 GMT
View Forum Message <> Reply to Message

Hi!

> > > AFAIR there were two security vulnerabilities in fuse's history, one
> > > of them an information leak in the kernel module, and the other one an
> > > mtab corruption issue in the fusermount utility. I don't think this
> > > is such a bad track record.
> >

> > Not bad indeed. But I'd consider 'kill -9 not working' to be DoS
> > vulnerability...
>
> The worst that can happen is that a sysadmin doesn't read the docs
> (likely) before enabling fuse on a multiuser system, and is surprised
> by a user doing funny things. And _then_ has to go read the docs, or
> google for some info. This is basically how things normally work, and
> I don't consider it a DoS.

No, this is not normal. Kill -9 has been estabilished long time ago,
and we should not be documenting its now-brokenness in
Documentation/filesystems/fuse.txt .

For example, my /etc/inittab currently has:

```
kb::kbrequest:/etc/rc/rc.reboot 2 0

#
# This file handles system shutdown and reboot.
#

PATH=/sbin:/bin:/usr/sbin:/usr/bin
sync &

# Kill all processes.
wall System is going down NOW\!
echo -n -e "\rSystem is going down: processes."
killall5 -15
echo -n "."
sleep 1
echo -n ". "
killall5 -9

# Before unmounting file systems write a reboot record to wtmp.
echo -n "wtmp "
halt -w

# Swap needs to be unmounted because otherwise busy filesystems
remain.
echo -n "swap "
swapoff -a
swapoff /c/swap
swapoff /c/swap2

# Unmount file systems
echo -n "umount."
umount -a || (
 sync &
```

```
 echo -n "umount-retry."
 sleep 1
 umount -a || sulogin
)
echo -n ". "
mount -n -o remount,ro /

# Now halt or reboot.
if [ "$2" = "0" ] ; then
 swapoff -a
 echo "halted."
 halt -p -f
else
 echo "rebooting..."
 reboot -d -f
fi
```

...this will break with FUSE enabled, right? (Minor security hole by allowing users to stop c-a-delete, where none existed before?)

I'm currently suspending by 'echo "mem" > /sys/power/state'. How should I do that _safely_ with FUSE enabled?

If I want to get rid of nasty user in multiuser system, I do su nastyuser 'kill -9 -1' . How do I do the equivalent with FUSE enabled? (Without affecting other users?)

Load average was never really meaningful number, but with FUSE enabled, users can set it to 666 without actually eating any CPU.

SIGSTOP used to work, allowing you to prevent user processes from working while you examine them. Now SIGSTOP can be delayed for arbitrary time.

Heck, imagine malicious user process misbehaves. Before FUSE, you could at least attach it with gdb to look what it is doing. Now you can't.

I really believe FUSE vs. signals needs fixing. Either that, or updating all the manpages

man 1 kill:

```
-       KILL       9   exit      this signal may not be blocked
+       KILL       9   exit      this signal may not be blocked, except by FUSE user mount
```

Pavel

--

(english) http://www.livejournal.com/~pavelmachek

(cesky, pictures) http://atrey.karlin.mff.cuni.cz/~pavel/picture/horses/blog.html

_____

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

---

Subject: Re: [patch 7/9] unprivileged mounts: allow unprivileged fuse mounts
Posted by Miklos Szeredi on Wed, 09 Jan 2008 13:16:21 GMT
View Forum Message <> Reply to Message

> ...this will break with FUSE enabled, right? (Minor security hole by
> allowing users to stop c-a-delete, where none existed before?)

Yup (or I don't know, I'm sure there was or is some problem with
ptrace, that could be used to create unkillable processes).

Fuse could actually be fixed to exit reliably for 'killall5 -9' (it
used to), but that has other problems, and it doesn't seem very
important to me. But this can be discussed.

What cannot be fixed is if one process is inside an fs operation
(e.g. unlink), holding a VFS lock (i_mutex) and another process goes
to uninterruptible sleep on that lock. There's no way (other than
rewriting the VFS) in which that second process could be killed unless
you kill the first one or the fuse server.

> I'm currently suspending by 'echo "mem" > /sys/power/state'. How
> should I do that _safely_ with FUSE enabled?

You can't. But that's only solvable with

- rewrite of VFS (see above)
- rewrite of freezer

> If I want to get rid of nasty user in multiuser system, I do
> su nastyuser 'kill -9 -1' . How do I do the equivalent with FUSE
> enabled? (Without affecting other users?)

