

# Sistem File

**Nama: Rizqi Bagus Andrean**

**Kelas: TI-1D**

**Absen: 25**

## Pokok Bahasan:

- Sistem File

## Tujuan Belajar:

Setelah mempelajari materi dalam bab ini, mahasiswa diharapkan mampu:

- Memahami atribut file dan ijin akses
- Memahami perintah untuk mengubah ijin akses suatu file

Menggunakan perintah-perintah untuk mengubah ijin akses

## Dasar Teori:

### 1. ATRIBUT FILE

File mempunyai beberapa atribut, antara lain :

- Tipe file : menentukan tipe dari file, yaitu :
- Ijin akses : menentukan hak user terhadap file ini.
- Jumlah link : jumlah link untuk file ini.
- Pemilik (Owner) : menentukan siapa pemilik file ini
- Group : menentukan group yang memiliki file ini
- Jumlah karakter : menentukan ukuran file dalam byte
- Waktu pembuatan : menentukan kapan file terakhir dimodifikasi
- Nama file : menentukan nama file yang dimaksud

Contoh:

auth 1639 Oct 31 20:19 /etc/passwd

Pemilik Group Jml link Waktu Nama file

### 2. IJIN AKSES

Setiap obyek pada Linux harus mempunyai pemilik, yaitu nama pemakai Linux (account) yang terdaftar pada */etc/passwd*.

Ijin akses dibagi menjadi 3 peran yaitu :

- Pemilik (Owner)
- Kelompok (Group)
- Lainnya (Others)

Setiap peran dapat melakukan 3 bentuk operasi yaitu :

- Pada File
  - R (Read) ⇔ Ijin untuk membaca

- W (Write) ⇔ Ijin untuk mengubah / membuat
- X (Execute) ⇔ Ijin untuk menjalankan program

- Pada Direktori

- R (Read) ⇔ Ijin untuk membaca daftar file dalam direktori
- W (Write) ⇔ Ijin untuk mengubah/membuat file di direktori
- X (Execute) ⇔ Ijin untuk masuk ke direktori (cd)

Pemilik File/Direktori dapat mengubah ijin akses sebagai berikut :

```
-rwxrwxrwx 1 student test 1639 Oct 31 20:19 file
```

Format untuk mengubah ijin akses

```
chmod [ugoa] [= + -] [rwx] File(s)
```

```
chmod [ugoa] [= + -] [rwx] Dir(s)
```

dimana

- u = user (pemilik)
- g = group (kelompok)
- o = others (lainnya)
- a = all

Format lain dari chmod adalah menggunakan bilangan octal sebagai berikut

```

r      w      x
4      2      1      =      7

```

### 3. USER MASK

Untuk menentukan ijin akses awal pada saat file atau direktori dibuat digunakan perintah umask. Untuk menghitung nilai default melalui umask pada file, maka dapat dilakukan kalkulasi sebagai berikut :

Kreasi file (biasa)	6 6 6
Nilai umask	0 2 2
	----- -
	6 4 4
 Kreasi direktori	 7 7 7
Nilai umask	0 2 2
	----- -
	7 5 5

## TUGAS PENDAHULUAN:

Jawablah pertanyaan- pertanyaan berikut ini:

1. Apa saja atribut file? Berikan contoh file dengan tipe file yang disebutkan pada dasar teori.

Atribut file adalah dimana sebuah file atau direktori bisa eksis atau sebuah file mempunyai atribut yang berbeda antara sistem operasi satu dengan yang lainnya. Atribut file digunakan oleh sistem operasi untuk memisahkan tipe-tipe file. Setiap file di windows mempunyai atribut tersendiri sesuai dengan kepentingan file tersebut terhadap sistem. Misalnya, tipe file yang tidak boleh dihapus oleh user diberi attribut system, jadi ketika file manager (windows explorer) akan menampilkannya, file beratribut sistem tersebut tidak akan ditampilkan.

