**Bekerja Dengan Bash Shell**

**Tujuan**

Setelah mempelajari materi dalam bab ini, mahasiswa diharapkan mampu:

* Memahami shell pada sistem operasi Linux.
* Menggunakan feature history pada Bash Shell.
* Mengubah feature history pada Bash Shell.
* Mengubah prompt shell.
* Melakukan konfigurasi Bash Shell untuk menjalankan skrip secara otomatis.
* Membuat dan mengeksekusi shell script sederhana melalui editor vi.
* Memahami job control.
* Memahami stack.
* Menggunakan alias.

**Dasar teori:**

1. SHELL

Shell adalah *Command executive*, artinya program yang menunggu instruksi dari pemakai, memeriksa sintak dari instruksi yang diberikan, kemudian mengeksekusi perintah tersebut. Shell ditandai dengan prompt. Untuk pemakai menggunakan prompt $ dan untuk superuser menggunakan promp #.

Beberapa macam shell

* /bin/sh

Bourne shell, dirancang oleh Steve Bourne dari AT&T

* /bin/csh

Dikembangkan oleh UNIX Berkeley yang dikenal dengan C -Shell

* /bin/bash

Kompatibel dengan Bourne Shell dan juga mengadaptasi kemampuan Korn-Shell.

Perbedaan mendasar antara Shell diatasi hampir tidak ada, kecuali pada fasilitas pemrograman dan editing.

1. PROFILE

Pada saat login, program akan menjalankan beberapa program yaitu :

* /etc/profile

Berisi shell script yang berlaku untuk seluruh pengguna Linux.

* Profil untuk setiap pemakai

Pada home directory, login pertama kali akan memeriksa file .bash\_profile . Bila tidak ada, maka file .bash\_login akan dicari. Bila .bash\_login tidak ada, maka dicari file bernama .profile .

* .bashrc

File ini akan dieksekusi untuk perpindahan dari satu shell ke shell yang lain melalui instruksi su.

* .bash\_logout

Pada saat logout, maka bash akan mencari file .bash\_logout. Bila ada, file tersebut akan dieksekusi sebe lum logout

Isi dari /etc/profile:

# System wide environment and startup programs

# Functions and aliases go in /etc/bashrc

PATH=”$PATH:/usr/X11R6/bin” PS1=”[u@\h \W]\\$ “

umask 022

USER=’id –un’ LOGNAME=$USER MAIL=”/var/spool/mail/$USER”

HOSTNAME=’/bin/hostname’ HISTSIZE=1000

HISTFILESIZE=1000

Export PATH PS1 HOSTNAME HISTSIZE HISTFILESIZE USER LOGNAME MAIL

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

PS1 adalah prompt dimana

\u = Nama User

\h = Nama Host

\W = Nama working direktory

1. HISTORY

History diadaptasi dari C- Shell, yaitu catatan dari semua instruksi yang sejauh ini telah dilakukan. Catatan ini dapat dilihat sebagai history, kemudian dapat dipilih kembali, diedit dan dieksekusi. History memudahkan pemakai untuk mengedit kembali instruksi kompleks dan panjang, terutama bila terjadi kesalahan pada penulisan instruksi maupun parameter.

Navigasi pada daftar history menggunakan karakter kontrool sebagai berikut :

^P (Ctrl-P) melihat instruksi sebelumnya

^N (Ctrl-N melihat instruksi berikutnya

!! eksekusi kembali instruksi sebelumnya

!! –3 3 instruksi sebelumnya akan diulang

!!88 ulangi instruksi no 88

1. BASH -SCRIPT

Bash- script adalah file yang berisi koleksi program yang dapat dieksekusi. Untuk eksekusi bash script gunakan . sebelum file bash- script yang berarti eksekusi shell dan tanda ./ berarti file bash-script berada pada direktori actual.

1. JOB CONTROL

Job adalah sebuah eksekusi program yang diberikan kepada kernel. Sebuah Job dianggap selesai, bila eksekusi program tersebut berakhir. Eksekusi Job adalah sama dengan eksekusi program, baik proses *Background* maupun proses *Foreground*.

