

---

Subject: [PATCH 3/3] lutimesat: actual syscall and wire-up on i386

Posted by [adobriyan](#) on Fri, 26 Jan 2007 11:17:03 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

lutimesat(2) does everything futimesat(2) does except it doesn't follow symlinks.

It could be used by tar(1) and cp(1).

FreeBSD and NetBSD have lutimes(2) which can be emulated by C library:

```
lutimesat(AT_FDCWD, filename, utimes)
```

Closes [http://bugme.osdl.org/show\\_bug.cgi?id=4433](http://bugme.osdl.org/show_bug.cgi?id=4433)

Signed-off-by: Alexey Dobriyan <[adobriyan@openvz.org](mailto:adobriyan@openvz.org)>

---

```
arch/i386/kernel/syscall_table.S | 1 +
fs/utimes.c                      | 9 ++++++++
include/asm-i386/unistd.h        | 3 +-
3 files changed, 12 insertions(+), 1 deletion(-)
```

```
--- a/arch/i386/kernel/syscall_table.S
+++ b/arch/i386/kernel/syscall_table.S
@@ -319,3 +319,4 @@ ENTRY(sys_call_table)
 .long sys_move_pages
 .long sys_getcpu
 .long sys_epoll_pwait
+ .long sys_lutimesat /* 320 */
--- a/fs/utimes.c
+++ b/fs/utimes.c
@@ -105,3 +105,12 @@ asmlinkage long sys_utimes(char __user *
{
    return sys_futimesat(AT_FDCWD, filename, utimes);
}
+
+asmlinkage long sys_lutimesat(int dfd, char __user *filename, struct timeval __user *utimes)
+{
+ struct timeval times[2];
+
+ if (utimes && copy_from_user(&times, utimes, sizeof(times)))
+ return -EFAULT;
+ return do_utimes(dfd, filename, utimes ? times : NULL, AT_SYMLINK_NOFOLLOW);
+}
--- a/include/asm-i386/unistd.h
+++ b/include/asm-i386/unistd.h
@@ -325,10 +325,11 @@ #define __NR_vmsplice 316
```

```
#define __NR_move_pages 317
#define __NR_getcpu 318
#define __NR_epoll_pwait 319
+#define __NR_lutimesat 320
```

```
#ifdef __KERNEL__
```

```
-#define NR_syscalls 320
+#define NR_syscalls 321
```

```
#define __ARCH_WANT_IPC_PARSE_VERSION
#define __ARCH_WANT_OLD_READDIR
```

---