

Unix-filsystemsintro og præcisering af mål med miniprojektet

Martin Toft

mt@cs.aau.dk

Ph.D.-studerende
Distribuerede og Indlejrede Systemer
Institut for Datalogi
Aalborg Universitet

3. maj 2010

Miniprojekt i Principper for Samtidighed og Styresystemer (PSS)

Martin Toft

`mt@cs.aau.dk`

19. april 2010

1 Introduktion

I dette miniprojekt vil vi udforske Linux-kernen [1] ved at skrive et kernemodul. Processer i user space skal kunne skrive til og læse fra modulet via en device-fil i kataloget `/dev`, og modulet skal tælle differencen i antal tegn, det har modtaget og sendt. Via en anden device-fil skal man kunne aflæse denne tæller. Optællingen skal ske på en måde, der er multiprocessor- og preemption-sikker.

Miniprojektet besvares ved at hente L^AT_EX-kildekoden til et besvarelsesdoku-

Spørgsmål: Hvad skal vi egentlig lave?

Unix-filsystemsintro

- Ikke drevbogstaver (C:, D:, osv.) som Windows
- En rod-mappe, /, hvor hele filsystemstræet starter
- man `hier` (i en terminal eller Google)
- Det grove overblik:

/bin	Eksekverbare filer	/proc	Systeminformation
/boot	Bootloader-filer	/root	Roots hjemmemappe
/dev	Device-filer	/sbin	Eksekverbare filer
/etc	"Editable Text Conf."	/tmp	Midlertidige filer
/home	Hjemmemapper	/usr	"Unix System Resources"
/lib	Libraries	/var	Bl.a. logfiler
/mnt	Midlertidige mounts		

Eksempler fra Ubuntu Linux

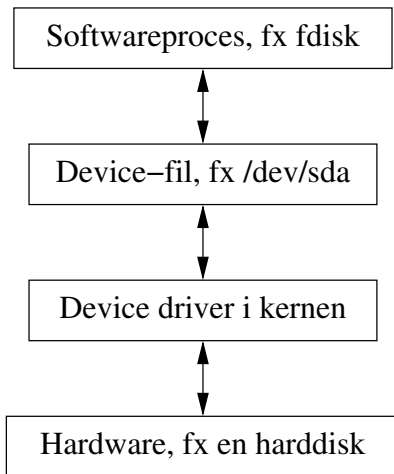
```
$ which ls fdisk
/bin/ls
/sbin/fdisk
$ file /boot/vmlinuz* | dd bs=1 count=50 2> /dev/null
/boot/vmlinuz-2.6.31-16-generic: Linux kernel x86
$ ls -l /dev/{null,zero,sda}
crw-rw-rw- 1 root root 1, 3 2010-05-01 13:47 /dev/null
brw-rw---- 1 root disk 8, 0 2010-05-01 13:47 /dev/sda
crw-rw-rw- 1 root root 1, 5 2010-05-01 13:47 /dev/zero
$ head -n 1 /etc/resolv.conf
nameserver 10.200.18.4
$ ls -ld /home/mt
drwxr-xr-x 56 mt mt 4096 2010-05-01 17:54 /home/mt
$ grep remaining /proc/acpi/battery/BAT0/state
remaining capacity:      5100 mAh
```

/dev

- Indeholder device-filer
- Device-fil = interface til device driver
- “In Unix, everything is a file”
- Software kommunikerer med et device via I/O-kald
- “Kendte” devices i /dev:

```
crw-rw----+ 1 root audio 14, 4 2010-05-01 13:47 audio
crw-rw----+ 1 root audio 14, 0 2010-05-01 13:47 mixer
crw-rw-rw-  1 root root   1, 3 2010-05-01 13:47 null
crw-rw-rw-  1 root root   1, 8 2010-05-01 13:47 random
brw-rw----  1 root disk   8, 0 2010-05-01 13:47 sda
crw-rw-rw-  1 root root   1, 9 2010-05-01 13:47 urandom
crw-rw-rw-  1 root root   1, 5 2010-05-01 13:47 zero
```

/dev



/dev

- Major- og minor-numre
- Block og character devices
- Oprettelse vha. `mknod`

```
crw-rw-rw-  1 root root    1, 3 2010-05-01 13:47 null
brw-rw----  1 root disk    8, 0 2010-05-01 13:47 sda
crw-rw-rw-  1 root root    1, 5 2010-05-01 13:47 zero
```

Miniprojekt

- Spørgsmål: Hvad skal vi egentlig lave?
- Svar: En character device **driver**, der opfylder kravene i dokumentet “Miniprojekt i Principper for Samtidighed og Styresystemer (PSS)”
- Gevinst: Pensumreduktion og indsigt i kerneprogrammering

```
$ wc -l helloworld.c chardev.c chardev2.c
  21 helloworld.c
 163 chardev.c
 167 chardev2.c
```