

BAB IV Shell Linux

- `/bin/bash`
- `/bin/csh`
- `/bin/bash`

Shell adalah "*command executive*" artinya program yang menunggu instruksi dari pemakai, memeriksa syntax dari instruksi yang diberikan, kemudian mengeksekusi perintah tersebut.

Shell ditandai dengan prompt. Untuk pemakai biasa berupa tanda "\$" dan "#" untuk Superuser(root). Linux mengembangkan shell yang kompatibel dengan Bourne-Shell juga mengadaptasi kemampuan yang ada di Korn-Shell. Shell ini disebut sebagai **bash** (*Bourne Again Shell*) dengan nama program `/bin/bash`.

Perbedaan mendasar antara shell tersebut hampir tidak ada, kecuali pada fasilitas pemrograman dan editing. Shell dapat dipilih pada saat membuat nomor account baru atau mengubahnya pada file `/etc/passwd`.

```
#head -n 1 /etc/passwd
```

```
root:x:0:0:root:/root:/bin/bash
```

Profile

Pada saat login, program akan menjalankan beberapa scripts, yaitu :

1) /etc/profile

Berisi shell script yang berlaku untuk seluruh pengguna Linux yang melakukan sesi login.

2) Profile untuk setiap pemakai

Pada home directory pemakai, login pertama kali akan memeriksa file **.bash_profile**. Bila ada maka file ini akan dieksekusi. Bila tidak ada, maka file **.bash_login** akan dicari bila ditemukan maka dieksekusi.

Bila **.bash_login** tidak ada juga, maka program mencari file **.profile** untuk di eksekusi.

3) **.bashrc**

File ini akan dieksekusi untuk perpindahan dari satu shell ke shell lain melalui instruksi "su" (*substitusi user*).

4) **.bash_logout**

Saat logout maka bash akan mencari file **.bash_logout** untuk dieksekusi sebelum logout.

/etc/profile

```
# /etc/profile

# System wide environment and startup programs
# Functions and aliases go in /etc/bashrc
PROFILE_LOADED=1

PATH="$PATH:/usr/X11R6/bin

USER=`id -un`
LOGNAME=$USER
MAIL="/var/spool/mail/$USER"

HOSTNAME=`/bin/hostname`
HISTSIZE=1000

export PATH PS1 USER LOGNAME MAIL HOSTNAME HISTSIZE
```



PATH

Merupakan daftar nama direktori. Bila sebuah instruksi diberikan dari prompt shell, maka instruksi tersebut akan dicari pada daftar tersebut.



PS1

Adalah prompt, untuk jelasnya :

```
PS1 =' [\u@\h \W]\$\ '
```

<code>\u</code>	= nama user
<code>\h</code>	= nama host
<code>\W</code>	= nama working directory
<code>\!</code>	= menunjukkan nomor history
<code>\\$</code>	= standar user prompt
<code>\/</code>	= slash
<code>\s</code>	= nama shell
<code>\t</code>	= waktu saat itu
<code>\w</code>	= full working directory

Variabel PS1 dapat diedit sesuai dengan keinginan kita, contoh :

```
[user@live/tmp]$export PS1 = '[\t\w]'
[15:39:11/tmp]
```

PS2

PS2 adalah prompt string yang akan dimunculkan, bila shell memerlukan input.

```
[user@live/tmp]echo $PS2
>
[user@live/tmp]echo "test test
>string belum lengkap
>"
test test
string belum lengkap
$
```

misal kita rubah PS2

```
$PS2="Prompt Baru>"
$echo "test test
Prompt Baru>"
test test
$
```

.bash_profile

File ini dijalankan pada Home Directory pemakai yang login.

.bash_logout

File `.bash_logout` akan di eksekusi sesaat sebelum logout dilakukan, oleh karena itu script ini sesuai untuk menghapus temporary files atau job lainnya.

```
#cat .bash_logout
#~/ .bash_logout
clear
```

History

Catatan dari semua instruksi yang sejauh ini telah dilakukan, dimana history tersebut dapat dipilih kembali, diedit dan dieksekusi.

History umumnya memudahkan pemakai untuk mengedit kembali instruksi kompleks dan panjang, terutama bila terjadi kesalahan pada penulisan maupun parameter. Besarnya history dapat diatur pada variabel HISTSIZE.

```
$echo $HISTSIZE
1000
```

Melihat daftar instruksi yang telah dilakukan dengan command **history**.

```
[user@mandiri tmp]$ history
 8  ./setup
 9  ls
10  ./setup
11  ls -al
12  ll
13  ls -al | head
14  cd ..
```

Untuk mengeksekusi kembali ketik **!<history>**

```
$!11
```

Instruksi LINUX-Shell

Format umum instruksi linux sebagai berikut:

```
$ Nama instruksi -[options] [argumen]
```



Manual

Linux menyediakan manual secara on-line, dokumentasi dari kelompok paket-paket.

