

Bab 5

Bash scripting

5.1 Shell

Shell adalah "**Command Executive**" artinya program yang menunggu instruksi user, memeriksa sintaks dan menerjemahkan instruksi yang diberikan kemudian mengeksekusinya. Pada umumnya shell ditandai dengan command prompt, di Linux untuk user biasa biasanya tanda \$ dan untuk super user biasanya tanda #. Shell ada bermacam-macam, di Linux biasa digunakan bash.

5.2 File permission

Perizinan file dan direktori dibagi atas 3 macam akses, antara lain :

- **READ** (r). Membaca file atau direktori
- **WRITE** (w). Menulis dan mencipta file atau direktori
- **EXECUTE** (x). Mengeksekusi file atau memasuki direktori

Kepemilikan file dan direktori dibagi atas 3 macam kepemilikan, antara lain:

- **Owner** (u), yaitu user tertentu.
- **Group** (g), yaitu group pemilik.
- **Others** (o) selain Owner dan Group di atas.

Untuk mengubah perizinan file atau direktori ini, gunakan perintah :

```
# chmod [ugoa] [= + -] [rwx] file_atau_direktori
```

atau

```
# chmod [angka_perizinan] file_atau_direktori
```

Keterangan :

u : user
g : group
o : other
a : all

= : set sebagai satu-satunya izin yang dimiliki
+ : penambahan izin
- : non-aktifkan suatu izin
r : akses read
w : akses write
x : akses execute
angka_perizinan : Owner-group-others dengan akses `rw-rw-rw`, tiap akses dimisalkan dengan bit '1' bila diberi akses dan bit '0' bila tidak diberi akses.
Contoh :

- Bila owner diberi seluruh akses, group hanya baca, others tidak ada akses sama sekali, maka angka_perizinan nya : 111-100-000 biner = 7-4-0 desimal ditulis 740
- Bila owner diberi seluruh akses, group baca-execute, others baca-execute, maka angka_perizinan nya : 111-101-101 biner = 7-5-5 desimal ditulis 755

5.3 File ownership

Kepemilikan file dan direktori dapat diubah dengan perintah :

```
# chown [-R] user[.group] file_atau_direktori
```

Keterangan :

- R : parameter ini menandakan perubahan rekursif untuk direktori
- user : user pemilik baru file atau direktori
- group : group pemilik baru file atau direktori

5.4 Akses eksekusi

Agar file atau script dapat dieksekusi maka file atau script tersebut harus memiliki izin `execute`. Mengubah akses suatu file menjadi `+x` atau bit `execute` aktif dapat dilakukan dengan perintah `chmod` diatas.

5.5 History

History diadaptasi dari **C-shell** (csh), yaitu pencatatan dari semua instruksi yang telah dilakukan. History dapat dipilih kembali dan perintah yang dipilih dapat dijalankan kembali. Variabel yang berkenaan dengan besar history sistem adalah variabel `HISTSIZE` yang di-set dalam system wide environment (/etc/profile).

Bila anda bekerja dengan prompt shell, gunakan panah atas dan bawah untuk memanggil instruksi yang telah masuk history.

5.6 Prompt string

Prompt String adalah kumpulan karakter yang setelah direpresentasikan oleh shell (/bin/bash) membentuk tampilan prompt yang siap menunggu instruksi. Untuk membentuk prompt string lakukan perintah berikut :

```
# PS1='\u@\h \w $ '
# export PS1
```

Keterangan :

PS1 : adalah variabel prompt string pertama
\u : menunjuk pada nama user
@ : karakter @
\h : menunjuk nama host
\w : menunjuk working directory

Alias

Alias adalah satu mekanisme untuk memberi nama alias pada satu atau sekelompok instruksi.

Sintaks alias adalah :

```
# alias
# alias NamaAlias='Kumpulan instruksi ?'
```

Contoh penggunaan alias :

```
# alias la='ls -la'
```

File Descriptor

Linux berkomunikasi dengan file melalui file descriptor yang direpresentasikan melalui angka yang dimulai dari 0,1,2 dan seterusnya.

3 buah file descriptor standar adalah :

- keyboard (standard input)
- layar monitor (standard output)
- layar monitor untuk tampilan error (standard error)

Dengan memanfaatkan pengalihan (redirection), kita dapat memisahkan output sebenarnya dan output error pada file berbeda.

Contoh :

```
# find / 1> finddb 2> /var/log/finddb.error
```

dapat ditulis :

```
# find / > finddb 2> /var/log/finddb.error
```

5.7 Flow control

Blok IF

```
if [ kondisi ]; then
else
fi
```

Blok CASE

```
case "$variabel" in
kondisi1)
perintah1
;;
kondisi2)
perintah2
;;
*)
perintah_lainnya
exit 1
esac
```

Blok FOR

```
for variabel in kelompok_item ; do
perintah $variabel
done
```

Blok WHILE

```
while kondisi ; do perintah ; done
until kondisi ; do perintah ; done
```

5.8 Prosedur

Kelompok perintah dapat dikumpulkan sehingga dengan memanggil suatu prosedur tertentu dapat melakukan beberapa perintah berurut. Sintaks :

```
NamaProsedur() {
Perintah1
Perintah2
?
}
```