- Tipe file : menentukan tipe dari file, yaitu :

Karakter	Arti
-	File biasa
d	Direktori
l	Symbolic link
b	Block special file
c	Character special file
s	Socket link
p	FIFO

- Ijin akses : menentukan hak user terhadap file ini.
- Jumlah link : jumlah link untuk file ini.
- Pemilik (Owner) : menentukan siapa pemilik file ini
- Group : menentukan group yang memiliki file ini
- Jumlah karakter : menentukan ukuran file dalam byte
- Waktu pembuatan : menentukan kapan file terakhir dimodifikasi
- Nama file : menentukan nama file yang dimaksud

Contoh:

```
-rw-rw-r-- 1 bin auth 1639 Oct 31 20:19 /etc/passwd
```

Diagram explaining the components of the file command output:

- Ijin akses**: Points to the permissions `-rw-rw-r--`.
- Jml link**: Points to the link count `1`.
- Pemilik**: Points to the owner `bin`.
- Group**: Points to the group `auth`.
- Jml karakter**: Points to the file size `1639`.
- Waktu**: Points to the modification date and time `Oct 31 20:19`.
- Nama file**: Points to the file path `/etc/passwd`.

Contoh file dengan tipe file yang disebutkan pada dasar teori:

- `-rwr--`==Tipe file biasa
- `drw--`==Tipe file direktori
- `lrw--`==Tipe file symbolic link
- `brw--`==Tipe file block

2. Apa yang dimaksud ijin akses? Bagaimana contoh penggunaan perintah `chmod` untuk mengubah ijin akses.

Ijin akses bisa dibidang sebuah hak yang diberikan pada seluruh User atau file untuk melakukan sesuatu akses. Jadi hak akses ini dapat diatur sedemikian rupa supaya file atau

direktori itu hanya bisa di read, atau write, dan execute. Selain itu kita juga bisa mengatur bagaimana hak akses dari setiap user pada suatu file, atau direktori. Hal yang harus dikuasai dalam konfigurasi Hak Akses pada Linux:

1. Mengetahui Fungsi CHMOD
2. Mengetahui Fungsi setiap atribut tambahan dalam perintah CHMOD
3. Dapat mengatur hak akses bagi setiap user
4. Dapat mengatur hak akses bagi setiap group
5. Dapat merencanakan hak akses yang diaplikasikan di kehidupan sehari-hari atau Industri

Sesuai dengan semua yang harus kita kuasai kita harus memahami fungsi dari perintah "CHMOD" dan atribut tambahan pada perintah CHMOD tersebut. Chmod merupakan salah satu perintah pada Linux yang berfungsi untuk mengurangi atau memberikan izin ( Hak Akses ) pada suatu file atau direktori dan User serta Group.

Format untuk mengubah izin akses

```
chmod [ugoa] [= + -] [rwx] File(s)
chmod [ugoa] [= + -] [rwx] Dir(s)
```

dimana

u = user (pemilik)  
g = group (kelompok)  
o = others (lainnya)  
a = all

Format lain dari chmod adalah menggunakan bilangan octal sebagai berikut

	w	x	
4	2	1	= 7

### 3. Berikut contoh penggunaan perintah umask untuk mengubah izin akses.

User mask adalah kemampuan sistem operasi berbasis Linux untuk menambahkan pengaturan hak akses default untuk file-file baru (termasuk folder). Untuk tujuan edukasional, berikut adalah nilai-nilai oktal yang dapat digunakan untuk mengatur hak akses atas file:

0 – Read, Write, Execute (rwx)  
1 – Read and Write (rw-)  
2 – Read and Execute (r-x)  
3 – Read only (r -)  
4 – Write andExecute (wx) 5  
– Writeonly (-w-)  
6 – Execute only (- x)  
7 – No permissions (—)

Untuk menghitung nilai defaultmelalui umask pada file, maka dapat dilakukan kalkulasi sebagai berikut :

Kreasi file (biasa)	6 6 6
Nilai umask	0 2 2
	-----
	6 4 4

```
Kreasi direktori    7 7 7
Nilai umask         0 2 2
-----
7 5 5
```

## 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 : Ijin Akses

1. Melihat identitas diri melalui etc/passwd atau etc/group, informasi apa yang ditampilkan?

```
$ id
$ grep <user> /etc/passwd
$ grep [Nomor group id]/etc/group
```

```
Last login: Thu May 9 04:10:56 2024
root@bagusok:~# id
uid=0(root) gid=0(root) groups=0(root)
root@bagusok:~# grep root /etc/passwd
root:x:0:0:root:/bin/bash
root@bagusok:~# grep 0/etc/group
```

Analisa: Percobaan ini untuk mengetahui id dari pengguna dan mencetak atau memberikan informasi tentang nomor id pada group dan user.

