

PERINTAH-PERINTAH DASAR LINUX

Pada saat pertama kali menggunakan UNIX / Linux, setiap user harus memasukan login dan password. Pada sistem operasi DOS, setiap user yang memakai komputer tidak pernah ditanya login dan password, karena DOS memang dirancang tidak untuk sistem multi-user. Sedangkan UNIX / Linux, sejak awal pengembangannya memang ditujukan untuk sistem multi-user sehingga dilengkapi dengan sistem keamanan yang handal.

1. login

fungsi : untuk masuk ke dalam jaringan

keterangan : Setiap pemakai sah dari sistem UNIX mempunyai identifikasi pemakai sendiri (ID).

2. password

Fungsi : Memasukkan kata sandi setelah login.

Keterangan : Untuk pemakai yang baru didaftar oleh SUPER USER maka user tidak perlu memasukkan kata sandi.

Untuk menjaga kerahasiaan, pengetikan tombol password tidak ditampilkan di layar.

```
Linux 2.0.0 (higo.eepis-its.ac-id.net) (ttyp2)
higo login: syarif
Password:
Last login: Mon Nov 23 15:10:29 from 10.252.1.2
Linux 2.0.0.
You have mail.

higo:~$
```

Jika seorang user tidak mempunyai hak atau memasukkan password yang keliru maka akan tampil pesan :

```
login: aku
Password:
Login incorrect
```

3. passwd

Fungsi : Untuk membuat atau mengubah kata sandi

```
$ passwd
Changing password for syarif
Enter old password:
Enter new password:
The password must have at least 6 characters, try again.
Enter new password:
```

Keterangan :

Seperti uraian keterangan di password, bila pertama kali log in ke sistem UNIX, harus dibuat kata sandi untuk diri sendiri. Harus dipikirkan suatu urutan huruf dan angka yang lebih panjang dari 6 karakter untuk kata sandi tersebut. Program password meminta pengetikan kata sandi sebanyak 2 kali. Bila ternyata kedua ketikan bukan

kata yang sama, akan diberikan kesempatan sekali lagi. Sekali diterapkan sebuah sandi, maka tidak dapat log in tanpa memberitahukannya pada UNIX, oleh karena itu password jangan sampai terlupa.

4. who

Fungsi : untuk mengetahui daftar pemakai yang sedang aktif (login)

```
$ who
root      tty1      Nov 19 15:34
sulis     tttyp1    Nov 23 14:34 (:0.0)
syarif    tttyp2    Nov 23 15:44 (10.252.1.201)
```

Keterangan : perintah who akan memberikan daftar pemakai yang baru saja log in dengan menunjukkan nama pemakai, terminal ID, dan waktu masing-masing log in.

5. finger

Fungsi : finger mempunyai kegunaan hampir sama dengan who, hanya saja finger menyediakan informasi identitas user yang lebih lengkap dari ada who.

Contoh perbedaan pemakaian who dan finger

```
$ who
nanang   tttyp1    Nov 26 14:59 (:0.0)
nanang   tttyp2    Nov 26 16:18 (:0.0)
sulis    tttyp3    Nov 26 17:03 (surat.eepis-its.)
syarif   tttyp4    Nov 26 18:01 (10.252.1.201)

$ finger
Login      Name                      Tty  Idle  Login Time
nanang    Nanang Syahroni          p1   2:17  Nov 26 14:59 (:0.0)
nanang    Nanang Syahroni          p2   2:11  Nov 26 16:18 (:0.0)
sulis    Sulistyo Mahargyo.B     p3    1    Nov 26 17:03 (surat.eepis-its.)
syarif   Iwan Syarif            p4                    Nov 26 18:01 (10.252.1.201)
```

6. logout

Fungsi : Untuk keluar dari sistem

Keterangan : bila pemakai akan mengakhiri penggunaan terminal sebaiknya menjalankan perintah ini, agar hak akses pada log in-nya tidak disalahgunakan oleh orang lain yang tidak berhak.

7. exit

Fungsi : Untuk keluar dari sistem

Keterangan : sama dengan perintah log out.

8. whoami

Fungsi : untuk mengetahui siapa user yang sedang login di suatu komputer / terminal
Keterangan : digunakan bila anda menemukan terminal yang belum logout atau exit dan ingin mengetahui milik siapa terminal tersebut diaktifkan.

9. date

Fungsi : Menunjukkan atau mengatur tanggal

```
$ date  
Mon Nov 23 15:55:04 JVT 1998
```

Keterangan : Melaporkan atau mengatur tanggal dan waktu sistem. Bila tidak ada argumen yang ditentukan, akan dilaporkan tanggal dan waktu sekarang.

10. cal

Fungsi : mencetak kalender mulai tahun 0000 s/d 9999

```
$ cal  
November 1998  
Su Mo Tu We Th Fr Sa  
1 2 3 4 5 6 7  
8 9 10 11 12 13 14  
15 16 17 18 19 20 21  
22 23 24 25 26 27 28  
29 30
```

Keterangan : membuat kalender bulan dan tahun yang ditentukan. Bila bulan tidak disertakan, maka dibuat kalender untuk 1 tahun.

