

Jobsheet 7

Bekerja dengan Bash Shell

Nama: Rizqi Bagus Andrean

Kelas: TI-1D

Absen: 25

Pokok Bahasan:

- History pada Bash Shell
- Membuat Bash Shell Script

Tujuan Belajar:

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 sebuah program yang membaca input standar yaitu keyboard. Ketika kita mengetikkan sesuatu di keyboard maka shell akan menafsirkan apa yang kita ketikkan. Sebagai contoh apabila kita ketikkan `ls` pada shell dan `ls` ternyata merupakan program yang dapat di eksekusi maka shell akan menjalankan program tersebut.

Misal :

```
antoro@drutz:~$ ls
test
antoro@drutz:~$ hello
bash: hello: command not found
antoro@drutz:~$
```

Kita dapat melihat bahwa ketika `ls` kita ketikkan maka program `ls` dijalankan, sedangkan apabila kita jalankan `hello` dan karena memang tidak ada program `hello` maka shell tidak dapat mengeksekusinya.

Jenis-Jenis Shell

Jenis shell yang digunakan di unix maupun linux banyak macamnya, tetapi dilihat dari segi kuantitas pemakainya shell yang banyak dipakai adalah sebagai berikut :

- Bourne shell (sh)
- C shell (csh)
- Korn shell (ksh)
- Bourne Again shell (bash)

Yang paling banyak di gunakan adalah bash, dimana bash ini merupakan pengembangan dari sh dan ksh. Mayoritas distribusi linux saat ini menggunakan bash sebagai default shell-nya.

2. PROFILE

Pada saat kita login dan memulai sebuah session baru bash shell, maka bash akan menjalankan file /etc/profile, .profile, /etc/bash.bashrc .bashrc. File /etc/profile dan /etc/ bash.bashrc digunakan untuk semua user yang ada, jika kita melakukan modifikasi file ini maka akan ber-efek pada semua user yang bisa login kedalam system. Sedangkan .profile dan .bashrc hanya terdapat pada masing-masing home direktori dari user yang bersangkutan. File /etc/profile dan .profile hanya akan dieksekusi oleh login shell, jika dalam session aktif kita menjalankan perintah

```
$ bash
```

Maka file .profile tidak akan dijalankan, agar .profile dijalankan kita harus menggunakan perintah.

```
$ bash --login
```

Perintah ini berguna untuk melihat hasil dari file .profile yang dimodifikasi tanpa harus logout dari session. Sekarang kita akan mencoba memodifikasi file .profile, buka dengan teks editor, dan tambahkan baris berikut dibagian paling bawah

```
date
```

Simpan hasil perubahan tersebut, kemudian jalankan perintah berikut

```
$ bash --login
```

```
Tue Nov 13 20:55:13 WIT 2007
```

Untuk keluar ketik perintah

```
$ logout
```

Modifikasi seperti merubah prompt shell secara permanen, mode editing yang digunakan oleh shell bisa ditempatkan pada file .profile. Untuk interaktif session yang bukan login session, maka bash akan melihat pada file /etc/bash.bashrc dan .bashrc. Pada .bashrc, umumnya digunakan untuk menempatkan sebuah fungsi alias bagi sebuah command, misalnya alias ls="ls -qF". Pada distro Ubuntu, file .bashrc ini juga dijalankan pada waktu login session, hal ini bisa dilihat pada file .profile di Ubuntu terdapat baris berikut

```
# if running bash
if [ -n "$BASH_VERSION" ]; then
    # include .bashrc if it exists
    if [ -f ~/.bashrc ]; then
```

```
        . ~/.bashrc
    fi
fi
```

.bash_logout merupakan file yang dijalankan pada saat kita logout dari session bash shell. Jika kita ingin menghapus isi direktori tertentu pada saat logout, maka perintah tersebut harus ditempatkan pada file .bash_logout. Sedangkan .bash_history berisikan command-command yang pernah kita input pada shell.