2. Memeriksa direktori home

```
$ ls -ld /home/<user>
```

Analisa: Percobaan ini digunakan untuk mengetahui direktori user yang kita gunakan.

```
root@bagusok:~# ls -ld /home/bagus
drwxr-xr-x 3 bagus bagus 4096 Mar 26 04:11 /home/bagus
root@bagusok:~#
```

3. Mengubah Ijin akses (chmod). Perhatikan ijin akses setiap perubahan !

```
$ touch f1 f2 f3
$ ls -l
$ chmod u+x f1
$ ls -l f1
$ chmod g=w f1
$ ls -l f1
$ chmod o-r f1
$ ls -l f1
```

```

drwxr-xr-x 3 bagus bagus 4096 Mar 20 04:11 /home/bagus
root@bagusok:~# touch f1 f2 f3
root@bagusok:~# ls -l
total 20
-rw-r--r-- 1 root root    0 May  9 04:14 f1
-rw-r--r-- 1 root root    0 May  9 04:14 f2
-rw-r--r-- 1 root root    0 May  9 04:14 f3
drwxr-xr-x 2 root root 4096 Apr 23 07:01 jobsheet8
-rw-r--r-- 1 root root 2040 Apr 23 06:06 passwd
drwx----- 3 root root 4096 Apr 16 05:38 snap
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:~# chmod u+x f1
root@bagusok:~# ls -l f1
ls: cannot access '-l': No such file or directory
f1
root@bagusok:~# chmod g=w f1
root@bagusok:~# ls -l f1
ls: cannot access '-l': No such file or directory
f1
root@bagusok:~# chmod o-r f1
root@bagusok:~# ls -l f1
ls: cannot access '-l': No such file or directory
f1

```

```
$ chmod a=x f2
```

```
$ ls -l f2
```

```
$ chmod u+x,g-r,o=w f3
```

```
$ ls -l f3
$ chmod 751 f1
$ chmod 624 f2
$ chmod 430 f3
```

```
root@bagusok:~# ls -l f3
-rw-r--r-- 1 root root 0 May 9 04:14 f3
root@bagusok:~# chmod 751 f1
root@bagusok:~# chmod 624 f2
root@bagusok:~# chmod 430 f3
root@bagusok:~# ls -l f1 f2 f3
ls: cannot access 'f1': No such file or direc
```

```
$ ls -l f1 f2 f3
```

Analisa: Percobaan ini digunakan

```
root@bagusok:~# ls -ls
total 20
0 -rwxr-x--x 1 root root 0 May 9 04:14 f1
0 -rw--w-r-- 1 root root 0 May 9 04:14 f2
0 -r---wx--- 1 root root 0 May 9 04:14 f3
4 drwxr-xr-x 2 root root 4096 Apr 23 07:01 jobsheet8
4 -rw-r--r-- 1 root root 2040 Apr 23 06:06 passwd
4 drwx----- 3 root root 4096 Apr 16 05:38 snap
4 drwxr-xr-x 2 root root 4096 Mar 26 04:47 test
4 drwxr-xr-x 2 root root 4096 Apr 16 06:20 uas
root@bagusok:~#
```

untuk

mengubah ijin akses dengan perintah `chmod u+x f1` Maka hasilnya `-rwx-w-r--` untuk emngubah ijin akses kita lakukan perintah `chmod g=w f1` maka hasilnya akan seperti ini `-rwx-w--`.

4. Mengganti kepemilikan digunakan perintah `chown`. Masuk ke root untuk mengganti kepemilikan tersebut.

```
$ su root
$ echo Hallo > f1
$ ls -l f1
```

```
-rwxr-x--x 1 root root 6 May 9 04:17 f1
```

```
$ chown <user-baru> f1 contoh: chown student1 f1
$ ls -l f1
```

```
-rwxr-x--x 1 bagusok root 6 May 9 04:17 f1
```

Analisa: Percobaan ini disini saya mengubah kepemilikan dengan perintah `chown`, untuk mengubah kepemilikan kita harus menambahkan user baru dengan contoh `useradd students` disitu terdapat kepemilikan file `f1` yaitu `students`, dengan mengganti kepemilikan kita ketik perintah `chown students f1` maka hasilnya yang tadinya kepemilikan file `f1` adalah `students` setelah diganti kepemilikannya menjadi `student1`.