11. ls

Fungsi : menampilkan daftar file dalam directory aktif.

Keterangan : Perintah ini akan menampilkan informasi mengenai directory dan file. Bentuk sederhana perintah ls akan menampilkan hanya nama file. Bentuk panjang ditandai dengan menggunakan option -l, yang akan menampilkan nama-nama file beserta informasi untuk setiap file yang ditampilkan.

Option-option yang disediakan :

- a tampilkan semua file di directory termasuk isian.
- o tampilkan hanya nama directory
- g cetak ID kelompok hanya untuk bentuk panjang
- I cetak nomor untuk setiap pemakai
- l tampilkan seluruh file secara lengkap
- o cetak ID pemakai (bila pemakai -l)
- r ubah urutan pengaturan nama file yang telah disusun
- t atur nama file berdasarkan waktu modifikasi terakhir, tidak berdasarkan nama
- o atur nama file berdasarkan waktu akses terakhir

Untuk option nama, bila “nama” merupakan sebuah directory, perintah itu mencetak informasi yang diminta dengan pilihan (option) bagi semua file dalam directory. Bila “nama” sebuah file, maka hanya informasi file bersangkutan yang dicetak.

Perbandingan perinta dir pada DOS dan perintah ls pada UNIX

Perintah DOS	Perintah UNIX	Keterangan
Dir	ls -l	Melihat daftar file secara detil
Dir/w	Ls	Melihat daftar file secara melebar
Dir/p	ls more	Melihat daftar file per halaman
Dir a*.*	ls a*.*	Melihat file dengan nama depan a
Dir > prn	ls > lpr	Mencetak daftar file ke printer
Dir > file1.txt	ls > file1.txt	Menyimpan daftar file ke “file1.txt”

Bentuk tampilan perintah ls

```
$ ls -l
-rw-r--r-- 1 syarif users 981 Sep 2 08:34 PIC22370.PCX
-rw-r--r-- 1 syarif users 981 Sep 2 08:33 PIC27480.PCX
-rw-r--r-- 1 syarif users 66091 Aug 12 12:14 bufferm.pdf
-rw-r--r-- 1 syarif users 171 Nov 23 19:51 coba1
-rw-r--r-- 1 syarif users 177 Nov 23 19:51 coba2
-rw-r--r-- 1 syarif users 171 Nov 23 19:52 coba3
drwx----- 2 syarif users 1024 Nov 19 11:55 nsmail/
drwx----- 2 syarif users 1024 Aug 12 11:34 syarif/
drwxr-xr-x 2 syarif users 1024 Aug 13 12:57 tmp/
-rw-r--r-- 1 syarif users 24093 Aug 12 12:17 tut1.1998.ps
-rw-r--r-- 1 syarif users 58527 Aug 12 12:15 tut1_2.pdf
-rw-r--r-- 1 syarif users 15687 Aug 12 12:17 tut2.1998.ps
-rw-r--r-- 1 syarif users 17984 Aug 12 12:18 tut3.1998.ps
$
```

Perintah **ls -l** sama dengan perintah **dir** pada DOS (seperti terlihat di atas), sedangkan perintah **ls** sama dengan perintah **dir/w** (seperti terlihat di bawah ini)

```
$ ls
PIC22370.PCX      dbms1.pdf          nsmail/        tut1.1998.ps
PIC27480.PCX      dead.letter       prakt_1.c      tut1_2.pdf
bufferm.pdf        introduction.pdf   server.c*    tut2.1998.ps
client.c           lpr                  syarif/       tut3.1998.ps
```

12. chmod

Fungsi : mengubah proteksi file

Format : chmod bilangan nama file

```
$ ls -l c*
-rw----- 1 syarif users 5935104 Aug 27 13:59 core
$ chmod 777 core
$ ls -l c*
-rwxrwxrwx 1 syarif users 5935104 Aug 27 13:59 core*
```

13. clear

Fungsi : bersihkan layar, (sama dengan perintah CLS di DOS)

Format : clear

14. cmp

Fungsi : bandingkan file1 dan file2 serta laporkan perbedaannya.

Format : cmp file1 dan file2

Keterangan : perintah ini tidak akan melaporkan apa-apa jika file tersebut identik (sama persis), tetapi jika ada perbedaan maka akan dilaporkan seperti berikut ini :

```
$ cmp coba1 coba2
cba1 coba2 differ: char 45, line 5
```

artinya file coba1 mempunyai perbedaan dengan file coba2 pada posisi 45 baris ke-5

15. cp

Fungsi : copy file1 menjadi file2

Format :

\$ cp file1 file2 ---- mengcopy file1 ke file2

\$ cp coba3 /home/syarif/nsmail ---- mengcopy file coba3 ke direktori lain

Keterangan : cp akan meng-copy satu file ke file lain atau meng-copy satu file atau lebih, ke sebuah direktori.