You can still do that. If a process cannot be killed with 'kill -9',
due to being deadlocked with itself through fuse (not an easy feat to
accomplish), then it's not going to do any more harm, and you _can_
get rid of it by forced umounting the filesystem, or if it has been
detached, through the fusectl filesystem.

> Load average was never really meaningful number, but with FUSE
> enabled, users can set it to 666 without actually eating any CPU.
>

> SIGSTOP used to work, allowing you to prevent user processes from
> working while you examine them. Now SIGSTOP can be delayed for
> arbitrary time.

Making filesystem operations restartable is not easy. I would say
near impossible, but I haven't given a lot of enery into investigating.

> Heck, imagine malicious user process misbehaves. Before FUSE, you
> could at least attach it with gdb to look what it is doing. Now you
> can't.

Sure, but you can check in other ways (/proc/$PID/wchan), sysrq-t.

> I really believe FUSE vs. signals needs fixing. Either that, or
> updating all the manpages
>
> man 1 kill:
> - KILL 9 exit this signal may not be blocked
> + KILL 9 exit this signal may not be blocked, except by FUSE user mount

Heh, there are all very interesting, but most of these issues are not
even on my todo list (which has grown into quite a big pile over the
years), which means, that they don't seem to matter to people in
practice.

You seem to be implying that fuse is worthless if these issues are not
fixed, but that is very far from the truth, I think.

Miklos

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

---

Subject: Re: [patch 7/9] unprivileged mounts: allow unprivileged fuse mounts
Posted by Pavel Machek on Wed, 09 Jan 2008 13:35:06 GMT
View Forum Message <> Reply to Message

Hi!

> > ...this will break with FUSE enabled, right? (Minor security hole by
> > allowing users to stop c-a-delete, where none existed before?)
>
> Yup (or I don't know, I'm sure there was or is some problem with
> ptrace, that could be used to create unkillable processes).
>
> Fuse could actually be fixed to exit reliably for 'killall5 -9' (it

> used to), but that has other problems, and it doesn't seem very
> important to me. But this can be discussed.

I think it is better to fix fuse than to rewrite all the shutdown scripts.

> What cannot be fixed is if one process is inside an fs operation
> (e.g. unlink), holding a VFS lock (i_mutex) and another process goes
> to uninterruptible sleep on that lock. There's no way (other than
> rewriting the VFS) in which that second process could be killed unless
> you kill the first one or the fuse server.

I believe VFS should be rewritten here. Perhaps new "TASK_KILLABLE"
state can help?

> > I really believe FUSE vs. signals needs fixing. Either that, or
> > updating all the manpages
> >
> > man 1 kill:
> > -   KILL   9   exit   this signal may not be blocked
> > +   KILL   9   exit   this signal may not be blocked, except by FUSE user mount
>
> Heh, there are all very interesting, but most of these issues are not
> even on my todo list (which has grown into quite a big pile over the
> years), which means, that they don't seem to matter to people in
> practice.
>
> You seem to be implying that fuse is worthless if these issues are not
> fixed, but that is very far from the truth, I think.

I'm not saying fuse is worthless. It is a nice toy for single-user
systems. But I do not think we should be merging "allow ordinary users
to mount their own fuse's" before issues above are fixed.
        Pavel
--
(english) http://www.livejournal.com/~pavelmachek
(cesky, pictures) http://atrey.karlin.mff.cuni.cz/~pavel/picture/horses/blog.html

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

---

Subject: Re: [patch 7/9] unprivileged mounts: allow unprivileged fuse mounts
Posted by Miklos Szeredi on Wed, 09 Jan 2008 13:48:49 GMT
View Forum Message <> Reply to Message

> I'm not saying fuse is worthless. It is a nice toy for single-user
> systems. But I do not think we should be merging "allow ordinary users

> to mount their own fuse's" before issues above are fixed.

I think multi user systems are not all that interesting. And I
suspect very few of them want reliably working suspend/hibernate
(which they wouldn't get due to other issues anyway), or have weird
shutdown scripts which stop when they are unable to umount
filesystems.

For paranoid sysadmins, I suggest not enabling fuse for unprivileged
users, which is pretty easy to do: just don't set /dev/fuse to be
world read-writable (which is the default BTW).