3. HISTORY

Semua perintah-perintah yang sudah pernah anda ketikkan, akan disimpan ke dalam history. Untuk menampilkan anda bisa menggunakan perintah history. Secara default perintah-perintah yang bisa ditampung dalam history adalah 1000 perintah, untuk konfigurasi berada pada /home/user/.bashrc pada baris "export HISTSIZE=1000". Penyimpanan file history biasanya disimpan secara default berada pada /home/user/ dengan nama file .bash_history (jika menggunakan shell bash), .zsh_history(jika menggunakan shell zsh).

4. 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.

5. 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.

6. EDITOR vi/vim

vi merupakan teks editor yang pertama kalinya dibuat untuk UNIX, dimana teks editor ini sangat sederhana, memiliki ukuran yang kecil tapi mempunyai fungsionalitas yang besar. Saat ini banyak versi dari vi yang ada seperti nvi, vim, vile, elvis. Pada GNU/Linux umumnya menggunakan vim yang diciptakan oleh Bram Moolenaar's berdasarkan vi. Pada Ubuntu saat kita mengetikkan vi atau vim pada shell, maka program yang dijalankan adalah sama yaitu vim. VIM merupakan singkatan dari VI Improved, dimana vim memiliki fitur seperti multiple undo yang tidak terdapat pada vi versi lama. Untuk memulai vi/vim pada shell (jika menggunakan GUI, jalankan gnometerminal) bisa dengan cara

```
vi
vi options file
vi options +num file
```

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

- Command line

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

- Editing

Editor vi menginterpretasikan input sebagai teks yang akan dimasukkan kedalam 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.

TUGAS PENDAHULUAN:

Jawablah pertanyaan-pertanyaan di bawah ini :

1. Apa yang dimaksud dengan shell dan sebutkan shell yang ada di system operasi linux.

Jawab :

Shell adalah Command executive, artinya program yang menunggu intruksi dari pemakai, memeriksa sintak dari instruksi yang diberikan, kemudian mengeksekusi perintah tersebut.

- /bin/sh : Bourno shell, dirancang oleh Steve Bourne dari AT&T
- /bin/csh : Dikembangkan oleh UNIX Beerkeley yang dikenal dengan C –Shell
- /bin/bash : Kompatibel dengan Bourne Shell dan juga mengadaptasi kemampuan Kom-Shell.

2. Apa yang dimaksud dengan profile pada bash shell

Jawab : Profile pada bash shell merupakan profil untuk setiap pemakai pada home directory.

3. Apa yang anda ketahui mengenai file .bashtrc.

Jawab : File .bashrc adalah file yang akan dieksekusi untuk perpindahan dari satu shell ke shell yang lain melalui intruksi su.

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!

Jawab :

History yaitu catatan dari semua intruksi yang sejauh ini telah dilakukan. History memudahkan pemakai untuk mengedit kembali instruksi kompleks dan panjang, terutama bila terjadi kesalahan pada penulisan instruksi maupun parameter

^P (Ctrl -P) melihat instruksi

^N (Ctrl -N) melihat instruksi berikutnya

```
!! eksekusi kembali instruksi sebelumnya
!! -3 3 instruksi sebelumnya akan diulang
!!88 ulangi instruksi no 88
```

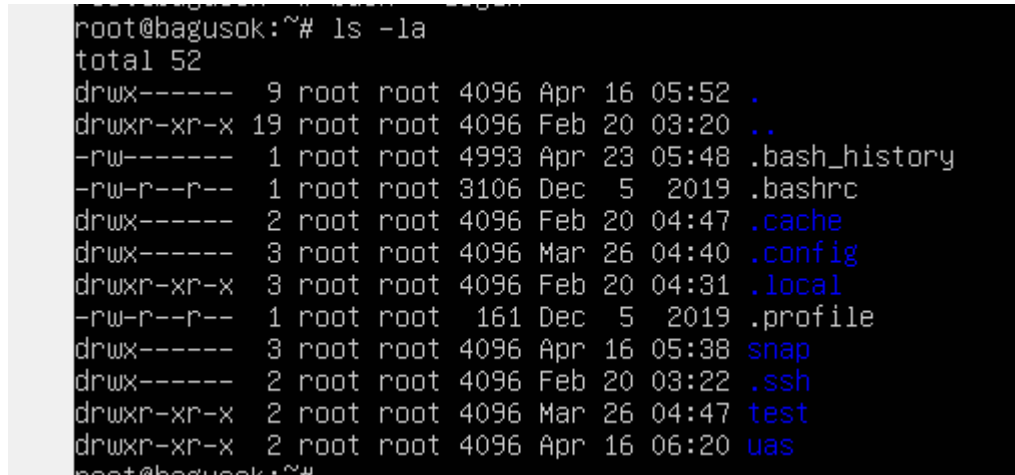
PERCOBAAN:

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 : Status Proses

1. File `.bash_profile` dijalankan pada home direktori pemakai yang login. File `.bash_profile` adalah hidden file ,sehingga untuk melihatnya digunakan opsi `-a` pada instruksi `ls`.

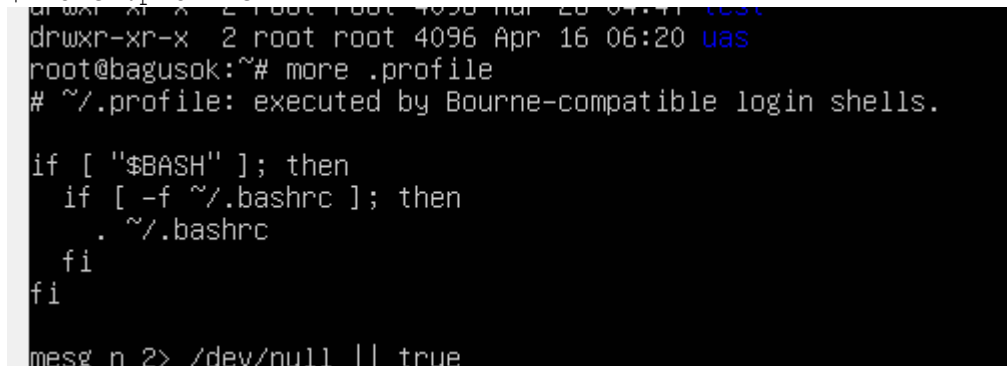
```
$ ls -a
```



```
root@bagusok:~# ls -la
total 52
drwx-----  9 root root 4096 Apr 16 05:52 .
drwxr-xr-x 19 root root 4096 Feb 20 03:20 ..
-rw-----  1 root root 4993 Apr 23 05:48 .bash_history
-rw-r--r--  1 root root 3106 Dec  5 2019 .bashrc
drwx-----  2 root root 4096 Feb 20 04:47 .cache
drwx-----  3 root root 4096 Mar 26 04:40 .config
drwxr-xr-x  3 root root 4096 Feb 20 04:31 .local
-rw-r--r--  1 root root  161 Dec  5 2019 .profile
drwx-----  3 root root 4096 Apr 16 05:38 .snap
drwx-----  2 root root 4096 Feb 20 03:22 .ssh
drwxr-xr-x  2 root root 4096 Mar 26 04:47 .test
drwxr-xr-x  2 root root 4096 Apr 16 06:20 .uas
root@bagusok:~#
```

Analisis: Gambar di atas merupakan cara untuk menampilkan direktori secara keseluruhan. Perintah `ls -a` berfungsi untuk menampilkan file `.bash_profile` (file yang tersembunyi) pada home direktori pemakai yang login.

```
$ more .profile
```



```
drwxr-xr-x  2 root root 4096 Apr 16 06:20 .uas
root@bagusok:~# more .profile
# ~/.profile: executed by Bourne-compatible login shells.

if [ "$BASH" ]; then
  if [ -f ~/.bashrc ]; then
    . ~/.bashrc
  fi
fi

mesg n 2> /dev/null || true
```

Analisis: Gambar di atas merupakan perintah untuk menampilkan isi dari `.profile` (file yang tersembunyi) pada home direktori pemakai yang login.