16. rm

Fungsi : menghapus file

Format : rm nama-file

Perintah DOS	Perintah UNIX	Keterangan
Del file1	rm -l	Menghapus file1
Del a.*.*	rm a.*.*	Menghapus file dengan nama depan a
Del file1 Del coba.txt Del buku.sam	rm file coba.txt buku.sam	Menghapus banyak file sekaligus

17. mv

Fungsi : memindahkan file1 menjadi file2

Format : mv file1 file2

\$ mv file1 file2 ---- mengcopy file1 ke file2

\$ mv coba3 /home/syarif/nsmail ---- mengcopy file coba3 ke direktori lain

Keterangan : mv akan memindahkan satu file ke file lain atau memindahkan satu file atau lebih, ke sebuah direktori

18. cat

Fungsi : menampilkan isi sebuah file (sama dengan perintah TYPE pada DOS)

Cat berfungsi untuk mencetak ke layar monitor isi dari sebuah file text. Jika yang file dilihat menggunakan perintah ini bukan file text maka akan keluar karakter-karakter aneh pada layar.. Untuk menghindari tercetaknya karakter-karakter aneh tersebut dapat digunakan perintah cat –v

```
$ cat server.c

#include <stdio.h>
main(argc,argv)
int argc;
char *argv[];
{
    if(argc < 2){
        fprintf(stderr, "Give me an argument, please. \n");
        exit(1);
}
```

19. more

Fungsi : menampilkan isi text file per layar

Format : more nama file

Contoh :

```
$ more server.c -----→ menampilkan isi file server.c
```

```
#include <stdio.h>
#include <sys/types.h>
{
    struct sockaddr_in server;
    static char buf[512], *mes = "I am fine !";
    if(argc < 2){
        fprintf(stderr, "Give me an argument, please. \n");
        exit(1);
    }
--More --(33%)
```

Keterangan : more menampilkan isi file dari titik tertentu kemudian menampilkan sisa file pada CRT sebanyak satu halaman setiap kali. Setiap layar CRT penuh maka penampilan file akan berhenti dan menunggu kita mengetikkan sebuah karakter atau return.

20. head

Fungsi : perintah head digunakan untuk menampilkan 10 baris pertama dari suatu file teks. Jika ingin menampilkan hanya 4 baris pertama maka digunakan perintah :

```
$ head -4 coba.txt
commands common to all logins
export OPENWINHOME=/usr/openwin
export MINICOM="-c on"
export win/man
```

21. tail

Fungsi : perintah tail digunakan untuk menampilkan 10 baris terakhir dari suatu file teks. Jika ingin menampilkan hanya 3 baris pertama maka digunakan perintah :

```
$ tail -3 coba.txt
-rw-r--r-- 1 syarif users 58527 Aug 12 12:15 tut1_2.pdf
-rw-r--r-- 1 syarif users 15687 Aug 12 12:17 tut2.1998.ps
-rw-r--r-- 1 syarif users 17984 Aug 12 12:18 tut3.1998.ps
```

22. history

Fungsi : mencetak perintah-perintah sebelumnya

```
$ history|more
```

```
1 netscape
2 bg
3 ls -l
4 cd syarif
5 ls -l
6 ls -al
7 cd
8 cd
```

Tidak semua versi UNIX menyediakan perintah ini. Perintah history ini dapat digunakan untuk mengingat kembali perintah-perintah yang dijalankan sebelumnya. Perintah ini juga dapat digunakan untuk menjalankan lagi suatu perintah tanpa mengetik ulang.

Sistem operasi DOS juga menyediakan fasilitas ini yaitu perintah DOSKEY.COM

23. wc

Fungsi : menghitung jumlah kata, jumlah baris dan jumlah karakter dalam suatu file

Format : wc nama-file

Contoh :

```
$ wc lpr
15   15  165 lpr      ----> file lpr terdiri atas 13 baris,
                                         15 kata dan 165 karakter
```

24. alias

Fungsi : untuk mengganti nama perintah

Dengan menggunakan fasilitas ini, seorang user dapat mendefinisikan perintah UNIX dengan perintah lain yang mudah diingat. Fasilitas ini juga sangat menguntungkan bagi pemakai DOS karena bisa mengalihaskan perintah-perintah UNIX dengan perintah-perintah DOS. Perintah alias hanya ada di C Shell dan tidak dijumpai pada Bourne Shell.

Contoh :

> alias dir/w ls	---- perintah ls dialiskan menjadi dir/w		
> dir/w	---- setelah itu perintah dir/w dikenali sebagai ls		
PIC22370.PCX	dbms1.pdf	output	tut1_2.pdf
PIC27480.PCX	dead.letter	profile	tut2.1998.ps
bufferm.pdf	introduction.pdf	syarif	tut3.1998.ps
coba.txt	lpr	tmp	
daadstip.pdf	nsmail	tut1.1998.ps	

Perintah alias juga dapat digunakan untuk menyingkat perintah.

> alias h history	---- perintah history disingkat menjadi h
> h	
1 19:30 dirp	
2 19:30 alias dir/p ls	
3 19:30 dir/p	
4 19:30 dir/w ls -l	

Untuk membatalkan perintah alias digunakan perintah unalias

> unalias h history	---- perintah alias h dibatalkan
> h	---- memanggil perintah h
h: Command not found.	--- perintah h sudah tidak dikenali

25. man

Fungsi : singkatan dari **manual** yaitu untuk menampilkan informasi bantu untuk semua perintah UNIX. Perintah ini sangat bermanfaat bagi setiap pemakai UNIX karena dapat membantu mengingat kembali perintah-perintah UNIX.

