

Re: PSL_RF inclusion in PSL_USERCHANGE for i386

Source: <http://unix.derkeiler.com/Mailing-Lists/FreeBSD/arch/2007-09/msg00011.html>

- *From:* Kostik Belousov <kostikbel@xxxxxxxxxx>
 - *Date:* Tue, 4 Sep 2007 06:53:06 +0300
-

On Mon, Sep 03, 2007 at 11:18:37AM +1000, Bruce Evans wrote:

On Sun, 2 Sep 2007, Roman Divacky wrote:

in i386/i386/machdep.c the set_regs() function sets i386 registers (called by ptrace for example). it checks what eflags are being changed and compares that with a mask of allowed flags to be changed. the mask is defined in psl.h like this:

```
#define PSL_USERCHANGE (PSL_C | PSL_PF | PSL_AF | PSL_Z | PSL_N  
| PSL_T \  
| PSL_D | PSL_V | PSL_NT | PSL_AC | PSL_ID)
```

PSL_RF (Flag to ensure single-step only happens once per instruction).
Can someone tell me why this is omitted? I think its because of having in-kernel debugger.

I think it is just because user mode cannot set this flag directly, except probably in vm86 mode (vm86 support code already has special cases for it). (Old) docs say that it can be set by popfl and ired, but popfl doesn't set it for me now and user mode cannot execute ired (?).

It can. It would result in exception when the normal privilege checks triggers, but would execute as expected otherwise. For instance,

```
#include <sys/syscall.h>
```

```
.text
```

```
.globl main  
.type main, @function  
main: pushl $12 /* _exit() code */
```

Re: PSL_RF inclusion in PSL_USERCHANGE for i386

```
pushfl
pushl %cs
pushl $2f

iretl

1: movl $SYS_exit, %eax
pushl %eax
int $0x80
```

```
2: pushl $hello
call printf
popl %eax
jmp 1b
```

```
.size main, . - main
```

```
hello: .asciz "Hello from iret\n"
```

Attachment: [pgpAY5vWXBRJ2.pgp](#)

Description: PGP signature