

Bab 5

Sistem File Linux Lanjutan

Program D3 PJJ PENS ITS

Tujuan

- Setelah mempelajari materi dalam bab ini, mahasiswa diharapkan mampu:
 - Memahami bagaimana informasi sistem file diorganisasi
 - Mengerti fungsi *dentries* dan *inodes*
 - Memahami bagaimana perintah *cp*, *mv* dan *rm* bekerja pada level *inodes*
 - Mengerti bagaimana *symbolic link* dan *hard link* bekerja
 - Mengetahui bagaimana akses ke media *removable*
 - Dapat membuat file *archive* menggunakan *tar* dan *gzip*

Materi

- Partisi dan Sistem File
- Inode
- Direktori
- Symbolic Link
- Tipe File
- Memeriksa Ruang Bebas pada Disk
- Media Removable
- Membuat File Archive
- Kompresi File

Partisi dan Sistem File

- Disk drive dan media penyimpan dibagi ke dalam beberapa partisi. Partisi diformat dalam suatu sistem file.
- Sistem file adalah struktur data penulisan ke media sehingga user dapat menyimpan dan mengakses file.

Ext3

- Sistem file untuk Linux Red Hat adalah *Third Extended Linux Filesystem* atau ext3.
 - Sistem file ini merupakan versi pengembangan dari ext2.
 - Ext3 dibuat lebih stabil dan lebih kuat.
 - Ext2/3 mendukung berbagai fitur yang tidak terdapat pada sistem file Unix lain seperti *extended attribute* (EAs) dan *POSIX Access Control Lists* (ACLs).
- Linux Red Hat mendukung lebih dari 20 tipe sistem file. Sistem file lain yang digunakan untuk floppy disk dan iso9660 yang digunakan CD adalah msdos. Sistem file ext2 juga biasa digunakan untuk memformat floppy karena lebih kuat dan fleksibel daripada msdos dan menggunakan ruang disk yang lebih kecil untuk struktur datanya dibandingkan ext3.

Sistem File Lain

- Linux Red Hat mendukung lebih dari 20 tipe sistem file.
- Sistem file lain yang digunakan untuk floppy disk dan iso9660 yang digunakan CD adalah msdos.
- Sistem file ext2 juga biasa digunakan untuk memformat floppy karena lebih kuat dan fleksibel daripada msdos dan menggunakan ruang disk yang lebih kecil untuk struktur datanya dibandingkan ext3.

Inode

- Inode berisi metadata tentang file.
 - Tipe file
 - Ijin akses file
 - Jumlah link: jumlah nama file yang dihubungkan dengan nomor inode
 - Nomor ID user dari pemilik file
 - Nomor ID grup dari pemilik grup
 - Waktu, terdiri dari waktu akses terakhir, waktu modifikasi terakhir dan waktu perubahan inode
 - Lokasi data pada hard disk
 - Metadata lain tentang file

Direktori

- Komputer mereferensikan suatu file sebagai nomor inode.
- User mereferensikan suatu file berdasarkan nama file.
- Direktori memetakan nama file menjadi nomor inode.
- Bila nama file direferensi dengan perintah atau aplikasi, Linux mereferensi direktori dimana file berada, menentukan nomor inode yang berhubungan dengan nama file dan mencari informasi inode dalam tabel inod dan jika user mempunyai ijin akses, menghasilkan nama file.

Perintah ls -li

```
$ ls -li
total 28
80788 -rw-r----- 1 student student 5120 Sep 18 11:26 myData
37777 drwxr-x--- 2 student student 4096 Sep 18 11:25 newStuff
80787 -rw-r----- 1 student student 1536 Sep 18 11:26 notes
```

Nomor inode untuk file `myData` adalah 80788.