Format : man nama-perintah

Contoh :

```
$ man who
WHO(1L)
NAME
    who - show who is logged on
SYNOPSIS
    who [-imqsuwHT] [--count] [--idle] [--heading] [--help]
DESCRIPTION
    This manual page documents the GNU version of who. If
```

given no non-option arguments, who prints the following information for each user currently logged

26. grep

Fungsi : mencari isi suatu file di sembarang directory

Format : grep –n ‘nama-file’ di-direktori

Keterangan : grep akan mencari suatu variable dalam suatu baris tertentu, di dalam sembarang direktori pada semua file. Grep sangat berguna untuk menemukan kata tertentu dalam beberapa dokumen atau mencari adanya sebuah variable dalam sekelompok program. Misalnya : grep –n ‘shutdown’ /etc/*

Contoh :

```
$ grep main *.c          mencari kata main pada semua file *.c
client.c:main(argc,argv)  menampilkan kata-kata yang ditemukan
mumt.c:main()
prakt_1.c:main(argc,argv)
server.c:main(argc,argv)
```

gunakan option –n untuk menampilkan nomer baris.

```
$ grep -n exit server.c
21: exit(1);
36: exit(1);
58: exit(1);
```

27. mkdir

Fungsi : membuat direktori

Format : mkdir nama-direktori

Keterangan : di DOS peritahnya adalah MD (make directory)

~\$ **mkdir coba1 coba2 coba3** (membuat 3 direktori sekaligus)

28. rmdir

Fungsi : menghapus direktori

Format : rmdir nama-direktori

Keterangan : di DOS peritahnya adalah RD (remove directory)

~\$ **rmdir coba1 coba2 coba3** (menghapus 3 direktori sekaligus)
jika directory yang dihapus tidak ada maka akan ditampilkan pesan

SISTEM FILE DAN DIRECTORY

Perintah-perintah UNIX yang berhubungan dengan file dan directory sebagai berikut :

1. **pwd**

Fungsi : menunjukkan directory aktif

Keterangan : Perintah ini singkatan dari “print working directory”. Seorang pemakai selalu berhubungan dengan sebuah directory kerja, yang ada kalanya pemakai tersebut tersesat sampai jauh, maka perintah ini bermanfaat untuk mengetahui posisi pemakai dalam suatu directory.

2. **cd [nama]**

Fungsi : mengubah directory

Keterangan : Membawa pemakai ke directory kerja baru. Bila nama directory tidak ditentukan, cd mengatur supaya directory kerja berada pada home directory pemakai.

3. **ls [options] nama**

Fungsi : menampilkan daftar file dalam directory aktif.

Keterangan :

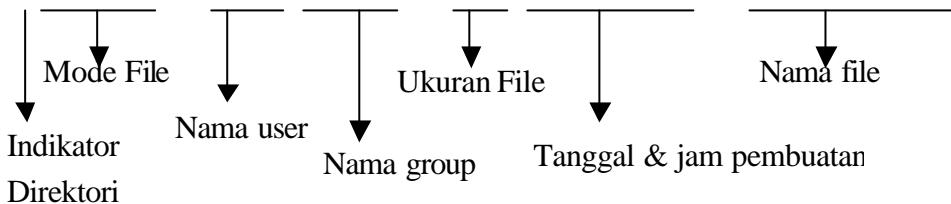
Perintah ini akan menampilkan informasi mengenai directory dan file. Bentuk sederhana perintah ls akan menampilkan hanya nama file. Bentuk panjang ditandai dengan menggunakan option -l, yang akan menampilkan nama-nama file beserta informasi untuk setiap file yang ditampilkan.

PROTEKSI FILE DAN SEKURITI PADA UNIX

Masalah keamanan data amatlah penting dalam bidang teknologi informasi. UNIX dalam hal ini jauh lebih unggul dari pada DOS yang tidak menyediakan fasilitas ini. DOS hanya mempunyai fasilitas ATTRIBUT dengan pilihan Read Only dan Hidden. DOS hanya menyediakan fasilitas untuk menyembunyikan file (hidden) dan memproteksi file (read-only) agar tidak dapat dihapus atau dimodifikasi. Sedangkan fasilitas kepemilikan file tidak disediakan.

