

**MEMAHAMI KONSEP
PROSES.
MEMAHAMI HUBUNGAN
PROSES PARENT DAN
CHILD.
MEMAHAMI SINYAL DAN
PENGIRIMAN SINYAL.**

05

EDISI I - 2007

**MATA DIKLAT :
SISTEM OPERASI**

**PROGRAM KEAHLIAN :
SEMUA PROGRAM KEAHLIAN**

**DEPARTEMEN PENDIDIKAN NASIONAL
BIRO PERENCANAAN DAN KERJASAMA LUAR NEGERI**

2007

**MEMAHAMI KONSEP PROSES.
MEMAHAMI HUBUNGAN PROSES PARENT
DAN CHILD.
MEMAHAMI SINYAL DAN PENGIRIMAN
SINYAL.**

EDISI I - 2007

5.1 PENDAHULUAN

5.1.1 Deskripsi

NAMA MODUL : SISTEM OPERASI

**KOMPETENSI : MEMAHAMI KONSEP PROSES.
MEMAHAMI HUBUNGAN PROSES PARENT
DAN CHILD.
MEMAHAMI SINYAL DAN PENGIRIMAN
SINYAL.**

SUB KOMPETENSI : Proses Dan Manajemen Proses

KRITERIA KERJA : -

LINGKUP BELAJAR :

- Perintah untuk menampilkan status proses : ps, pstree
- Variasi opsi pada perintah ps
- Perintah mengirim sinyal : kill

5.2 Perintah Untuk Menampilkan Status Proses

Yang termasuk perintah untuk menampilkan status proses, diantaranya

5.2.1 ps

5.2.2 pstree

5.2.1 ps

Utilitas ps berguna untuk menampilkan status dari proses yang sedang berjalan pada sistem. Contoh :

```
# ps
  PID TTY          TIME CMD
 8907 tty2        00:00:00 bash
 9203 tty2        00:00:00 ps
#_
```

Arti dari informasi di atas :

- Kolom PID menyatakan nomor proses
- Kolom TTY menyatakan identitas terminal yang berkaitan dengan proses
- Kolom TIME menyatakan waktu total CPU untuk menjalankan proses (dalam satuan detik)
- Kolom COMMAND adalah nama proses yang sedang berjalan.

5.2.2 pstree

-

5.3 Variasi Opsi Pada Perintah PS

5.3.1 Menampilkan Informasi Yang Lengkap

Dengan menyertakan pilihan -f (dari kata “full listing”), informasi dari setiap proses akan menjadi lebih lengkap.

```
# ps -f
UID      PID  PPID  C    STIME TTY     TIME      CMD
andi    8907  1094  0    21:30 tty2    00:00:00  ~bash
andi    9205  8907  0    21:45 tty2    00:00:00  ps -f
# _
```

Arti dari informasi di atas :

- Kolom PID menyatakan nomor proses
- Kolom TTY menyatakan identitas terminal yang berkaitan dengan proses
- Kolom TIME menyatakan waktu total CPU untuk menjalankan proses (dalam satuan detik)
- Kolom COMMAND adalah nama proses yang sedang berjalan.
- UID menyatakan nama pemakai yang menjalankan proses
- STIME menyatakan saat awal proses dimulai
- PPID menyatakan proses induk dari proses yang sedang berjalan.

5.3.2 Menampilkan Semua Proses

Dengan memberikan pilihan -e (dari kata “everything”), semua proses yang sedang berjalan pada sistem akan ditampilkan.

```
# ps -e
      PID  TTY          TIME CMD
      1  ?          00:00:04 init
      2  ?          00:00:00 keventd
      3  ?          00:00:00 kamp-idled
      4  ?          00:00:00 ksoftirqd_CPU0
      5  ?          00:00:00 kswapd
      6  ?          00:00:00 kreclaimd
      7  ?          00:00:00 bdflush
      8  ?          00:00:00 kupdated
      9  ?          00:00:00 mdrecoveryd
     13  ?          00:00:00 kjournald
     88  ?          00:00:00 khubd
    213  ?          00:00:00 kjournald
    214  ?          00:00:00 kjournald
    215  ?          00:00:00 kjournald
    216  ?          00:00:00 kjournald
    669  ?          00:00:00 syslogd
    674  ?          00:00:00 klogd
    694  ?          00:00:00 portmap
```

```

722 ? 00:00:00 rpc, statd
834 ? 00:00:00 apmd
890 ? 00:00:00 sshd
923 ? 00:00:00 xinetd
963 ? 00:00:00 sendmail
982 ttys0 00:00:00 gpm
1000 ? 00:00:00 crond
1050 ? 00:00:00 xfs
1086 ? 00:00:00 atd
1094 tty2 00:00:00 login
1095 tty3 00:00:00 login
1096 tty4 00:00:00 mingetty
1097 tty5 00:00:00 mingetty
1098 tty6 00:00:00 mingetty
8906 tty1 00:00:00 login
8907 tty2 00:00:00 bash
9072 tty1 00:00:00 bash
9112 tty3 00:00:00 bash
9207 tty2 00:00:00 ps