Untuk melihat manual binary ketik :

```
$man <binary>
```

```
$man -k <keyword>
```

misalnya :

```
$man joe
```



apropos

Memberikan jawaban berupa hasil akses ke database "whatis" yang berkaitan dengan kunci kata yang ditanyakan.

```
$apropos xawtv
```

```
xawtv      (1) - a X11 program for watching TV
```



Penanganan File

File pada direktori dapat dilihat dengan instruksi "ls" (list).

```
$ls
```

```
tmp/      nsmail/  report.txt
```

Melihat secara detail (long)

```
$ls -l
```

Melihat semua file termasuk hidden file (di linux hidden file selalu diawali dengan titik)

```
$ls -a
```

```
.bash_logout      .xawtv
.bash_history     .ggviewrc
.radiofm          .screenrc
```

Tipe File

Tipe file dapat di ketahui dengan menggunakan instruksi :

```
$file <nama file>
```

```
$file /etc/passwd
```

```
/etc/passwd: ASCII text
```

```
$file /usr/bin/telnet
```

```
/usr/bin/telnet: ELF 32-bit LSB executable, Intel 80386,
version 1, dynamically linked (uses shared libs), stripped
```

Menyalin File

Menggunakan instruksi "cp" (copy) :

```
cp [option] file-asal file-tujuan
```

```
$cp /tmp/test.txt /home/mandiri/test-juga.txt
```

menyalin file test.txt yang terdapat di /tmp ke tujuan
diretori /home/mandiri dengan nama baru test-juga.txt Untuk

melihat option nya dapat dilihat melalui manualnya (ketik `man cp`).

Menampilkan isi file

Menggunakan instruksi `more`, `cat`, `pg` dan lainnya.

```
$cat test.txt  
  
test test test test test  
test test test test test  
test test test test test  
test test test test test
```

Menampilkan file per satu layar

penuh

```
$more test.txt  
  
test test test test test  
test test test test test  
test test test test test  
test test test test test
```

jika isi file melebihi satu layar maka tampilan terhenti dan dapat dilanjutkan dengan menekan spacebar. Untuk

lengkapnya silahkan lihat manualnya.

Mengubah nama file

Memakai instruksi `mv` (move), yaitu mengubah nama file menjadi nama yang lain.

```
$mv /tmp/test.txt /tmp/test-rubah.txt
```

Menghapus file

Memakai instruksi `rm` (remove).

```
$rm /tmp/test.txt
```

Menampilkan kata dalam suatu file

Memakai instruksi `grep`.

```
$grep root /etc/passwd  
root:x:0:0:root:/root:/bin/bash  
operator:x:11:0:operator:/root:
```

Membuat direktori

Memakai instruksi `mkdir`.

```
$mkdir dirbaru  
$mkdir dir1 dir2 dir3 #membuat 3 direktori sekaligus
```

Menghapus direktori

Memakai Instruksi "rmdir", dengan anggapan bahwa tidak ada file dalam direktori tersebut.

```
$rmdir dir1
```

Menampilkan direktori saat ini

Memakai instruksi "pwd" (print working directory).

```
$pwd
```

```
/tmp
```

Berpindah direktori

Memakai instruksi "cd" (change directory).

```
$pwd
```

```
/tmp
```

```
$cd /home/mandiri
```

```
$pwd
```

```
/home/mandiri
```

cut

Perintah cut mendefinisikan suatu file yang berisi data berdasarkan kolom

BAB IV Shell Linux2

Syntax :

```
$ cut [option] file
```

Contoh : kita akan cut file 1 dari kolom 1-3

```
$ cut -c 1-7 test
```

```
test
```

```
test1
```

```
test2
```

```
$ cat 1
```

```
test test
```

```
test1 test1
```

```
test2 test2
```

Dari contoh dapat terlihat bahwa cut hanya menampilkan isi dari file test dari kolom 1-3

find

Perintah ini berfungsi untuk mencari file ataupun directori.