5. Ubahlah ijin akses home directory <user> (student) pada root sehingga <user-baru> (student1) pada satu group dapat mengakses home direktory <user>. Hal ini dimaksudkan agar file `f1` yang sudah diubah kepemilikannya dapat diakses <user-baru>. Perubahan ijin akses home directory <user> hanya dapat dilakukan pada root.

```
$ chmod g+rw /home/<user> contoh : chmod g+rw /home/student
$ ls -l /home
```



```
$ exit
```

```
root@bagusok:~# chmod g+rx /home/bagus
root@bagusok:~# ls -la /home
total 12
drwxr-xr-x  3 root  root  4096 Mar 26 04:11 .
drwxr-xr-x 19 root  root  4096 Feb 20 03:20 ..
drwxrwxr-x  3 bagus bagus 4096 Mar 26 04:11 bagus
root@bagusok:~#
```

Analisa: Percobaan ini mengubah ijin akses home directory.

6. Sekarang cobalah untuk substitute user ke <user-baru> (student1).

Cobalah untuk mengakses file f1

```
$ su <user-baru>
$ ls -l f1
$ cat f1
$ exit
```

Analisa: Percobaan ini yaitu mensubstitusi user ke user baru, dan mengakses file f1.

7. Mengubah group dengan perintah \$ chgrp

```
$ grep root /etc/group
$ grep other /etc/group
root@bagusok:~# grep root /etc/group
root:x:0:
root@bagusok:~# grep other /etc/group
$ su
```

```
$ chgrp root f1
```

```
$ ls -l f1
```

```
-rwxr-x--x 1 bagusok root 6 May 9 04:17 f1
```

```
$ chgrp <group-baru> f3
```

```
addgroup: the group 'anton' already exists
root@bagusok:~# chgrp anton f3
root@bagusok:~# ls -la
```

```
-rw--w-r-- 1 root root 0 May 9 04:14 f2
-r---wx--- 1 root anton 0 May 9 04:14 f3
```

```
$ ls -l f3
```

```
$ exit
```

Analisa: Percobaan ini dengan perintah chgrp maka akan mengubah group. Sebelumnya dengan menambahkan addgroup terlebih dahulu.

## Percobaan 2 : User Mask

### 1. Menentukan ijin akses awal pada saat file atau direktori dibuat

```
$ touch myfile
```

```
$ ls -l myfile
```

```
root@bagusok:~# ls myfile -la
-rw-r--r-- 1 root root 0 May 9 04:27 myfile
root@bagusok:~# |
```

Analisa: Percobaan ini yaitu bertujuan untuk menentukan ijin akses awal pada saat direktori dibuat.

### 2. Melihat nilai umask

```
$ umask
```

```
root@bagusok:~# umask
0022
```

Analisa: Percobaan ini digunakan untuk melihat nilai umask dengan perintah \$umask.

### 3. Modifikasi nilai umask

```
$ umask 027
```

```
$ umask
```

```
0022
root@bagusok:~# umask 027
root@bagusok:~# umask
0027
root@bagusok:~# |
```

```
$ touch file_baru
```

```
$ mkdir mydir
```

```
$ ls -l
```

```
$ umask 077
```

```
$ touch xfiles
```

```
$ mkdir xdir
```

```
$ ls -l
```

```

0027
root@bagusok:~# touch test
root@bagusok:~# mkdir mydir
root@bagusok:~# ls -l
total 28
-rwxr-x--x 1 bagusok root      6 May  9 04:17 f1
-rw--w-r-- 1 root    root      0 May  9 04:14 f2
-r---wx--- 1 root    anton     0 May  9 04:14 f3
drwxr-xr-x 2 root    root    4096 Apr 23 07:01 jobsheet8
drwxr-x--- 2 root    root    4096 May  9 04:28 mydir
-rw-r--r-- 1 root    root      0 May  9 04:27 myfile
-rw-r--r-- 1 root    root   2040 Apr 23 06:06 passwd
drwx----- 3 root    root    4096 Apr 16 05:38 snap
drwxr-xr-x 2 root    root    4096 May  9 04:28 test
drwxr-xr-x 2 root    root    4096 Apr 16 06:20 uas
root@bagusok:~# umask 077
root@bagusok:~# touch xfiles
root@bagusok:~# mkdir xdir
root@bagusok:~# ls -l
total 32
-rwxr-x--x 1 bagusok root      6 May  9 04:17 f1
-rw--w-r-- 1 root    root      0 May  9 04:14 f2
-r---wx--- 1 root    anton     0 May  9 04:14 f3
drwxr-xr-x 2 root    root    4096 Apr 23 07:01 jobsheet8
drwxr-x--- 2 root    root    4096 May  9 04:28 mydir
-rw-r--r-- 1 root    root      0 May  9 04:27 myfile
-rw-r--r-- 1 root    root   2040 Apr 23 06:06 passwd
drwx----- 3 root    root    4096 Apr 16 05:38 snap
drwxr-xr-x 2 root    root    4096 May  9 04:28 test
drwxr-xr-x 2 root    root    4096 Apr 16 06:20 uas
drwx----- 2 root    root    4096 May  9 04:28 xdir
-rw----- 1 root    root      0 May  9 04:28 xfiles
root@bagusok:~# |