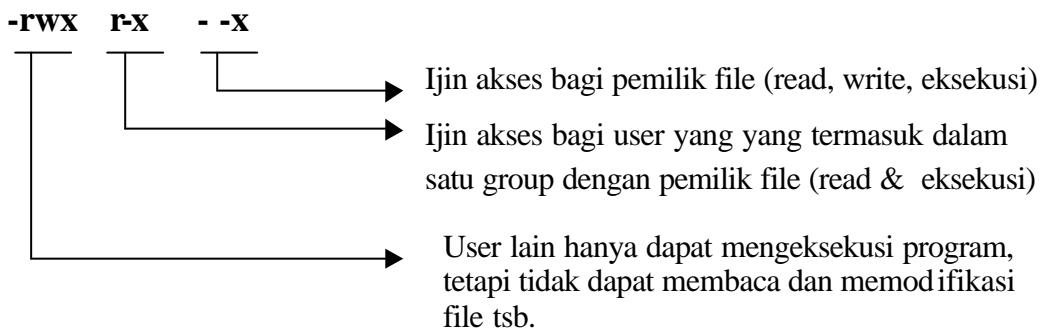
UNIX menyediakan sekuriti yang sangat baik sekali. Tiap-tiap file pada UNIX mempunyai tingkat sekuriti sendiri seperti terlihat pada tulisan di bawah ini:

```
$ ls -l
-rw-r--r--  1 syarif  users   66091 Aug 12 12:14  buffer.pdf
-rw-r--r--  1 syarif  users    171 Nov 23 19:51  coba1
-rw-r--r--  1 syarif  users    177 Nov 23 19:51  coba2
-rw-r--r--  1 syarif  users    171 Nov 23 19:52  coba3
drwx-----  2 syarif  users   1024 Nov 19 11:55  nsmail/
drwx-----  2 syarif  users   1024 Aug 12 11:34  syarif/
drwxr-xr-x  2 syarif  users   1024 Aug 13 12:57  tmp/
```



Kalau dibandingkan dengan DOS, tampilan direktori di atas sedikit berbeda. Selain nama file, ukuran file, tanggal dan jam pembuatan, juga ada tambahan informasi berupa nama user, nama group dan mode file.

Untuk keterangan kolom directory indicator adalah merupakan informasi untuk file yang ditampilkan bahwa nama yang ditampilkan adalah sebuah sub directory. Kolom kedua menunjukkan hak akses bagi sekelompok atau hanya untuk satu pemakai itu sendiri, penjelasan selengkapnya sebagai berikut :



Arti dari karakter rwx, r (read) menyatakan ijin untuk membaca file, w (write) menyatakan ijin untuk menulis (memodifikasi), merubah maupun menghapus, dan x (execute) menyatakan ijin untuk menjalankan program. Karakter pada posisi bersangkutan menunjukkan bahwa ijin telah dicabut.

MENGUBAH IJIN AKSES

Ijin akses terhadap file atau suatu directory dapat diubah-ubah dengan perintah chmod, yang bentuk perintahnya sebagai berikut :

chmod modus nama file [who] operasi ijin
Fungsi : merubah ijin akses

Keterangan :
Mengubah modus (file protection bits) sebuah file. Modus file adalah susunan bit dalam isi directory file yang menentukan siapa yang mendapat ijin untuk membaca,

menulis, mengerjakan file. Hanya pemilik file atau super user yang boleh mengubah modusnya.

Bentuk pertama menggunakan bilangan oktal untuk mengatur modus. Yang kedua menggunakan simbol untuk menentukan modus baru.

Argumen : Pembacaan, penulisan dan penggerjaan file menginginkan bit ditentukan dalam bentuk bilangan oktal. Bilangan oktal terdiri atas 4 digit. Most significant digit berisi bit untuk mengontrol penggerjaan, digit berikut mempunyai bit, bit ijin pemilik, digit berikutnya mempunyai bit ijin kelompok dan least significant digit mengandung bit ijin umum.

Bilangan oktal diperoleh dengan logic OE dengan bit berikut :

4000	menentukan ID pemilik pada penggerjaan
2000	menentukan ID kelompok pada penggerjaan
1000	meninggalkan program dalam memori
0400	ijin membaca (Read) untuk pemakai
0200	ijin menulis (Write) untuk pemakai
0100	ijin mengerjakan (eXsekusi) untuk pemakai
0040	ijin membaca (Read) untuk anggota kelompok group
0020	ijin menulis untuk anggota kelompok
0010	ijin mengerjakan untuk anggota kelompok
0004	ijin membaca untuk orang lain
0002	ijin menulis untuk orang lain
0001	ijin mengerjakan untuk orang lain
who	menentukan untuk siapa ijin diberikan
+	kombinasi 3 huruf,”o”,”g” dan “u” atau huruf tunggal “a”.

Default dari who ini adalah “a”.

a	(all) semua
g	(group) kelompok
u	(user) pemakai
+	menambah ijin
-	menghapus ijin
ijin	r membaca
	w menulis
	x membaca
	s mengatur ID pemilik/kelompok
	t jangan menukar

Contoh mengubah mode file

```
$ ls -l coba
-rw-r--r-- 1 syarif users 206 Nov 26 13:17 coba          (kondisi mula-mula)
$ chmod 777 coba                                     (ubah mode file)
$ ls -l coba
-rwxrwxrwx 1 syarif users 206 Nov 26 13:17 coba*      (mode di set 777)
```

\$ chmod 700 coba	(ubah mode lagi)
\$ ls -l coba	
-rwx----- 1 syarif users 206 Nov 26 13:17 coba*	(mode di set 700)
\$ chmod 751 coba	(ubah mode file)
\$ ls -l coba	
-rwxr-x--x 1 syarif users 206 Nov 26 13:17 coba*	(mode di set 751)
\$ chmod 000 coba	(ubah mode file)
\$ ls -l coba	
--- --- --- 1 syarif users 206 Nov 26 13:17 coba*	(mode di set 000)
\$ chmod a+x coba	(ubah mode file)
\$ ls -l coba	
---x--x--x 1 syarif users 206 Nov 26 13:17 coba*	(mode di set 751)
\$ chmod u+rwx coba	(ubah mode file)
\$ ls -l coba	
-rwxr-x--x 1 syarif users 206 Nov 26 13:17 coba*	(user +r, +w, +x)

PERINTAH – PERINTAH LANJUTAN

1. Perintah Input-Output dan Proses Piping

Sama halnya dengan DOS, UNIX juga mampu mengenali beberapa karakter khusus dalam suatu perintah. Yang pertama adalah karakter > yang digunakan untuk menyalurkan standard output dari terminal ke sebuah file.