So your reasons just don't warrant a big effort involving VFS hacking,
etc. Patches are of course welcome.

Miklos

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

---

Subject: Re: [patch 7/9] unprivileged mounts: allow unprivileged fuse mounts
Posted by Pavel Machek on Wed, 09 Jan 2008 14:00:28 GMT
View Forum Message <> Reply to Message

On Wed 2008-01-09 14:48:53, Miklos Szeredi wrote:
> > I'm not saying fuse is worthless. It is a nice toy for single-user
> > systems. But I do not think we should be merging "allow ordinary users
> > to mount their own fuse's" before issues above are fixed.
>
> I think multi user systems are not all that interesting. And I
> suspect very few of them want reliably working suspend/hibernate
> (which they wouldn't get due to other issues anyway), or have weird
> shutdown scripts which stop when they are unable to umount
> filesystems.

Weird shutdown scripts? I believe all shutdown scripts have this issue
-- if you want to [cleanly] unmount your / filesystem, you need all
the opens for write closed, right...? Which self-deadlocked fused
holding files open will prevent.

> For paranoid sysadmins, I suggest not enabling fuse for unprivileged
> users, which is pretty easy to do: just don't set /dev/fuse to be
> world read-writable (which is the default BTW).
>
> So your reasons just don't warrant a big effort involving VFS hacking,
> etc. Patches are of course welcome.

Well, I believe code with obscure, but almost impossible to fix
problems should not be merged... because that means it will not be
fixed and we'll just have to live with broken kill -9 forever.

Anyway, I believe it would be fair to mention kill -9 no longer
working and shutdown/hibernation/multiuser problems it implies in the
changelogs and probably sysctl documentation or how is this enabled.
Pavel
--
(english) http://www.livejournal.com/~pavelmachek
(cesky, pictures) http://atrey.karlin.mff.cuni.cz/~pavel/picture/horses/blog.html

_____

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

---

Subject: Re: [patch 7/9] unprivileged mounts: allow unprivileged fuse mounts
Posted by Miklos Szeredi on Wed, 09 Jan 2008 14:14:13 GMT
View Forum Message <> Reply to Message

> > > I'm not saying fuse is worthless. It is a nice toy for single-user
> > > systems. But I do not think we should be merging "allow ordinary users
> > > to mount their own fuse's" before issues above are fixed.
> >
> > I think multi user systems are not all that interesting. And I
> > suspect very few of them want reliably working suspend/hibernate
> > (which they wouldn't get due to other issues anyway), or have weird
> > shutdown scripts which stop when they are unable to umount
> > filesystems.
>
> Weird shutdown scripts? I believe all shutdown scripts have this issue
> -- if you want to [cleanly] unmount your / filesystem, you need all
> the opens for write closed, right...? Which self-deadlocked fused
> holding files open will prevent.
>
> > For paranoid sysadmins, I suggest not enabling fuse for unprivileged
> > users, which is pretty easy to do: just don't set /dev/fuse to be
> > world read-writable (which is the default BTW).
> >
> > So your reasons just don't warrant a big effort involving VFS hacking,
> > etc. Patches are of course welcome.
>
> Well, I believe code with obscure, but almost impossible to fix
> problems should not be merged...

That code _has_ been merged, something like 3 years ago, and is doing

fine, thank you.

The unprivileged mounts code, which we should be discussing, doesn't
change anything about that, except to not require another suid-root
utility. Many distributions enabling unprivileged mounting by default
now, so it's not as if there's some great danger in doing this
slightly differently.

> Anyway, I believe it would be fair to mention kill -9 no longer
> working and shutdown/hibernation/multiuser problems it implies in the
> changelogs and probably sysctl documentation or how is this enabled.

Sure.

Miklos

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

---

Subject: Re: [patch 7/9] unprivileged mounts: allow unprivileged fuse mounts
Posted by serue on Mon, 14 Jan 2008 23:24:19 GMT
View Forum Message <> Reply to Message

Quoting Miklos Szeredi (miklos@szeredi.hu):
> From: Miklos Szeredi <mszeredi@suse.cz>
>
> Use FS_SAFE for "fuse" fs type, but not for "fuseblk".
>
> FUSE was designed from the beginning to be safe for unprivileged users. This
> has also been verified in practice over many years. In addition unprivileged
> mounts require the parent mount to be owned by the user, which is more strict
> than the current userspace policy.
>
> This will enable future installations to remove the suid-root fusermount
> utility.
>
> Don't require the "user_id=" and "group_id=" options for unprivileged mounts,
> but if they are present, verify them for sanity.
>
> Disallow the "allow_other" option for unprivileged mounts.
>
> Signed-off-by: Miklos Szeredi <mszeredi@suse.cz>