2. File `.bash_logout` akan dieksekusi 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
```

Analisis: Gambar di atas merupakan perintah untuk menampilkan `.bash_logout`. Perintah ini akan dieksekusi 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 mengakses history dalam beberapa cara. Cara paling mudah adalah menggunakan tombol 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
```

Analisis: Gambar di atas merupakan tampilan menggunakan Feature History Bash. Perintah `$ cd` berfungsi untuk memastikan sedang berada pada direktori home. Perintah `$ ls -l /etc` berfungsi untuk menampilkan output file atau direktori secara lengkap yaitu terdiri dari nama file, ukuran, tanggal dimodifikasi, pemilik, group dan mode atau atributnya.

```
$ ls -l
```

Analisis: Gambar di atas merupakan perintah menggunakan Feature History Bash. Perintah `$ ls -l` berfungsi untuk menampilkan daftar file pada direktori yang sedang aktif yaitu direktori home.

```
$ whoami
$ who
```

Analisis: Gambar di atas merupakan perintah menggunakan Feature History Bash. Perintah `$ whoami` dan `$ who` berfungsi untuk mengetahui siapa saja yang sedang aktif.

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

```
$ history
```

```

365 kill %1
366 kill 1
367 jobs
368 fg
369 fg
370 bg
371 yes &
372 fg %2
373 bg %2
374 ps -fae
375 kill -9 1
376 PS1="\w:"
377 pwd
378 top
379 renice
380 ls -l
381 rm -rf *
382 ls
383 yes
384 vat yes
385 cat yes
386 yes
387 ipconfig
388 ifconfig
389 logout
390 ifconfig
391 ping 8.8.8.8
392 ifconfig /release
393 ifconfig -h
394 ip a
395 restart
396 reboot
397 ls -la
398 more .profile
399 more .bash_logout
400 history
root@bagusok:~# _

```

Analisis: Gambar di atas merupakan perintah menggunakan Feature History Bash. Untuk memeriksa apakah perintah ini ditambahkan pada history, dapat menggunakan perintah \$history untuk melihat semua perintah yang pernah dimasukan.

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

\$!<Nomor Perintah> Contoh : !1000

```

400 history
root@bagusok:~# !397
ls -la
total 52
drwx----- 9 root root 4096 Apr 16 05:52 .
drwxr-xr-x 19 root root 4096 Feb 20 03:20 ..
-rw----- 1 root root 4993 Apr 23 05:48 .bash_history
-rw-r--r-- 1 root root 3106 Dec 5 2019 .bashrc
drwx----- 2 root root 4096 Feb 20 04:47 .cache
drwx----- 3 root root 4096 Mar 26 04:40 .config
drwxr-xr-x 3 root root 4096 Feb 20 04:31 .local
-rw-r--r-- 1 root root 161 Dec 5 2019 .profile
drwx----- 3 root root 4096 Apr 16 05:38 snap
drwx----- 2 root root 4096 Feb 20 03:22 .ssh
drwxr-xr-x 2 root root 4096 Mar 26 04:47 test
drwxr-xr-x 2 root root 4096 Apr 16 06:20 uas

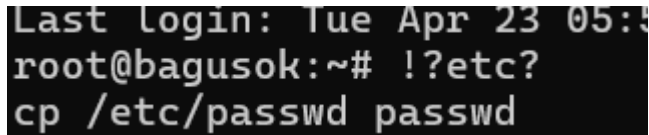
```

Analisis: Gambar di atas merupakan perintah menggunakan Feature History Bash. Dapat memilih perintah sebelumnya dengan menggunakan Panah Atas, tetapi hal ini tidak efisien untuk perintah yang semakin bertambah banyak. Cara yang mudah menggunakan nomor pada perintah history atau mencarinya. Untuk memilih dan mengeksekusi perintah dengan nomor, masukan kunci! Diikuti

nomor perintah, maka akan muncul output seperti gambar di atas. Contoh : !1000 setelah menginputkan perintah tersebut maka perintah who terpanggil lagi.

