Subject: [PATCH 2.6.19-rc3-mm1] VFS: BKL is not required for remount_fs()
Posted by vaverin on Mon, 30 Oct 2006 14:32:15 GMT
View Forum Message <> Reply to Message

From: Vasily Averin <vvs@sw.ru>

according to Documentation/filesystems/Locking remount_fs() is not requires BKL

Signed-off-by: Vasily Averin <vvs@sw.ru>

```
--- linux-2.6.19-rc3-mm1/fs/namespace.c.blkrm 2006-10-30 14:36:09.000000000 +0300
+++ linux-2.6.19-rc3-mm1/fs/namespace.c 2006-10-30 14:33:05.000000000 +0300
@@ -594,10 +594,8 @@ static int do_umount(struct vfsmount *mn
     */
    down_write(&sb->s_umount);
    if (!(sb->s_flags & MS_RDONLY)) {
-    lock_kernel();
     DQUOT_OFF(sb);
     retval = do_remount_sb(sb, MS_RDONLY, NULL, 0);
-    unlock_kernel();
    }
    up_write(&sb->s_umount);
    return retval;
--- linux-2.6.19-rc3-mm1/fs/super.c.blkrm 2006-10-30 14:36:01.000000000 +0300
+++ linux-2.6.19-rc3-mm1/fs/super.c 2006-10-30 14:33:05.000000000 +0300
@@ -609,16 +609,8 @@ static void do_emergency_remount(unsigne
    sb->s_count++;
    spin_unlock(&sb_lock);
    down_read(&sb->s_umount);
-   if (sb->s_root && sb->s_bdev && !(sb->s_flags & MS_RDONLY)) {
-    /*
-     * ->remount_fs needs lock_kernel().
-     *
-     * What lock protects sb->s_flags??
-     */
-    lock_kernel();
+   if (sb->s_root && sb->s_bdev && !(sb->s_flags & MS_RDONLY))
     do_remount_sb(sb, MS_RDONLY, NULL, 1);
-    unlock_kernel();
-   }
    drop_super(sb);
    spin_lock(&sb_lock);
   }
```