```

Analisa: Percobaan ini bertujuan untuk memodifikasi nilai umask pada directory.

## LATIHAN:

1. Lakukan tiga cara berbeda untuk setting ijin akses ke file atau direktori menjadi `r--r--r--`.  
Buatlah sebuah file dan lihat apakah yang anda lakukan benar.

```

root@bagusok:~/jb11# chmod 444 1
root@bagusok:~/jb11# chmod u=r,g=r,o=r 2
root@bagusok:~/jb11# chmod +r,+r,+r 3
root@bagusok:~/jb11# ls-la
ls-la: command not found
root@bagusok:~/jb11# ls -la
total 8
drwx----- 2 root root 4096 May  9 04:29 .
drwx----- 13 root root 4096 May  9 04:29 ..
-r--r--r-- 1 root root  0 May  9 04:29 1
-r--r--r-- 1 root root  0 May  9 04:29 2
-rw-r--r-- 1 root root  0 May  9 04:29 3
root@bagusok:~/jb11# chmod +r +r +r 3

```

Analisa : Saya membuat file dengan nama kadek dengan menggunakan perintah \$touch. Ketika file ini saya lihat dengan menggunakan perintah ls -l maka terdapat informasi rw yang artinya file tersebut memberikan izin kepada saya untuk membaca (read) dan mengubahnya (write), untuk mengubah ijin akses file tersebut agar hanya dapat memberikan ijin read, saya menggunakan perintah pada gambar kedua.

2. Buatlah suatu kelompok. Copy-kan /bin/sh ke home directory. Ketik "chmod +s sh". Cek ijin akses sh pada daftar direktori. Sekarang tanyakan ke teman satu kelompok anda untuk mengubah ke home directory anda dan menjalankan program ./sh dan menjalankan id command. Apa yang terjadi. Untuk keluar dari shell tekan exit.

```
root@bagusok:~/jb11# cp /bin/sh /home/anton
root@bagusok:~/jb11# cd /home/anton
root@bagusok:/home/anton# ls -l sh
-rwx----- 1 root root 129816 May  9 04:41 sh
root@bagusok:/home/anton# chmod +s sh
root@bagusok:/home/anton# ls -l sh
-rws--S--- 1 root root 129816 May  9 04:41 sh
root@bagusok:/home/anton# |
```

Analisa : Perintah chmod +s sh adalah perintah untuk mengubah file sh yang tadinya executable file menjadi socket link yang dapat dilihat perbedaannya saat kita lakukan ls -l sh maka nama file tersebut telah diberi tanda merah atau tanda yang berbeda dengan tanda sebelumnya. Selain itu perintah diatas juga menentukan hak izin akses untuk pengguna lain yang menggunakan file ini. Setelah itu saya lanjutkan dengan menjalankan perintah seperti ./sh.

3. Hapus sh dari home directory (atau setidaknya kerjakan perintah chmod -s sh)

```
root@bagusok:/home/anton# ls -l sh
-rwx----- 1 root root 129816 May  9 04:41 sh
```

Analisa : Perintah chmod -s sh ini dapat menghapus atau mengembalikan bentuk file yang telah diubah menjadi socket link ke mode file sebelumnya. Dimana kita mencabut hak izin akses kepada pengguna lain untuk file ini.