Syntax :

```
$ find /path [option]
```

Contoh :

```
$ find . -name 1
```

```
1          431.txt
```

```
12.txt
```

ln

Kegunaan perintah ini adalah untuk membuat link dari satu file/directori ke file/directori lain

Syntax :

```
$ ln -s /path/to/source target
```

Contoh :

```
$ ln -s 1 2
```

```
mandiri@faiz:~$ ls -l
```

```
total 2
```

```
-rw-r--r-- 1 mandiri mandiri 102 Nov 9 14:11 test
```

```
lrwxrwxrwx 1 mandiri mandiri 4 Nov 10 03:21 2 -> 1
```

Dari contoh dapat dilihat bahwa file 2 adalah merupakan link dari file 1.

locate

Perintah ini digunakan untuk mengetahui dimana letaknya sebuah file atau directori. Fungsinya kurang lebih sama dengan find, bedanya locate menggunakan sebuah database (biasanya terletak di `/var/lib/locatedb`) yang dapat di update menggunakan perintah `updatedb`.

Syntax :

```
$ locate <something>
```

Contoh :

```
$ locate 1
```

```
/tmp/1
```



ps

Perintah ini digunakan untuk melihat setiap proses yang dijalankan di dalam mesin beserta nomor prosesnya.

Syntax :

```
$ ps [option]kill
```

Perintah kill (mematikan) sesuai namanya adalah berfungsi untuk mengirimkan signal ke suatu proses. Sehingga bisa memerintahkan apakah suatu proses ditunda, dihentikan atau dilanjutkan. Tetapi paling sering digunakan untuk mematikan proses yang sedang berjalan.

Syntax :

```
$ kill <nomor-prosesnya>
```



tail

Perintah ini berlawanan dengan perintah cut, tail mendefinisikan sebuah data pada file menurut barisnya. Sedikit perbedaan adalah pada tail secara default

ditampilkan 10 baris terakhir dari isi file

Syntax :

```
$ tail [option] <namafile>
```

Contoh :

```
$ tail -n 1 /etc/passwd
```

```
mandiri:x:502:506::/home/mandiri:/bin/bash
```

Bila kita ingin menampilkan hanya n baris terakhir gunakan option [-[nilai]].



adduser, passwd dan userdel

Perintah `adduser` dan `userdel` adalah file-file untuk administrasi user. `adduser` digunakan untuk menambahkan user pada suatu mesin. Kemudian setelah ditambahkan ditentukan password user yang baru tersebut dengan perintah `passwd`. Sedangkan `userdel` digunakan untuk menghapus user di mesin. Untuk menjalankan perintah-perintah ini diperlukan root akses.

Syntax :

```
# adduser <namauser>
```

```
# passwd <namauser>
```

```
# userdel [-r] <namauser>
```

Editor



VI

Vim merupakan pengembangan dari editor vi yang mempunyai keunggulan dalam penggunaan warna. Vi merupakan editor yang paling terkenal di dunia UNIX. Vi ialah kepanjangan dari "Visual" editor. Vim mendukung penggunaan warna untuk menandakan kode program dan juga penanda teks seperti penggunaan huruf normal, tebal maupun cetak miring. Untuk menjalankan perintah vi, kita gunakan sintaks sbb :

```
vi <nama_file>
```

Status vi

Editor vi mempunyai 3 mode, yaitu :

-command mode, untuk menggunakannya tekan ESC terlebih dahulu.

-input mode, dengan mengetikkan perintah-perintah yang spesifik, seperti a I o 0 dan diakhiri dengan ESC atau dengan interrupt.

_ line mode, suatu keadaan setelah ada proses input : ?, /,

!

Notasi yang digunakan

CTRL-c, CTRL adalah tombol control pada keyboard, sehingga CTRL-c berarti "control c". CR (Carriage Return) ialah tombol enter.

Manipulasi file

Tombol Fungsi

:wq save file dan exit

:q keluar dari vi

:q! keluar dari vi tanpa menyimpan

:w menyimpan file

Perpindahan kursor

Tombol Fungsi

h pindah satu karakter ke kiri

i pindah satu karakter ke kanan

j pindah satu baris ke bawah

k pindah satu baris ke atas

CTRL-d menggulung setengah halaman ke bawah

CTRL-u menggulung setengah halaman ke atas

CTRL-f menggulung satu halaman ke bawah

CTRL-b menggulung satu halaman ke atas

:0 pindah ke awal file

:\$ pindah ke akhir file

0 pindah ke awal baris

\$ pindah ke akhir baris

CR pindah ke awal baris berikutnya

<n> pindah menuju kolom ke- <n>

<n>G pindah menuju baris ke- <n>

w pindah ke awal kata yang "sedikit" berikutnya

W pindah ke awal kata yang "banyak" berikutnya

b pindah ke awal kata yang "sedikit" berikutnya

B pindah ke awal kata yang "banyak" berikutnya

e pindah ke akhir kata yang "sedikit" berikutnya

E pindah ke akhir kata yang "banyak" berikutnya

) pindah ke akhir kalimat

(pindah ke awal kalimat

} pindah ke akhir paragraf

{ pindah ke awal paragraf

Penghapusan

Tombol Fungsi

x Menghapus karakter yang ada di kursor

X Menghapus karakter sebelum kursor

D Menghapus satu baris

d^ Menghapus mulai baris awal hingga ke posisi kursor

<n>dd Menghapus <n> baris dimulai dengan baris pada posisi kursor

d<n>w Menghapus <n> kata dimulai dari kata pada posisi kursor

Copy dan paste

Tombol Fungsi

Y : copy baris saat ini ke dalam "buffer yank"