Contoh :

\$ ls > output.txt	----- hasil dari perintah ls disimpan ke sebuah file dengan nama output.txt
\$ more output.txt	----- melihat isi file output.txt

dead.letter
introduction.pdf
nsmail/
output
syarif/
tmp/
tut1.1998.ps
tut2.1998.ps

Dengan perintah di atas sebuah file baru dengan nama ‘output.txt’ akan terbentuk dan isi dari file tersebut adalah hasil dari perintah ls.

Selain itu user juga dapat menambahkan informasi di akhir sebuah file dengan cara menggunakan notasi >> seperti contoh berikut ini :

```
$ date >> output.txt ----- hasil dari perintah date ditambahkan
                                         ke akhir file dengan nama output.txt
$ more output.txt      ----- melihat isi file output.txt
dead.letter
introduction.pdf
nsmail/
output
syarif/
tmp/
tut1.1998.ps
tut2.1998.ps
Thu Nov 26 14:29:38 JVT 1998 ----- > baris tambahan dari perintah date
```

Input dari suatu perintah juga dapat diarahkan dengan menggunakan tanda <. Dengan menggunakan notasi ini, input yang biasanya didapat secara interaktif dari keyboard akan sebaliknya diarahkan untuk mengambil input dari sebuah file.

```
$ ls -l > coba.txt      ---→ hasil perintah ls disimpan menjadi coba.txt
$ wc < coba.txt        --→ wc menghitung baris, kata & karakter
18   155   1126          dari file coba.txt
```

Penggunaan alias seperti contoh-contoh di atas hanyalah berlaku sementara, artinya setelah logout maka perintah-perintah tersebut sudah tidak dikenali lagi. Ada suatu cara agar perintah alias tersebut dapat bersifat permanen, yaitu dengan cara mengetikkan perintah-perintah alias tersebut dan menyimpannya pada suatu file yang bernama .cshrc yang dapat ditemukan pada setiap home directory user.

Dalam hal proses input-output UNIX mempunyai proses yang disebut dengan proses piping. Konsep dasar piping berhubungan erat dengan menyalurkan input output. Pipe bisa dianalogikan dengan sebuah pipa yang menghubungkan dua buah program sedemikian hingga output dari program pertama menjadi input dari program berikutnya. UNIX menyediakan fasilitas piping dimana user bisa menggunakan satu atau lebih pipe secara bersamaan.

Contoh proses piping dalam UNIX :

```
$ finger | sort | more
Login Name           Tty Idle Login Time Office  Office Phone
amran Amran.D       p1 8:13 Nov 27 08:32 (:0.0)
syarif Iwan Syarif  p3   Nov 27 16:29 (10.252.1.201)
```

Keterangan : perintah di atas berarti setelan menjalankan perintah finger kemudian hasilnya diurutkan sesuai abjad, kemudian menampilkannya per layar.

```
$ finger | sort | more | wc
4   35   269
```

Keterangan : perintah di atas berarti pertama-tama perintah finger dijalankan, lalu hasilnya diurutkan sesuai abjad, kemudian dihitung berupa jumlah baris, jumlah kata dan jumlah karakter dari output yang dihasilkan.

2. adduser

Fungsi : menambah user baru

Format : adduser

Keterangan : hanya super-user yang dapat menjalankan perintah ini.

higo:~# adduser

Adding a new user. The username should not exceed 8 characters in length, or you may run into problems later.

**Enter login name for new account (^C to quit): iwan
That name is in use, choose another.**

Adding a new user. The username should not exceed 8 characters in length, or you may run into problems later.

Enter login name for new account (^C to quit):

Pada contoh di atas, nama user yang dimasukkan sudah ada sehingga UNIX menolaknya. UNIX menyarankan agar panjang nama user tidak melebihi 8 karakter. Untuk membatalkan perintah ini tekan ^C.

higo:~# adduser

Enter login name for new account (^C to quit): iwanarif

Editing information for new user [iwanarif]

Full Name: Iwan Syarif

GID [100]:

Group 'users', GID 100

First unused uid is 562

UID [562]:

Home Directory [/home/iwanarif]:

Shell [/bin/bash]:

Password [iwanarif]:

Information for new user [iwanarif]:

Home directory: [/home/iwanarif] Shell: [/bin/bash]

uid: [562] gid: [100]

Is this correct? [y/N]:

Tekan Y untuk menyimpan data user baru.