Inode dan Direktori

Nama

Dihubungkan dengan inode dari direktori parent

report

Metadata Inode

Properti dan pointer blok ke disk

```
Type: Directory
drwxrwxrwx prince prince
Blocks: 1 Links: 4
Access: 2006-12-16 16:20:42
Modify: 2006-12-16 16:20:42
Change: 2006-12-16 16:20:42
```

Isi

Untuk direktori : nama/daftar inode
Untuk file : data file

"."	592253
".."	249482
"html"	592255
"text"	592254

Perintah cp

- langkah-langkah yang dilakukan :
 - Mengalokasikan nomor inode baru, menempatkan entri baru dalam tabel inode
 - Membuat entry yang menghubungkan nama dengan nomor inode
 - Meng-copy data ke dalam file baru.

```
$ cp penguin tux
$ ls -li penguin tux
246688 -rw-rw---- 1 student student 26 Nov 12 20:56 penguin
246575 -rw-rw---- 2 student student 26 nov 12 20:25 tux
```

Perintah mv

- Jika tujuan perintah mv sama dengan sistem file asal, langkah-langkah yang dilakukan :
 - Membuat masukan ke direktori baru dengan nama file baru
 - Menghapus masukan ke direktori lama dengan nama file lama

```
$ ls -li tux
246575 -rw-rw-r-- 1 student student 26 Nov 12 20:56 tux
$ mv tux fedora
$ ls -li fedora
246575 -rw-rw-r-- 1 student student 26 Nov 12 20:56 fedora
```

Perintah rm

- Langkah-langkah yang dilakukan:
 - Mengurangi jumlah link, kemudian membebaskan nomor inode untuk digunakan kembali
 - Menempatkan blok data pada daftar blok yang bebas
 - Menghapus masukan direktori
- Data secara aktual tidak dihapus, tetapi akan di *override* jika blok data digunakan file lain.

Symbolic Link

- File yang menunjuk ke file lain.

```
$ ls -l pf  
lrwxrwxrwx 1 student student 11 Nov 12 20:56 pf -> /etc/passwd
```
- Tipe file l pada izin akses menunjukkan file pf adalah *symbolic link*.
- Isi file pf sama dengan file yang direferensi yaitu /etc/passwd.
- Izin akses dari file duplikat tidak sama dengan file asal.
- Ukuran dari file tidak sama dengan ukuran file asal.
- Ukuran file duplikat adalah jumlah karakter pada nama path.
- Perintah:

`ln -s fileAsli fileDuplikat`

Hard Link (1)

- Path yang mereferensi inode, sehingga semua file pasti dilakukan *hard link* minimal satu kali.
- Memungkinkan untuk memberi nama file lebih dari satu yang menunjuk ke nomor inode yang sama.
- Perintah:

In *fileAsli fileDuplikat*

Hard Link (2)

- Contoh

```
$ ls -li fedora redhat
```

```
246575 -rw-rw-r-- 2 student student 26 Nov 12 20:56 fedora
```

```
246575 -rw-rw-r-- 2 student student 26 Nov 12 20:56 redhat
```

- File *fedora* dan *redhat* mempunyai nomor inode yang sama, hanya ada satu file tetapi terdapat dua titik masukan.
- Jika satu nama path dihapus dengan perintah *rm*, jumlah link dikurangi satu.
- Jika nama path akhir dihapus, jumlah link menjadi 0 dan dihapus.

Tipe File

- Ordinary file
- Direktori
- Link File
- Character Device (Peralatan I/O)
- Block Device (Peralatan I/O)
- Named Pipe (FIFO)
- Socket

Memeriksa Ruang pada Disk

- Perintah `df` dan `du` digunakan untuk melihat berapa besar ruang disk digunakan.
- Perintah `df` melaporkan jumlah total kilobyte, jumlah kilobyte yang digunakan dan kilobyte yang bebas dari sistem file.
- Perintah `du` melaporkan jumlah kilobyte yang digunakan per direktori.

Media Removable

- Sebelum mengakses data yang baru dimasukkan ke dalam media *removable* (floppy disk, CD, zip disk) sistem file dari media harus di-*mount*.
- Yang dimaksud *mounting* adalah membuat sistem file asing sebagai bagian dari pohon direktori utama.
- Perintah :

```
$ mount /media/floppy mount /media/cdrom
```
- Untuk mengakses cd writer dapat melakukan *mount* pada /media/cdrecorder.