<n>yy copy <n> baris dimulai dari baris saat ini ke "buffer yank"

p : Paste isi yang ada di "buffer yank" ke posisi setelah kursor

P Paste isi yang ada di "buffer yank" ke pisisi sebelum kursor

Perintah cari dan ganti

Tombol Fungsi

/<teks> Mencari <teks> yang ada setelah posisi kursor

?<teks> Mencari <teks> yang ada sebelum posisi kursor

/ Mengulangi pencarian terakhir setelah posisi kursor

? Mengulangi pencarian terakhir sebleum posisi kursor

<addr>s/<from>/<to>g Mencari kata <from> dan menggantikannya dengan kata

<to> di baris ini atau di kisaran <add>Contoh :

3,20s/lama/baru/g contoh ini akan mencari kata "lama"

diganti dengan kata "baru" yang ada diantara baris 3-20

Perintah lainnya

Tombol Fungsi

u Undo perubahan terakhir

CTRL-g Menampilkan nama file dan baris kursor

Insert Memulai mengedit (mengetik)

Delete Menghapus karakter di posisi kursor



Pico

Pico adalah teks editor jenis lain yang berdasarkan pada program compose email Pine. Navigasi untuk perintahnya diletakkan di bawah dari layar. Semua karakter yang diketik langsung disisipkan ke dalam teks. Perintah untuk menjalankan editor pico adalah sbb :

```
pico <nama file>
```

Tombol Fungsi

^G Help

^O Menyimpan file

^R Membuka file

^K Menghapus satu baris

^U Undo untuk perintah menghapus

^W Mencari kata

^C Menampilkan posisi kursor saat ini

^X Keluar dari pico



Joe

Perintah-perintah dalam editor joe ini sangat mirip dengan perintah-perintah pada WordStar. Jika kita telah terbiasa dengan perintah pada WordStar maka tidak akan kesulitan dalam menggunakan editor joe. Untuk menjalankan joe, kita dapat menggunakan perintah sbb :

```
joe <nama file>
```

Tombol Fungsi

^KD simpan file dan kembali ke menu edit

^KX simpan file dan keluar

^KB memberikan tanda awal untuk mem-blok

^KK memberikan tanda akhir untuk mem-blok

^KC mengkopi daerah yang telah diblok

^KY menghapus daerah yang telah diblok

^KM memindahkan daerah yang telah diblok

^Y menghapus satu baris

^X memindahkan kursor ke kata berikutnya

^Z memindahkan kursor ke kata sebelumnya

^KU memindahkan kursor ke tempat paling awal dari file

^KV memindahkan kursor ke tempat paling akhir dari file

^A memindahkan kursor ke awal baris

^E memindahkan kursor ke akhir baris

^U menggulung layar ke layar sebelumnya

^V menggulung layar ke layar berikut



MC (*Midnight Commander*)

MC atau *Midnight Commander* merupakan aplikasi yang sangat berguna. Kemampuan mc sebagai editor hanya salah satu dari kemampuan-kemampuan lainnya. Untuk menjalankannya, digunakan perintah sbb :

```
mc -e <nama file> atau
```

mc kemudian insert file yang akan diedit lalu tekan F4 untuk mengedit. Perintah-perintah yang dapat digunakan sebagai berikut :

Tombol Fungsi

- F1** Keterangan bantu (help)
- F2** Simpan file
- F3** Untuk memblok
- F4** Untuk mengganti kata
- F5** Menyalin suatu blok
- F6** Memindahkan suatu blok
- F7** Mencari kata yang ingin diedit
- F8** Menghapus suatu blok
- F9** Memunculkan menu navigasi
- F10** Mengakhiri editor

Masing-masing editor ini mempunyai kelebihan dan kekurangan. Contohnya seperti pada editor pico yang sangat baik berjalan di terminal apapun, termasuk pada terminal bila kita sedang telnet dengan menggunakan telnet client standar dari Windows. Sedangkan vim/vi, kita gunakan karena banyak feature-feature yang sangat menarik sehingga membantu kita untuk lebih cepat dalam mengedit file. Editor joe dan mc, digunakan karena sederhana dan tampilannya yang menarik.