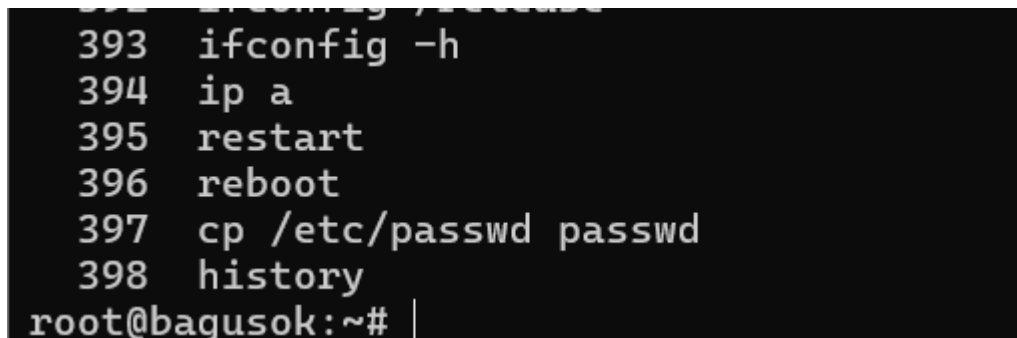
5. Anda dapat mencari perintah dengan menyertakan perintah yang diinginkan. Misalnya !?etc? akan menjalankan perintah ls -l /etc yang sebelumnya digunakan.

```
$ !?etc?
```



```
Last login: Tue Apr 23 05:3
root@bagusok:~# !?etc?
cp /etc/passwd passwd
```

Analisis: Gambar di atas merupakan perintah menggunakan Feature History Bash. Dapat mencari perintah dengan menyertakan perintah yang diinginkan. Misalnya !?etc? akan menjalankan perintah ls -l /etc yang sebelumnya digunakan.



```
393 ifconfig -h
394 ip a
395 restart
396 reboot
397 cp /etc/passwd passwd
398 history
root@bagusok:~# |
```

6. Kemudian gunakan perintah history, maka akan terlihat perintah ls -l etc yang kedua dan bukan !?etc?
- ```
$ history
```

Analisis: Gambar di atas merupakan perintah menggunakan Feature History Bash. Gunakan perintah history, maka akan terlihat perintah ls -l etc yang kedua dan bukan !?etc?.

7. Apabila string tidak ditemukan pada perintah history maka akan terdapat pesan error

```
$!?wombat99?
```

```
root@bagusok:~# !?wombat99?
-bash: !?wombat99?: event not found
root@bagusok:~# |
```

Analisis: Gambar di atas merupakan perintah menggunakan Feature History Bash. Apabila string tidak ditemukan pada perintah history maka akan terdapat pesan error dengan menggunakan perintah \$ !?wombat99?, maka akan muncul output error karena tidak diketemukan.

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

```
$!who
```

```
-bash: !?wombat99?: event not found
root@bagusok:~# !who
who am i
root pts/0 2024-04-23 06:06 (192.168.64.185)
root@bagusok:~# |
```

```
$!whoa
```

Analisis: Gambar di atas merupakan perintah menggunakan Feature History Bash. Jika diketikkan !who maka yang dijalankan adalah perintah who. Tetapi bila Anda ketikkan !whoa maka yang dijalankan adalah perintah whoami.