Perintah mtools

- perintah DOS standar.
- Drive a: dipetakan ke floppy.
- Manipulasi media *removable* dapat menggunakan perintah DOS dengan menambahkan huruf "m" di depannya

```
$ mdir a:  
$ mcopy myfile a:
```

Mounting CD dan DVD

- Bila menggunakan sistem X Window, masukkan CD ke dalam drive dan secara otomatis akan dilakukam *mount* cd dan menambahkan ke desktop.
- Titik *mount* dihubungkan dengan perangkat bila perlu melakukan *mount* secara manual tergantung yang digunakan CD/DVD reader atau writer. Reader harus di-*mount* pada */media/cdrom* dan writer pada */media/cdrecorder*.
- Bila selesai menggunakan disk dapat dihentikan melalui icon desktop dengan melakukan klik kanan dan memilih "Eject" atau menjalankan perintah eject dari prompt. Jika hanya terdapat satu perangkat, menjalankan perintah eject sudah cukup, tetapi bila terdapat lebih dari satu perangkat, harus menambahkan node perangkat sebagai argument.

Mounting Media USB

- Disk USB dideteksi kernel sebagai perangkat SCSI, sehingga dapat direferensi melalui */dev/sda*, */dev/sdb* dan seterusnya.
- Secara normal dikenali sebagai partisi pertama (seperti */dev/sda1*) tetapi mungkin juga menggunakan partisi lainnya.
- Di-mount pada */media/Device ID* yang digunakan oleh vendor.

Mounting Floppy Disk

- Harus dilakukan secara manual melalui /media/floppy.
- Floppy disk DOS dapat diakses dengan mtools dan menggunakan perintah DOS dengan lebih transparan, misalnya:
\$ mdir a:
\$ mcopy myfile a:
- Tipe mem-format floppy disk :
 - Level rendah
\$ fdformat /dev/fdH1440
 - Format sistem file
 - Perintah mkfs

Membuat File Archive (1)

- Untuk backup, menyimpan dan mengirim beberapa file melalui jaringan.
- Perintah standar Linux untuk membuat file archive adalah tar yang merupakan kepanjangan dari *tape archive*.
- Sintak
tar cvf nama_file_archive nama_file_asal
c : membuat satu file *archive* baru
v : menghasilkan pesan file yang dihasilkan
f nama_file_archive : nama file *archive* baru.
- File *archive* yang dihasilkan mempunyai ekstensi .tar.

Membuat File Archive (2)

- Untuk melihat file *archive* dapat menggunakan sintak:

```
tar tf nama_file_archive  
tar tvf nama_file_archive
```

pilihan v menampilkan daftar yang panjang.

- Untuk membuka file *archive* dapat menggunakan sintak:

```
tar xvf nama_file_archive
```

Kompresi File

- File yang jarang digunakan biasanya dilakukan kompresi.
- Keuntungannya adalah mendapatkan ukuran file yang lebih kecil.
- File teks dapat dikompresi lebih dari 75%.
- File biner biasanya tidak dapat di-kompres dengan baik sekitar 0-25%.
- File *archive* tar biasanya juga dilakukan kompresi.

gzip

- Bila suatu file dilakukan kompresi dengan gzip, nama file menjadi berekstensi .gz.
- Untuk membuka file kompresi dapat dilakukan gunzip dan membuat file asli.
- Untuk file teks, perintah ini mampu melakukan kompresi lebih dari 75%.
- Sintak yang digunakan untuk melakukan kompresi dan membuka file kompresi:

```
gzip nama_file  
gunzip nama_file
```

bzip2

- Utilitas Linux yang lebih baru untuk kompresi adalah bzip2.
- Hasil kompresi adalah file berekstensi .bz2 dan untuk membukan file kompresi menggunakan perintah bunzip2.
- Sintak yang digunakan untuk melakukan kompresi dan membuka file kompresi:

```
bzip2_file  
bunzip2 nama_file
```