4. Modifikasi ijin akses ke home directory anda sehingga sangat privat. Cek apakah teman anda tidak dapat mengakses directory anda. Kemudian kembalikan ijin akses ke semula.

```
/home/anton
$ sudo chmod 000 /home/
[sudo] password for anton:
$ cd /home
-sh: 12: cd: can't cd to /home
$ ls -l home
ls: cannot access 'home': No such file or directory
$ sudo chmod 755 /home/
$ ls -l /home/
total 12
drwxr-xr-x 4 anton anton 4096 May  9 04:41 anton
drwxrwxr-x 3 bagus bagus 4096 Mar 26 04:11 bagus
drwxr-xr-x 2 root  root  4096 May  9 04:20 bagusok
$ █
```

Analisa : Perintah chmod 000 tersebut merupakan perintah untuk menghapus / menghentikan semua izin akses yang ada pada home baik itu izin untuk membaca, melihat dan lain-lain yang berhubungan dengan proses pengaksesan file atau directory. Untuk membuktikannya dapat dilakukan dengan perintah ls atau langsung dengan grafik pada home maka akan muncul bahwa data tidak dapat dibuka atau tidak memiliki izin. Sedangkan cara untuk

mengembalikan pengaturannya seperti semula dapat dilakukan dengan perintah `chmod 755`.

5. Ketikkan `umask 000` dan kemudian buatlah file yang bernama `world.txt` yang berisi beberapa kata "hello world". Lihat izin akses pada file. Apa yang terjadi? Sekarang ketikkan `umask 022` dan buatlah file bernama `world2.txt`. Apakah perintah tersebut lebih berguna

```
$ umask 000
$ nano world.txt
$ ls -l world.txt
-rw-rw-rw- 1 anton anton 14 May  9 04:58 world.txt
$
```

```
-rw-rw-rw- 1 anton anton 14 May  9 04:58 world.txt
$ umask 022
$ touch world2.txt
$ ls -ls
total 136
 4 drwxrwxr-x 2 anton anton  4096 May  9 04:23 app
128 -rwx----- 1 root  root 129816 May  9 04:41 sh
 0 -rw-r--r-- 1 anton anton      0 May  9 04:59 world2
.txt
 4 -rw-rw-rw- 1 anton anton    14 May  9 04:58 world.
txt
$
```

?

Analisa : \$ mask 000 pada perintah diatas berfungsi untuk mengatur izin akses file yang akan dibuat setelahnya dengan format pemberian izin akses untuk dapat membaca dan menulis atau melakukan perubahan yang ditandai dengan format `-rw -r -r--`, karena 2 adalah blangan octal dari read(r) yang apabila dijalankan menjadi seperti berikut:

6. Buatlah file yang bernama "hello.txt" pada home directory menggunakan perintah `cat -u >hello.txt`. Tanyakan ke teman Anda untuk masuk ke home directory Anda dan menjalankan `tail -f hello.txt`. Sekarang ketikkan beberapa baris dalam hello.txt. Apa yang terjadi pada layer teman Anda ?

```
txt
$ touch hello.txt
$ cat -u > hello.txt
^C
$ nano hello*
$ tail -f hello.txt
shjshjshks
█
```

Analisa : Kedua perintah diatas merupakan perintah untuk standar input dan output. Misalnya perintah `cat -u > hello.txt` ini merupakan perintah membelokan standar output ke file `hello.txt` dan untuk berhenti dapat dilakukan dengan menekan `Ctrl + C`.

## Kesimpulan:

- Sebuah file dapat kita manajemen atau dapat kita atur sesuai dengan tingkat keamanan yang ingin kita tentukan pada file.
- Untuk proses ini dapat kita lakukan dengan menggunakan manajemen system file dimana sebuah file atau directory dapat kita tentukan hak izin aksennya, guna untuk menjaga data yang kita miliki.
  - Atribut file adalah dimana sebuah file atau directori bisa eksis atau sebuah file mempunyai atribut yang berbeda antara sistem operasi satu dengan yang lainnya. Atribut file digunakan oleh sistem operasi untuk memisahkan tipe-tipe file. Setiap file di windows mempunyai atribut tersendiri sesuai dengan kepentingan file tersebut terhadap sistem. Misalnya, tipe file yang tidak boleh dihapus oleh user diberi attribut system, jadi ketika file manager (windows explorer) akan menampilkannya, file beratribut sistem tersebut tidak akan ditampilkan.
  - Ijin akses bisa dibidang sebuah hak yang diberikan pada seluruh User atau file untuk melakukan sesuatu akses. Jadi Hak akses ini dapat diatur sedemikian rupa supaya file atau direktori itu hanya bisa di read, atau write, dan execute. Selain itu kita juga bisa mengatur bagaimana hak akses dari setiap user pada suatu file, atau direktori.
- User mask adalah kemampuan sistem operasi berbasis Linux untuk menambahkan pengaturan hak akses default untuk file-file baru (termasuk folder).

