Subject: Re: [PATCH 7/8] user ns: handle file sigio
Posted by serue on Fri, 26 Jan 2007 05:38:08 GMT
View Forum Message <> Reply to Message

Quoting Serge E. Hallyn (serue@us.ibm.com):
> Quoting Andrew Morton (akpm@osdl.org):
> > On Wed, 24 Jan 2007 12:58:45 -0600
> > "Serge E. Hallyn" <serue@us.ibm.com> wrote:
> >
> > > > If we need to I can see doing something special if the process setting
> > > > fown has CAP_KILL
> > >
> > > Obviously CAP_KILL is insufficient :) I assume you mean a new
> > > CAP_XNS_CAP_KILL?
> > >
> > > > and bypassing the security checks that way, but
> > > > hard coding rules like that when it doesn't appear we have any
> > > > experience to indicate we need the extra functionality looks
> > > > premature.
> > >
> > > Ok, in this case actually I suspect you're right and we can just ditch
> > > the exception. But in general the security discussion is one we should
> > > still have.
> >
> > People like security.
> >
> > Where do we now stand with this patch, and with "[PATCH 4/8] user ns: hook permission"?
>
> Later today I can send a patch against this set which removes the
> the init_task exceptions (out of patch 3 and patch 7), but I'd prefer
> to leave the MS_SHARED_NS option (patch 6) in.
>
> thanks,
> -serge

Boots with USER_NS=n (given Cedric's patch to fix that original problem)
and passes my testcases with USER_NS=y.

From: Serge E. Hallyn <serue@us.ibm.com>
Subject: [PATCH] user namespace: remove exceptions for initial namespace

Both sigio and file access checks for user namespace equivalence
were being skipped for processes in the initial namespace.
Remove these exceptions, enforcing the same cross-namespace
checks for all processes in all user namespaces.

Signed-off-by: Serge E. Hallyn <serue@us.ibm.com>

```
---

 fs/fcntl.c           |    3 +--
 include/linux/sched.h |    4 +---
 2 files changed, 2 insertions(+), 5 deletions(-)

939c4da5209a2c00aca70048915007d0eef8ad75
diff --git a/fs/fcntl.c b/fs/fcntl.c
index 6a774c1..d7113d5 100644
--- a/fs/fcntl.c
+++ b/fs/fcntl.c
@@ -460,8 +460,7 @@ static const long band_table[NSIGPOLL] =
 static inline int sigio_perm(struct task_struct *p,
                              struct fown_struct *fown, int sig)
 {
- if (fown->user_ns != init_task.nsproxy->user_ns &&
-     fown->user_ns != p->nsproxy->user_ns)
+ if (fown->user_ns != p->nsproxy->user_ns)
    return 0;
   return (((fown->euid == 0) ||
     (fown->euid == p->suid) || (fown->euid == p->uid) ||
diff --git a/include/linux/sched.h b/include/linux/sched.h
index edbdce2..5c3438b 100644
--- a/include/linux/sched.h
+++ b/include/linux/sched.h
@@ -1614,12 +1614,10 @@ extern int cond_resched_softirq(void);
 static inline int task_mnt_same_uidns(struct task_struct *tsk,
       struct vfsmount *mnt)
 {
- if (tsk->nsproxy == init_task.nsproxy)
+ if (mnt->mnt_user_ns == tsk->nsproxy->user_ns)
    return 1;
    if (mnt->mnt_flags & MNT_SHARE_NS)
     return 1;
-- if (mnt->mnt_user_ns == tsk->nsproxy->user_ns)
--  return 1;
   return 0;
 }
 #else
--
1.1.6
_____________________________________________________
Containers mailing list
Containers@lists.osdl.org
https://lists.osdl.org/mailman/listinfo/containers
```