```

#_

Semakin banyak pemakai yang sedang menggunakan sistem, maka akan bertambah banyak proses yang akan ditampilkan. Tanda ? menyatakan bahwa proses tersebut adalah proses daemon (proses yang hidup terus selama sistem hidup dan berjalan di latar belakang).

5.3.3 Tanpa Menyertakan Pemimpin Grup Proses

Sejumlah proses yang ditampilkan melalui `ps -e` berkedudukan sebagai pemimpin grup proses (proses group leader), yaitu proses yang menjalankan proses yang lain. Sebagai contoh, jika anda memberikan perintah :

```
ps -f
```

maka hasil yang diperoleh :

```

# ps -f
UID    PID  PPID  C    STIME TTY    TIME    CMD
andi   8907  1094  0    21:30 tty2   00:00:00 ~bash
andi   9205  8907  0    21:45 tty2   00:00:00 ps -f
# _

```

proses **sh** sebenarnya adalah pemimpin grup proses. Tampak bahwa PID dari **sh** tercantum pada PPID dari proses **ps -f**.

Untuk memperoleh proses selain pemimpin grup proses, Anda dapat memberikan pilihan `-d`.

Contoh :

```
# ps -d
```

PID	TTY	TIME	CMD
1	?	00:00:04	init
2	?	00:00:00	keventd
3	?	00:00:00	kamp-idled
4	?	00:00:00	ksoftirqd_CPU0
5	?	00:00:00	kswapd
6	?	00:00:00	kreclaimd
7	?	00:00:00	bdflood
8	?	00:00:00	kupdated
9	?	00:00:00	mdrecoveryd
13	?	00:00:00	kjournald
88	?	00:00:00	khubd
213	?	00:00:00	kjournald
214	?	00:00:00	kjournald
215	?	00:00:00	kjournald
216	?	00:00:00	kjournald
1050	?	00:00:00	xfs
8907	tty2	00:00:00	bash
9072	tty1	00:00:00	bash
9112	tty3	00:00:00	bash
9210	tty2	00:00:00	ps

_

Proses-proses di ataslah yang sebenarnya dalam keadaan aktif.

5.3.4 Menampilkan Proses yang Terkait dengan Seorang Pemakai

ps juga menyediakan pilihan yang digunakan untuk melihat proses-proses yang terkait dengan seorang pemakai. Pilihan yang diperlukan adalah -u. Contohnya :

ps -u andi

PID	TTY	TIME	CMD
9072	tty1	00:00:00	bash
9212	tty1	00:00:00	vi

_

5.3.5 Menampilkan Proses yang Terkait dengan Seorang Pemakai

Dengan memberikan pilihan -t, proses yang terkait dengan suatu terminal akan ditampilkan. Contoh :

ps -t tty05

PID	TTY	TIME	CMD
8906	tty1	00:00:00	login
9072	tty1	00:00:00	bash
9212	tty1	00:00:00	vi

_

5.4 Perintah Mengirim Sinyal

Yang termasuk perintah mengirim sinyal : kill

5.4.1 Kill

Karakter **kill** bermanfaat untuk membatalkan semua karakter yang telah diketikkan. Tombol yang biasa digunakan untuk menghasilkan karakter kill adalah sebagai berikut :

Kode Spesial	Tombol
Kill	@ Ctrl+X Ctrl+U

Tombol yang digunakan karakter **kill** bergantung oleh sistem yang anda miliki. Untuk mengetahuinya anda dapat memanfaatkan perintah **stty** dengan pilihan -a, seperti halnya pada <erase>.

```
# stty -a
ed 38400 baud; rows 24; columns 80; line = 0;
intr = ^C; quit = ^\; erase = ^?; kill ^U; <dan seterusnya>
# _
```

Pada contoh diatas, tombol yang digunakan untuk membangkitkan kode kill adalah <Ctrl+U>.

```
# datte <kill>
```

```
—
```

```
# _
```

Contoh diatas, perintah **datte** diabaikan, **shell** menunggu perintah selanjutnya.