Subject: Lecture on Operating System Lecture Exercise 3.1
From: IKENOYA Katsutoshi <j05002@ie.u-ryukyu.ac.jp>
Date: Tue, 06 Feb 2007 16:40:11 +0900
To: Shinji KONO <kono@ie.u-ryukyu.ac.jp>

学籍番号 : 055702B

・修正点
library側を修正しました。

問題3.1
Linux のkernel のソースを読んで、system call 関連のソースは、Linux では
どこにあるかを指摘せよ。(注: kernel 側とlibrary側の二つがある)

・kernel側は，"/usr/src/linux-2.4.27/arch/i386/kernel/entry.S"にある。
以下はentry.Sの抜粋である。

```
ENTRY(system_call)
pushl %eax # save orig_eax
SAVE_ALL
GET_CURRENT(%ebx)
testb $0x02,tsk_ptrace(%ebx) # PT_TRACESYS
jne tracesys
cmpl $(NR_syscalls),%eax
jae badsys
call *SYMBOL_NAME(sys_call_table)(,%eax,4)
movl %eax,EAX(%esp) # save the return value
```

省略

```
ENTRY(sys_call_table)
.long SYMBOL_NAME(sys_ni_syscall) /* 0 - old "setup()" system call*/
.long SYMBOL_NAME(sys_exit)
.long SYMBOL_NAME(sys_fork)
.long SYMBOL_NAME(sys_read)
.long SYMBOL_NAME(sys_write)
.long SYMBOL_NAME(sys_open) /* 5 */
.long SYMBOL_NAME(sys_close)
.long SYMBOL_NAME(sys_waitpid)
.long SYMBOL_NAME(sys_creat)
.long SYMBOL_NAME(sys_link)
.long SYMBOL_NAME(sys_unlink) /* 10 */
.long SYMBOL_NAME(sys_execve)
.long SYMBOL_NAME(sys_chdir)
.long SYMBOL_NAME(sys_time)
```
省略

call *SYMBOL_NAME(sys_call_table)(,%eax,4) で eax に指定された
システムコール番号の処理を呼び出す。
eax のシステムコール番号と実際にコールされる sys_XXXX 関数との
対応表は ENTRY(sys_call_table)にある。

・library側

glibc-2.4/sysdeps/unix/sysv/linux/i386/sysdep.h
を見てみた。

省略

```
#ifdef I386_USE_SYSENTER
# ifdef SHARED
# define ENTER_KERNEL call *%gs:SYSINFO_OFFSET
# else
# define ENTER_KERNEL call *_dl_sysinfo
# endif
#else
# define ENTER_KERNEL int $0x80
#endif
```

省略

```
#define DO_CALL(syscall_name, args) ¥
PUSHARGS_##args ¥
DOARGS_##args ¥
movl $SYS_ify (syscall_name), %eax; ¥
ENTER_KERNEL ¥
POPARGS_##args
```

省略

システムコール関連の記述があるので、library側は
glibc-2.4/sysdeps/unix/sysv/linux/i386/sysdep.h
にある。