1. EDIOR vi

Vi adalah full screen editor, artinya editor tersebut dapat memanfaatkan fasilitas satu layar penuh. Vi mempunyai 2 buah modus, yaitu :

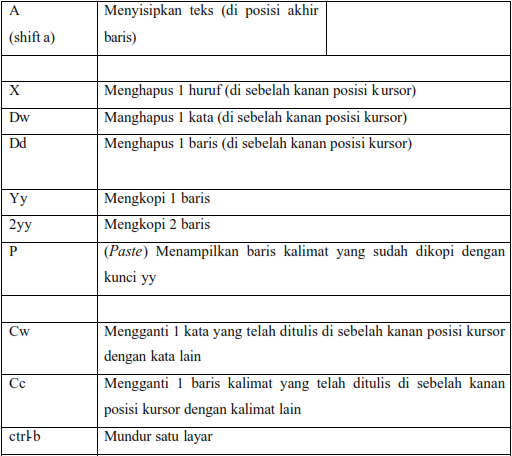
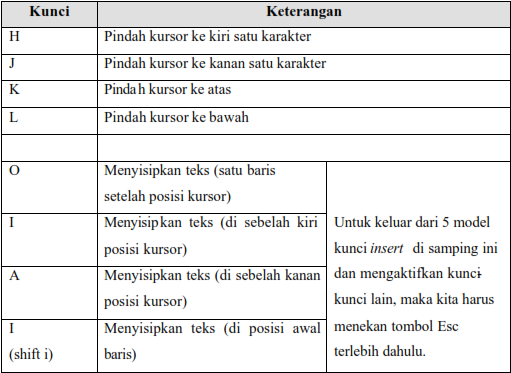
* Command line

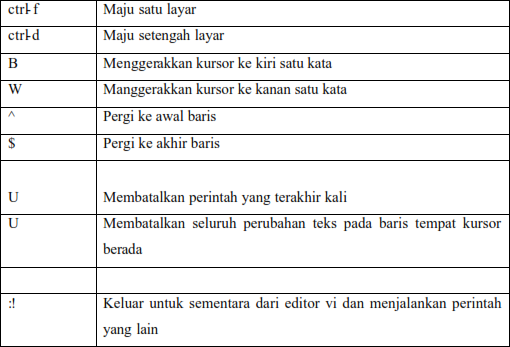
Editor vi mengintepretasikan input sebagai instruksi untuk dieksekusi oleh editor, contoh seperti mencari teks, mengganti teks secara otomatis dan lainnya.

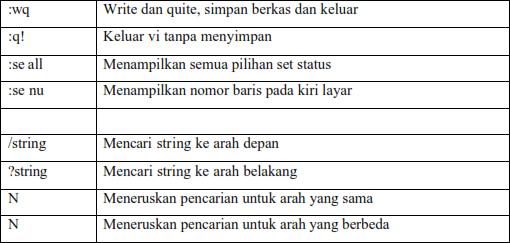
* Editing
* Editor vi mengintepretasikan input sebagai teks yang akan dimasukkan ke dalam buffer editor. Pada bagian bawah layar akan tampil teks “INSERTING”.

Pada awal vi dijalankan, maka program memasuki command mode. Dengan menekan tombol “i” maka akan memasuki editing. Untuk kembali ke command mode, tekan tombol Esc.

Kunci-kunci teks editor vi dapat dilihat pada tabel sebagai berikut :







TUGAS

Jawablah pertanyaan- pertanyaan di bawah ini :

1. Apa yang dimaksud dengan shell dan sebuatkan shell yang ada di system operasi Linix
2. Apa yang dimaksud dengan profile pada Bash Shell.
3. Apa yang Anda ketahui mengenai file .bashrc.
4. Apa yang dimaksud dengan history pada Bash Shell. Apa kegunaan perintah history, sebutkan cara- cara untuk mengetahui history perintah- perintah yang pernah digunakan oleh user!
5. Cobalah menggunakan editor vi untuk mengetik dan pahami perintah- perintah yang ada seperti yang terdapat pada dasar teori (untuk dilakukan, tidak perlu dijawab sebagai tugas pendahuluan). Perintah- perintah yang penting : insert huruf(kalimat), delete (per huruf, per kata dan per baris), simpan file dan keluar dari editori vi.

Praktikum

1. Login sebagai user.
2. Bukalah Console Terminal dan lakukan percobaan-percobaan di bawah ini kemudian analisa hasil percobaan.
3. Selesaikan soal- soal latihan.

Percobaan 1 : Profile

1. File .bash\_profile dijalankan pada home direktori pemakai yang login. File .bash\_profile adalah *hidden file* , sehingga untuk melihatnya gunakan opsi a pada instruksi ls.

$ *ls –a*

$ *more .bash\_profile*

1. File .bash\_logout akan diekseksi sesaat sebelum logout, berfungsi sebagai *house clearing jobs*, artinya membersihkan semuanya, misalnya menghapus temporary file atau job lainnya. Melihat file .bash\_logout dengan instruksi

$ *cat .bash\_logout*

Percobaan 2 : Menggunakan Feature History Bash

1. Bash shell menyimpan ”history” perintah yang digunakan sebelumnya. Anda dapat mengaksis history dalam beberapa cara. Cara paling mudah adalah menggunakan Panah Atas. Maka perintah sebelumnya akan ditampilkan.
2. Berikutnya, berikan Bash shell beberapa perintah untuk diingat. Masukkan perintah berikut dan tekan Enter pada setiap baris.

