

Shell Scripting

Praktikum Sistem Operasi

Ilmu Komputer IPB

2017

Shell Scripting

- ▶ menyimpan perintah *shell* ke dalam suatu *file*
- ▶ fitur pemrograman: variabel, kontrol aliran, fungsi
- ▶ berguna untuk:
 - ▶ pemrosesan teks
 - ▶ otomatisasi administrasi sistem

Contoh

```
$ cat > hello
#!/bin/sh
echo "Hello world"
exit 0
^D
```

```
$ chmod +x hello
```

```
$ ./hello
Hello world
```

Referensi

- ▶ `man sh`
- ▶ <http://wiki.bash-hackers.org>
- ▶ <http://www.commandlinefu.com>

Ekspansi

Pathname Expansion

- ▶ *: nol atau lebih karakter apapun
- ▶ ?: tepat satu karakter apapun
- ▶ [...]: tepat satu karakter di dalam *range*
- ▶ [!...]: tepat satu karakter selain dalam *range*

Pathname Expansion

```
ls /bin/e*  
# /bin/echo /bin/ed /bin/egrep
```

```
ls /bin/e?  
# /bin/ed
```

```
ls /bin/[cde]?  
# /bin/cp /bin/dd /bin/df /bin/ed
```

```
ls /bin/[!a-n]?  
# /bin/ps /bin/rm /bin/sh /bin/ss /bin/su
```

Parameter Expansion

```
web="cs.ipb.ac.id"
```

```
echo $web
```

```
# cs.ipb.ac.id
```

```
echo ${#web}
```

```
# 12
```

```
echo ${web%.ac.id}
```

```
# cs.ipb
```

```
echo ${web#cs.}
```

```
# ipb.ac.id
```


Command Substitution

`$(...)`

Menjalankan perintah dan mengembalikan keluarannya.

```
echo "I am $(whoami)."  
# I am root.
```

```
echo "Today is $(date +%A)."  
# Today is Tuesday.
```

Arithmetic Expansion

```
$((...))
```

Mengevaluasi ekspresi aritmatika dan mengembalikan keluarannya.

```
x=5
```

```
y=3
```

```
echo $((x + y)) $((x * y)) $((x / y)) $((x % y))
```

```
# 8 15 1 2
```

```
echo $((x > y)) $((x == y)) $((x > y && y > 0))
```

```
# 1 0 1
```

Parameter Khusus

- ▶ `$@`: semua parameter
 - ▶ `$1`: parameter pertama
 - ▶ `$2`: parameter kedua
 - ▶ ...
- ▶ `$#`: jumlah semua parameter
- ▶ `$?`: status keluaran perintah terakhir
- ▶ `$$`: PID proses *shell*

Ekspresi

[

Mengecek *file* dan membandingkan nilai. Status keluaran berupa 0 (*true*) atau 1 (*false*).

[*EXPR*]

- ▶ *-f* *FILE*: *file* biasa?
- ▶ *-d* *FILE*: *file* direktori?
- ▶ *-r* *FILE*: *file* bisa dibaca?
- ▶ *-w* *FILE*: *file* bisa ditulis?
- ▶ *STR1* = *STR2*: kedua *string* sama?
- ▶ *INT1* *-eq* *INT2*: kedua angka sama?
- ▶ *INT1* *-lt* *INT2*: lebih kecil?
- ▶ *INT1* *-gt* *INT2*: lebih besar?

[

```
[ "hello" ]; echo $?  
# 0 (true)
```

```
[ -r /etc/passwd ]; echo $?  
# 0 (true)
```

```
[ -r /etc/passwd ] && [ ! -w /etc/passwd ]; echo $?  
# 0 (true)
```

```
[ "hello" = "world" ]; echo $?  
# 1 (false)
```

```
[ 3 -gt 2 ]; echo $?  
# 0 (true)
```

seq

Mencetak sekuens angka, berguna untuk *looping*.

```
seq [FIRST [INCREMENT]] LAST
```

```
seq 10
```

```
# 1 2 3 4 5 6 7 8 9 10
```

```
seq 1 2 10
```

```
# 1 3 5 7 9
```

read

Membaca satu baris masukan.

```
read NAME...
```

```
read input  
# <ketikkan: hello>
```

```
echo $input  
# hello
```


Kontrol Aliran

Percabangan

```
if ...  
    then ...  
elif ...  
    then ...  
else  
    ...  
fi
```