3. ps

Fungsi : digunakan untuk melaporkan informasi tentang proses yang aktif dalam sistem UNIX

Format : ps -cal

Keterangan : dapat menunjukkan program yang sedang dijalankan setiap terminal dan juga proses latar belakang yang sedang berjalan.

```
higo:/usr$ ps
  PID TTY STAT TIME COMMAND
 7241 p2 S  0:00 -bash
 7251 p2 T  0:00 (man)
 7253 p2 T  0:00 (sh)
 7254 p2 T  0:00 (gunzip)
 7255 p2 T  0:00 (less)
 7272 p2 T  0:00 (man)
 7283 p2 T  0:00 sh -c /usr/bin/gunzip -c /usr/man/cat1/mv.1.gz | /usr/bin/l
 7285 p2 T  0:00 /usr/bin/less -is
```

4. kill

Fungsi : digunakan untuk menghentikan proses yang sedang berjalan

Format : kill -no signal

Keterangan : Proses diidentifikasi dengan nomor proses (bisa disebut process id atau PID) yang ditemukan dari perintah ps. Perintah kill bekerja dengan mengirimkan sebuah sinyal terminate pada proses. Biasanya dipakai signal no. 9 atau no. 19.

Contoh :

kill 7473 menghentikan proses dengan nomer ID 7473

kill -9 7473 menghentikan proses 7473 langsung saat itu juga

5. &

Fungsi : Menjalankan program di belakang layar (multitasking)

Format : & nama-program

Keterangan : Setelah perintah ini dijalankan maka PID dari program akan dilaporkan oleh sistem. Jadi jika kita ingin membuang proses tadi kita bisa melakukan perintah kill -9 PID-nya.

Contoh proses belakang layar

\$ top& ---- menjalankan perintah top secara background

Keterangan :

Perintah top digunakan untuk melihat proses-proses penting yang sedang dikerjakan oleh CPU. Jika proses top dijalankan tidak dengan cara di belakang layar, maka user tidak dapat mengerjakan perintah yang lain karena perintah ini akan dikerjakan secara terus menerus oleh CPU. Bentuk tampilan yang muncul di layar jika menggunakan perintah top adalah sebagai berikut :

```
4:17pm up 2 days, 4:22, 3 users, load average: 0.17, 0.08, 0.02
84 processes: 54 sleeping, 1 running, 0 zombie, 29 stopped
CPU states: 0.3% user, 1.9% system, 0.0% nice, 97.8% idle
Mem: 14336K av, 14080K used, 256K free, 14060K shrd, 212K buff
Swap: 130748K av, 8728K used, 122020K free           2952K cached
```

Tetapi jika perintah top dikerjakan secara background (top&) maka di layar tidak akan ada tampilan apa-apa. Untuk mematikan perintah top gunakan perintah kill -9 top.

6. bc

Fungsi : perintah bc dapat digunakan sebagai calculator

Keterangan : fasilitas ini tidak ada pada versi UNIX standard.

Contoh :

```
higo:~$ bc
bc 1.03 (Nov 2, 1994)
Copyright (C) 1991, 1992, 1993, 1994 Free Software Foundation, Inc.
This is free software with ABSOLUTELY NO WARRANTY.
For details type `warranty'.
10+5
15

3+5*5+65-78*5^2
-1857
```

7. pr

Fungsi : Mencetak isi file ke printer

Format : pr nama-file > /dev/lp0

Keterangan : Hasil cetakan ke-printer akan dipisahkan perhalaman dengan sebuah judul (header) pada bagian atas halaman. Header akan berisi tanggal, nama file dan nomer halaman.

ELECTRONIC MAIL

Di dalam sistem UNIX, memperbolehkan pemakai saling mengirimkan pesan secara elektronik. Pesan dapat dikirimkan secara langsung ke terminal pemakai lain bila sedang log in. Kedua pemakai tersebut kemudian dapat melanjutkan percakapan melalui keyboard masing-masing.

Seorang pemakai juga dapat mengirimkan pesan ke pemakai lain dalam bentuk surat pengiriman (mail). Pesan tersebut secara otomatis oleh UNIX dimasukkan ke file dalam directory penerima. Bila penerima pesan tersebut log in, maka ia diberitahu akan adanya pesan pengiriman.

Perintah-perintah yang digunakan untuk keperluan komunikasi data antar pemakai sebagai berikut :

1. write pemakai [tty]

fungsi : mengirim pesan ke pemakai yang sedang login.
keterangan : Write akan membuat hubungan dari keyboard ke layar pemakai yang ditentukan. Apa saja yang diketikkan dari keyboard akan tampak di layar penerima. Untuk mengakhiri pengiriman pesan ketikkan ^D. Sewaktu pesan dikirimkan ke penerima, maka di layar penerima pesan akan ditampilkan siap yang mengirim pesan tersebut.

Option tty (terminal type), harus diikuti oleh nomor urut terminal yang diaktifkan. Option tty akan muncul bila terdapat lebih dari satu terminal yang diaktifkan dengan ID yang sama.

Seorang user pada komputer A akan mengirimkan pesan kepada user di komputer B dengan cara :

\$ write bambang

User bernama Doni mengirimkan pesan ke Bambang dengan cara mengetik pesan. Untuk mengakhiri pengetikan pesan digunakan tombol Ctrl-D.