9. Anda bisa menggantikan 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
```

Analisis: Gambar di atas merupakan perintah menggunakan Feature History Bash. Bisa menggantikan 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. Akan tetapi perintah ini gagal karena paket bunitils pada oprasi sistem linux yang digunakan belum terpasang.

### Percobaan 3 : Mengubah Feature History Bash

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

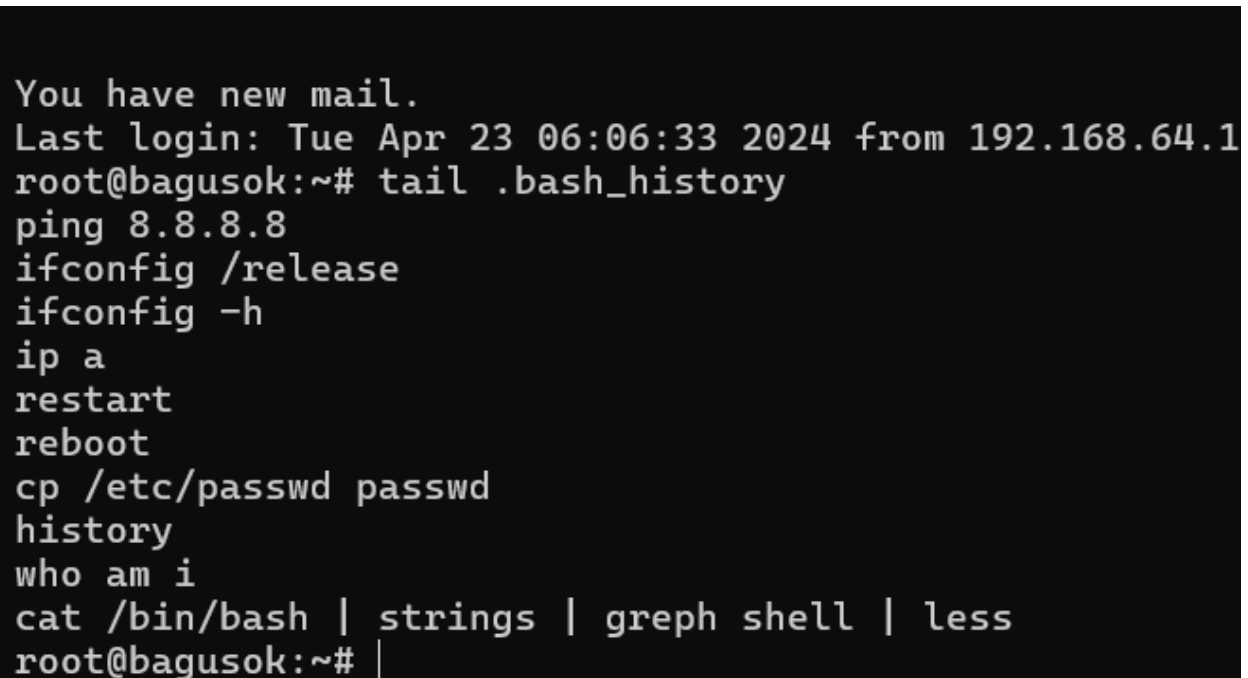
```
$ cd
```

Analisis: Gambar di atas merupakan perintah untuk memastikan berada di direktori home.

Bash shell akan menyimpan perintah history meskipun telah logout dan login kembali. File `.bash_history` menyimpan file history yang terdapat pada home directory dengan menggunakan perintah `$ cd`.

2. 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
```



```
You have new mail.
Last login: Tue Apr 23 06:06:33 2024 from 192.168.64.1
root@bagusok:~# tail .bash_history
ping 8.8.8.8
ifconfig /release
ifconfig -h
ip a
restart
reboot
cp /etc/passwd passwd
history
who am i
cat /bin/bash | strings | grep shell | less
root@bagusok:~# |
```

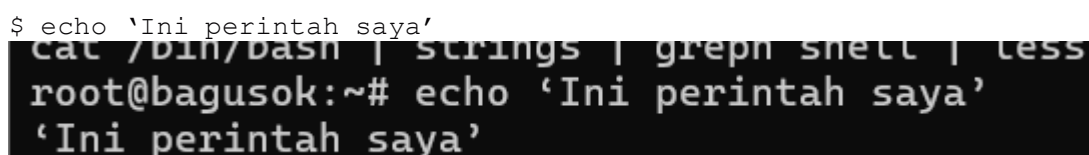
Analisis: Gambar di atas merupakan perintah mengubah Feature History Bash. Lihat beberapa baris pada file `.bash_history` dengan ketik `tail .bash_history` dan tekan Enter.

3. 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
```

Analisis: Gambar di atas merupakan perintah mengubah Feature History Bash. 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.

4. Ketik perintah berikut



```
$ echo 'Ini perintah saya'
root@bagusok:~# echo 'Ini perintah saya'
'Ini perintah saya'
```

Analisis: Gambar di atas merupakan perintah mengubah Feature History Bash. Ketikkan perintah `echo 'Ini perintah saya'`, maka akan muncul tulisan atau output `Ini perintah saya`.

5. Logout dan login 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 tersebut akan terdapat pada file .bash\_history.

```
$ history
```

```
395 restart
396 reboot
397 cp /etc/passwd passwd
398 history
399 who am i
400 cat /bin/bash | strings | greph shell | less
401 tail .bash_history
402 echo 'Ini perintah saya'
403 history
root@bagusok:~#
```

```
$ tail .bash_history
```

```
ping 8.8.8.8
ifconfig /release
ifconfig -h
ip a
restart
reboot
cp /etc/passwd passwd
history
who am i
cat /bin/bash | strings | greph shell | less
root@bagusok:~#
```