Kasus

```
case WORD in  
    PATTERN)  
        ... ;;  
esac
```

Perulangan

```
for NAME in WORDS
    do ...
done
```

Perulangan

```
while ...  
    do ...  
done
```

```
until ...  
    do ...  
done
```

Fungsi

```
NAME () {  
    ...  
}
```

Contoh

Percabangan

```
#!/bin/sh
# create public_html directory if not exist

webdir=~/public_html

if [ ! -d $webdir ]; then
    mkdir $webdir
fi

exit 0
```


Kasus

```
#!/bin/sh
# is today weekend?

case $(date +%a) in
    Sat|Sun)
        echo "weekend";;
    *)
        echo "weekday";;
esac
```

Perulangan

```
#!/bin/sh
# cube from 1 to 10

for i in $(seq 10); do
    echo $((i*i*i))
done

exit 0
```

Fungsi

```
#!/bin/sh  
# Caesar cipher (ROT13)
```

```
rot13 () {  
    tr A-Z N-ZA-M  
}
```

```
uppercase () {  
    tr a-z A-Z  
}
```

```
uppercase | rot13
```

```
exit 0
```

Perulangan dan Percabangan

```
#!/bin/sh
# integer divisible by 3 or 5 between 1-100

count=0
for i in $(seq 100); do
    if [  $((i\%3 == 0 \mid i\%5 == 0))$  -eq 1 ]; then
        count=$((count + 1))
    fi
done
echo $count
```

Latihan

Frekuensi Kata Terbanyak

Buatlah program “topwords” untuk mencetak 5 kata dengan frekuensi terbanyak dari masukan stdin!

```
$ man ls | ./topwords
21 of
20 sort
19 by
18 the
16 with
```

Contoh *pipeline*

1. ubah ke *lowercase*
2. jadikan satu kata satu baris
3. urutkan
4. hitung kemunculan kata yang sama
5. urutkan berdasarkan angka secara menurun
6. ambil 5 baris teratas

Identifikasi Penyerang

Server NCC diserang dari luar melalui *port* SSH. Cek isi *log file* `auth.log.gz`¹ dan identifikasi *n* alamat IP penyerang terbanyak!

```
$ ./top-ip-attack auth.log.gz 3
116.31.116.52    3859
221.194.47.229  1882
221.194.47.208  1819
```

¹<https://lms.ipb.ac.id/mod/folder/view.php?id=28806>

Contoh *pipeline*

1. tampilkan *file* log dengan `zcat`
2. ambil baris yang mengandung kata '`Failed`'
3. ekstrak alamat IP tiap baris
4. urutkan
5. hitung kemunculan alamat IP yang sama
6. urutkan berdasarkan angka secara menurun
7. ambil *n* baris teratas

Cek Tugas Email

Bantulah asprak SO untuk mengidentifikasi praktikan yang belum mengumpulkan tugas email. Diberikan *file* `auriza.mbox2`, ambil nama *user*-nya, kemudian bandingkan dengan daftar *user* pada *file* `passwd3`!

```
$ ./check-no-mail  
anni  
michaeln  
...
```

²<https://lms.ipb.ac.id/mod/folder/view.php?id=28806>

³<https://lms.ipb.ac.id/mod/folder/view.php?id=28806>

Contoh *pipeline*

1. Daftar pengirim email

- 1.1 tampilkan *file* kotak surat
- 1.2 ambil baris yang mengandung kata 'From '
- 1.3 ekstrak nama *user*
- 1.4 urutkan
- 1.5 hilangkan nama yang berulang

2. Daftar semua *user*

- 2.1 tampilkan *file* /etc/passwd
- 2.2 ambil baris yang mengandung kata 'G64'
- 2.3 ekstrak nama *user*
- 2.4 urutkan

3. Cari bedanya

- 3.1 bandingkan keluaran dua *pipeline* di atas dengan `diff`
- 3.2 sesuaikan format sesuai keluaran yang diminta

Tugas: *Spelling Checker*⁶

Buatlah program untuk mengecek kesalahan ejaan pada dokumen bahasa Inggris. Gunakan daftar kata pada *file words*⁴ untuk membandingkan ejaan. Lihat video berikut⁵ untuk bantuan *pipeline*-nya. Kumpulkan jawaban di LMS.

```
$ ./myspell sentence  
laborotories  
priveide  
timesharing  
unix
```

⁴ `/usr/share/dict/words`

⁵ <https://youtu.be/tc4ROCJYbm0?t=5m58s>

⁶ tugas bersifat **opsional**, plagiasi akan mendapat sanksi berat.