Pada komputer B yang digunakan oleh user bernama bambang akan tampil pesan yang diketik oleh Doni seperti pada tampilan berikut ini:

```
~$  
Message from doni@higo on ttyp2 at 20:21 ...  
Hallo Wan, kamu sekarang sedang ngapain ?  
EOF
```

2. mesg [pilihan]

fungsi : menolak pesan dari pemakai lain

keterangan : Anda dapat juga menolak pesan yang dikirim dengan memakai perintah write. Perintah ini tidak dapat menolak ijin bagi super user untuk mengirim pesan.

Pilihan yang digunakan :

n tidak menerima pesan

y mengaktifkan penerimaan pesan

Jika user tidak ingin diganggu oleh user yang lain yang akan mengirimkan pesan, maka dapat menggunakan perintah **mesg n**.

Contoh :

User bernama Iwan tidak ingin menerima pesan . (mesg n)

Kemudian user lain bernama Bambang mengirimkan pesan, maka komputer akan menolak pesan tersebut.

```
higo:~$ write iwan  
write: iwan has messages disabled
```

3. mail [penerima]

Fungsi : kirim dan membaca pesan berupa surat

keterangan : Mail adalah sebuah program pengiriman elektronik yang mengirimkan pesan ke pemakai lain atau membaca pesan dari pemakai lain. Anda dapat mengirimkan pesan dan bentuk pengiriman tercatat ke tiap pemakai yang dikehendaki yang mempunyai ID log in pada sistem. Pengiriman dikumpulkan dalam sebuah file di home directory penerima. Untuk membaca surat elektronik ini bisa setiap saat, tetapi kebanyakan membacanya ketika log in. Bila anda mengetikkan mail tanpa ID penerima maka berarti anda membaca surat elektronik dari pengirim. Untuk membaca satu per satu surat anda, dengan mengetikkan urutan dari suart yang masuk. Urutan yang digunakan 1 2 3 dst.

Tombol yang digunakan adalah :

+ (return) : menampilkan pesan berikutnya

^D : keluar dari program mail

d : menghapus pesan terakhir

Bila perintah ini diikuti oleh nomor urut surat, maka akan menghapus file surat tersebut. Bila disertakan argumen pemakai, maka mail menerima masukkan hingga ia mendeteksi sebuah tombol ^D dan mengirim masukkan yang dikumpulkan sebagai surat pesan pada home directory pemakai yang ditentukan.

Contoh : mengirim e-mail ke user bernama bambang

\$ mail bambang

Subject: Mencoba kirim e-mail

Hallo Mbang,

aku sedang latihan mengetik e-mail pakai LINUX.

Kok sulit sekali yah ?

.

EOT

Jika user bernama Bambang sedang login, maka e-mail tersebut langsung ditampilkan di layar, seperti pada contoh berikut ini :

\$

New mail for bambang@higo has arrived:

From: Iwan Syarif <syarif@higo.eepis-its.ac-id.net>

Subject: Mencoba kirim e-mail

Hallo Mbang,

aku sedang latihan mengetik e-mail pakai LINUX.

Kok sulit sekali yah ?

Jika user yang dikirim e-mail tidak sedang login, maka surat tersebut akan disimpan pada direktori user. User akan diberitahu oleh UNIX pada saat login jika ada surat masuk. Untuk mengetahui atau membaca e-mail yang masuk, ketik perintah **mail**.

```
~$ login bambang
Password:
Linux 2.0.0.
You have mail.      ----- pemberitahuan jika ada e-mail masuk

~$ mail          ----- melihat e-mail yang masuk
Mail version 5.5 6/1/90. Type ? for help.
"/var/spool/mail/syarif": 2 messages 1 new 2 unread
U 1 syarif@higo.eepis-it Thu Aug 27 15:31 11/435 "percobaan"
>N 2 umum@higo.eepis-its. Thu Nov 26 21:32 12/480 "Surat Penting "
&
```

4. wall

Fungsi : pengiriman pesan oleh super user

Keterangan : Bagi super user, sistem operasi UNIX menyediakan pengiriman pesan keseluruhan pemakai yang sedang log in saat itu dan perintah ini hanya dapat dilakukan oleh super user. Kegunaan dari perintah ini yaitu super user dapat memberitahukan kepada seluruh pemakai yang sedang log in, adanya suatu interupsi penting yang harus diperhatikan.

```
# wall
Hallo USERS, awas komputer mau dimatikan jam 19.00
Mohon data Anda disimpan dan komputernya
di shutdown dahulu. Terima kasih.
```

Untuk mengakhiri penulisan wall dengan cara menekan tombol Ctrl – D.
Setelah itu pada layar monitor user yang sedang login akan tampil :

```
Broadcast Message from syarif@higo
(/dev/ttyp3) at 17:20 ...

Hallo USERS, awas komputer mau dimatikan jam 19.00
Mohon data Anda disimpan dan komputernya
di shutdown dahulu. Terima kasih.
#
```

Selain itu perintah wall dapat juga mengirimkan pesan berupa file teks. Isi dari file teks tersebut akan ditampilkan di semua layar.

```
# wall coba.txt
```