Sounds like a sysctl to enable FS_SAFE for fuse will make this patch
acceptable to everyone?

```
> ---
>
> Index: linux/fs/fuse/inode.c
> ===================================================================
> --- linux.orig/fs/fuse/inode.c 2008-01-03 17:13:13.000000000 +0100
> +++ linux/fs/fuse/inode.c 2008-01-03 21:28:01.000000000 +0100
> @@ -357,6 +357,19 @@ static int parse_fuse_opt(char *opt, str
>   d->max_read = ~0;
>   d->blksize = 512;
>
> + /*
> +  * For unprivileged mounts use current uid/gid.  Still allow
> +  * "user_id" and "group_id" options for compatibility, but
> +  * only if they match these values.
> +  */
> + if (!capable(CAP_SYS_ADMIN)) {
> +  d->user_id = current->uid;
> +  d->user_id_present = 1;
> +  d->group_id = current->gid;
> +  d->group_id_present = 1;
> +
> + }
> +
>   while ((p = strsep(&opt, ",")) != NULL) {
>    int token;
>    int value;
> @@ -385,6 +398,8 @@ static int parse_fuse_opt(char *opt, str
>    case OPT_USER_ID:
>     if (match_int(&args[0], &value))
>      return 0;
> +   if (d->user_id_present && d->user_id != value)
> +    return 0;
>     d->user_id = value;
>     d->user_id_present = 1;
>     break;
> @@ -392,6 +407,8 @@ static int parse_fuse_opt(char *opt, str
>    case OPT_GROUP_ID:
>     if (match_int(&args[0], &value))
>      return 0;
> +   if (d->group_id_present && d->group_id != value)
> +    return 0;
>     d->group_id = value;
>     d->group_id_present = 1;
>     break;
> @@ -596,6 +613,10 @@ static int fuse_fill_super(struct super_
>   if (!parse_fuse_opt((char *) data, &d, is_bdev))
>    return -EINVAL;
>
```

> + /* This is a privileged option */
> + if ((d.flags & FUSE_ALLOW_OTHER) && !capable(CAP_SYS_ADMIN))
> + return -EPERM;
> +
>   if (is_bdev) {
> #ifdef CONFIG_BLOCK
>   if (!sb_set_blocksize(sb, d.blksize))
> @@ -696,9 +717,9 @@ static int fuse_get_sb(struct file_syste
> static struct file_system_type fuse_fs_type = {
>   .owner = THIS_MODULE,
>   .name = "fuse",
> - .fs_flags = FS_HAS_SUBTYPE,
>   .get_sb = fuse_get_sb,
>   .kill_sb = kill_anon_super,
> + .fs_flags = FS_HAS_SUBTYPE | FS_SAFE,
> };
>
> #ifdef CONFIG_BLOCK
>
> --

Subject: Re: [patch 7/9] unprivileged mounts: allow unprivileged fuse mounts
Posted by Miklos Szeredi on Tue, 15 Jan 2008 10:29:57 GMT
View Forum Message <> Reply to Message

> Sounds like a sysctl to enable FS_SAFE for fuse will make this patch
> acceptable to everyone?

I think the most generic approach, is to be able to set "safeness" for
any fs type, not just fuse (Karel's suggestion).

E.g:

  echo 1 > /proc/sys/fs/types/cifs/safe

This would also provide a way to query the FS_SAFE flag.

Miklos

Subject: Re: [patch 7/9] unprivileged mounts: allow unprivileged fuse mounts
Posted by serue on Tue, 15 Jan 2008 13:35:04 GMT
View Forum Message <> Reply to Message

Quoting Miklos Szeredi (miklos@szeredi.hu):
> > Sounds like a sysctl to enable FS_SAFE for fuse will make this patch
> > acceptable to everyone?
>
> I think the most generic approach, is to be able to set "safeness" for
> any fs type, not just fuse (Karel's suggestion).
>
> E.g:
>
> echo 1 > /proc/sys/fs/types/cifs/safe
>
> This would also provide a way to query the FS_SAFE flag.

That sounds good.

-serge

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