$ *cd*

$ *ls –l /etc*

$ *ls –l*

$ *whoami*

$ *who*

1. Untuk memeriksa apakah perintah ini ditambahkan pada history, dapat menggunakan perintah history untuk melihat semua perintah yang pernah dimasukkan.

$ *history*

1. Anda dapat memilih perintah sebelumnya dengan menggunakan Panah Atas , tetapi hal ini tidak efisien untuk perintah yang semakin bertambah banyak. Cara yang mudah menggunkaan nomor pada perintah history atau mencarinya. Untuk memilih dan mengeksekusi perintah dengan nomor, masukkan kunci ! diikuti nomor perintah.

$ *!<Nomor Perintah>* Contoh : !780

1. Anda dapat mencari perintah dengan menyertakan perintah yang diinginkan. Misalnya !?etc?! akan menjalankan perintah ls –l /etc yang sebelumnya digunakan.

$ *!?etc?*

1. Kemudian gunakan perintah history, maka akan terlihat perintah ls –l

/etc yang kedua dan bukan !?etc?

$ *history*

1. Apabila string tidka ditemukan pada perintha history maka akan terdapat pesan error.

$ *!?wombat99?*

1. Jika diketikkan !who maka yang dijalankan adalah perintah who. Tetapi bila Anda ketikkan !whoa maka yang dijalankan adalah perintah whoami.

$ *!who*

$ *!whoa*

1. Anda bisa menggantikant string pada perintah history, terutama pada perintah yang panjang. Misalnya ketik cat /bin/bash | strings | grep shell | less dan tekan Enter. Maka akan menampilkan semua string pada file /bin/bash yang berisi kata ”shell”. Untuk keluar tekan q. Jika ingin menampilkan kata ”alias”, maka Anda tidak perlu mengetik perintah yang panjang lagi, tetapi cukup ketik ^shell^alias^ dan tekan Enter maka akan menggantikan kata ”shell” dengan ”alias”.

$ *cat /bin/bash | strings | grep shell | less*

$ *^shell^alias^*

Percobaan 3 : Mengubah Feature History Bash

1. Bash shell akan menyimpan perintah history meskipun telah log out dan log in kembali. File .bash\_history menyimpan file history yang terdapat pada home directory.

$ *cd*

1. Lihat beberapa baris pada file .bash\_history dengan ketik tail .bash\_history dan tekan Enter. File ini bukan file yang up to date.

$ *tail .bash\_history*

1. Ketik history dan tekan Enter. Maka akan terlihat baris terakhir adalah perintah history dan baris sebelumnya adalah tail .bash\_history. Perintah history bersifat up to date, karena disimpan pada memory sistem.

$ *history*

1. Ketik perintah berikut

$ *echo ‘Ini perintah saya’*

1. Log out dan log in kembali sebagai user yang sama. Ketik history dan tekan Enter. Maka perintah echo ’Ini perintah saya’ akan berada pada baris terakhir. Lihat file .bash\_history, maka perintah tsb akan terdapat pada file .bash\_history.

$ *history*

$ *tail .bash\_history*

1. Ketik history|less untuk melihat perintah history terakhir pada screen. Tekan spacebar untuk melihat file lebih banyak. Untuk keluar tekan q

$ *history|less*

1. Untuk melihat berapa banyak perintah history yang ada pada file ketik berikut dan output yang keluar serupa di bawah ini

$ *wc –l .bash\_history*

1000 .bash\_history

1. Output menunjukkan bahwa 1000 perintah history disimpan pada file history. Untuk melihat jangkauan (limit) perintah history digunakan variabel HISTSIZE. Untuk melihat jangkauan history ketik sebagai berikut

$ *set|grep HISTSIZE*

1. Bila ingin memperbesar jangkauan file history, maka ubahlah variabel HISTSIZE pada skrip startup yang disebut .bashrc pada home directory.

$ *echo ‘HISTSIZE=5000’ >> .bashrc*

1. Log out dan log in kembali sebagai user yang sama. Lihat perubahan variabel HISTSIZE.

$ *set|grep HISTSIZE*

1. Ketikkan perintah history beberapa kali, maka perintah ini akan disimpan pada BASH history meskipun yang diketikkan perintahnya sama.
2. Anda dapat melakukan konfigurasi BASH agar tidak menambah perintah ke history jika perintah yang diketikkan sama dengan sebelumnya. Hal ini dilakukan dengan menambahkan variabel HISTCONTROL dan diberikan nilai ignoredups pada file .bashrc

$ *echo ‘HISTCONTROL=ignoredups’ >> .bashrc*

1. Log out dan log in kembali sebagai user yang sama . Ketikkan history beberapa kali dan perhatikan berapa kali history muncul.