
Subject: [patch 05/10] unprivileged mounts: allow unprivileged bind mounts
Posted by [Miklos Szeredi](#) on Wed, 16 Jan 2008 12:31:52 GMT
[View Forum Message](#) <> [Reply to Message](#)

From: Miklos Szeredi <mszeredi@suse.cz>

Allow bind mounts to unprivileged users if the following conditions are met:

- mountpoint is not a symlink
- parent mount is owned by the user
- the number of user mounts is below the maximum

Unprivileged mounts imply MS_SETUSER, and will also have the "nosuid" and "nodev" mount flags set.

In particular, if mounting process doesn't have CAP_SETUID capability, then the "nosuid" flag will be added, and if it doesn't have CAP_MKNOD capability, then the "nodev" flag will be added.

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

Acked-by: Serge Hallyn <serue@us.ibm.com>

Index: linux/fs/namespace.c

```
=====
--- linux.orig/fs/namespace.c 2008-01-16 13:25:07.000000000 +0100
+++ linux/fs/namespace.c 2008-01-16 13:25:08.000000000 +0100
@@ -511,6 +511,11 @@ static void __set_mnt_user(struct vfsmou
    WARN_ON(mnt->mnt_flags & MNT_USER);
    mnt->mnt_uid = current->fsuid;
    mnt->mnt_flags |= MNT_USER;
+
+ if (!capable(CAP_SETUID))
+   mnt->mnt_flags |= MNT_NOSUID;
+ if (!capable(CAP_MKNOD))
+   mnt->mnt_flags |= MNT_NODEV;
+ }

static void set_mnt_user(struct vfsmount *mnt)
@@ -1021,22 +1026,26 @@ asmlinkage long sys_oldumount(char __use

#endif

-static int mount_is_safe(struct nameidata *nd)
+/*
+ * Conditions for unprivileged mounts are:
+ * - mountpoint is not a symlink
+ * - mountpoint is in a mount owned by the user

```

```

+ */
+static bool permit_mount(struct nameidata *nd, int *flags)
{
+ struct inode *inode = nd->path.dentry->d_inode;
+
+ if (capable(CAP_SYS_ADMIN))
- return 0;
- return -EPERM;
-#ifdef notyet
- if (S_ISLNK(nd->path.dentry->d_inode->i_mode))
- return -EPERM;
- if (nd->path.dentry->d_inode->i_mode & S_ISVTX) {
- if (current->uid != nd->path.dentry->d_inode->i_uid)
- return -EPERM;
- }
- if (vfs_permission(nd, MAY_WRITE))
- return -EPERM;
- return 0;
-#endif
+ return true;
+
+ if (S_ISLNK(inode->i_mode))
+ return false;
+
+ if (!is_mount_owner(nd->path.mnt, current->fsuid))
+ return false;
+
+ *flags |= MS_SETUSER;
+ return true;
}

static int lives_below_in_same_fs(struct dentry *d, struct dentry *dentry)
@@ -1280,9 +1289,10 @@ static int do_loopback(struct nameidata
int clone_fl;
struct nameidata old_nd;
struct vfsmount *mnt = NULL;
- int err = mount_is_safe(nd);
- if (err)
- return err;
+ int err;
+
+ if (!permit_mount(nd, &flags))
+ return -EPERM;
+ if (!old_name || !*old_name)
+ return -EINVAL;
+ err = path_lookup(old_name, LOOKUP_FOLLOW, &old_nd);
--

```

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