

Praktikum

Manajemen File System

POKOK BAHASAN:

- ✓ Partisi
- ✓ Filesystem
- ✓ Swap
- ✓ fstab
- ✓ mount/umount
- ✓ diskquota
- ✓ mengecek dan memperbaiki filesystem

TUJUAN BELAJAR:

- ✓ Mahasiswa dapat melakukan partisi dan membuat file system
- ✓ Mahasiswa dapat menangani partisi swap
- ✓ Mahasiswa dapat menerapkan diskquota
- ✓ Mahasiswa dapat maintain filesystem yang ada
- ✓ Mahasiswa dapat melakukan mount dan unmount file sistem
- ✓ Mahasiswa dapat mengecek dan memperbaiki filesystem

DASAR TEORI:

Apa itu partisi ?

Partisi adalah membagi hard disk menjadi beberapa ruang logik yang lebih kecil. Partisi terdiri atas blok-blok yang berurutan yang dianggap sebagai satu bagian yang independen. Tabel partisi adalah tabel yang menghubungkan bagian dari hard disk ke partisi.

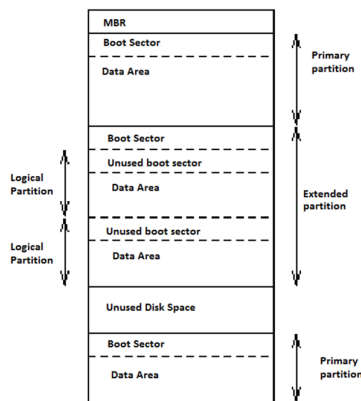
Mengapa butuh multiple partisi ?

- Untuk mengamankan data. Bila terjadi kerusakan pada filesystem, biasanya bersifat lokal terhadap partisi tertentu. Partisi yang lain tidak terpengaruh dengan kerusakan filesystem.
- Meningkatkan efisiensi. Anda dapat memformat partisi dengan ukuran blok yang berbeda, tergantung kebutuhan. Jika data anda berukuran kecil (misal, kurang dari 1k), jika partisi anda berukuran 4k/blok, maka anda membuang 3k/blok. Karena itu, penting untuk memiliki ukuran blok yang sesuai dengan ukuran file.

- Membatasi pertumbuhan data. Seringkali user atau proses dapat memakan/mengonsumsi begitu banyak space di disk sehingga OS tidak punya ruang lagi untuk operasi kesehariannya. Ini bisa menyebabkan OS mengalami collabs. Dengan membatasi ukuran space data, anda bisa mencegah OS kehabisan space untuk operasi kesehariannya.

Tipe partisi

- Partisi Primary :
 - Merupakan jenis partisi utama diharddisk untuk system operasi umumnya. Partisi primary hanya bisa dibuat maksimal 4 partisi, hal ini sangat berbeda dengan sistem DOS yang hanya mengijinkan satu jenis partisi primary untuk system. Jenis partisi ini menempati nomor partisi 1, 2, 3 dan 4. Misalnya harddisk sda bisa dibuat sda1, sda2, sda3 dan sda4.
- Partisi Extended :
 - Partisi Extended berfungsi untuk mengatasi keterbatasan pembagian partisi yang hanya melayani sampai 4 partisi. Partisi Extended tidak menangani pengolahan data secara langsung. Untuk dapat menggunakannya, kita harus menciptakan Partisi Logical terlebih dahulu.
- Partisi Logical :
 - Partisi yang akan selalu dibuat didalam partisi extended. Nomor partisi akan selalu dimulai 5 dan seterusnya. Jika ada 3 jenis partisi logical maka masing-masing akan menempati sda5, sda6, dan sda7.



Gambar 1. Partisi Hard Disk (primary,extended dan logical)

Skema Partisi Rekomendasi

| Partisi | Min | Max |
|---------|-----------|------------------------------------|
| / | 150M-310M | 2G-8G |
| /usr | 600M-750M | 5G-6G |
| /var | 30M-40M | 300M-500M |
| /home | 100M/user | Tergantung kebutuhan user |
| /boot | 100M | Tergantung jumlah OS yang disimpan |
| /tmp | 40M | 100M |

Software Partisi

- sfdisk : fdisk versi teks based
- cfdisk : fdisk versi curses-based (GUI)
- parted : GNU partititon editor
- Partition Magic : Utilitas komersial untuk membuat, merubah ukuran, menggabungkan dan mengkonversi partisi tanpa menghilangkan data
- Disk Drake : utilitas Perl/GTK untuk membuat, merubah ukuran dan mengkonversi partisi

File System

Filesystem adalah metode untuk mengorganisasi data agar secara efisien mampu menyediakan kapabilitas untuk menyimpan, mencari dan mengupdate data serta melakukan manajemen space kosong pada disk, sesuai dengan karakteristik devais.

Sebuah harddisk dipartisi terlebih dahulu sebelum diformat sesuai dengan kebutuhan user menggunakan berbagai filesystem yang tersedia. Ketika disk diformat, filesystem dibuat dengan tujuan memudahkan user untuk mengakses dan mengorganisir file di dalam hard disk.

Tiap partisi diformat dengan satu filesystem yang ada. Sebuah file system memiliki kelebihan dan kekurangannya sendiri. Berikut ini diberikan tabel tipe filesystem yang ada (tidak hanya Linux), bersama tahun penciptaannya, OS yang dipakai, maksimum ukuran file dan maksimum ukuran partisi, serta kemampuan journaling.

| FS Name | Year Introduced | Original OS | Max File Size | Max FS Size | Journaling |
|-----------|-----------------|-------------|---------------|--------------|------------|
| FAT16 | 1983 | MSDOS V2 | 4GB | 16MB to 8GB | N |
| FAT32 | 1997 | Windows 95 | 4GB | 8GB to 2TB | N |
| HPFS | 1988 | OS/2 | 4GB | 2TB | N |
| NTFS | 1993 | Windows NT | 16EB | 16EB | Y |
| HFS+ | 1998 | Mac OS | 8EB | ? | N |
| UFS2 | 2002 | FreeBSD | 512GB to 32PB | 1YB | N |
| ext2 | 1993 | Linux | 16GB to 2TB4 | 2TB to 32TB | N |
| ext3 | 1999 | Linux | 16GB to 2TB4 | 2TB to 32TB | Y |
| ReiserFS3 | 2001 | Linux | 8TB8 | 16TB | Y |
| ReiserFS4 | 2005 | Linux | ? | ? | Y |
| XFS | 1994 | IRIX | 9EB | 9EB | Y |
| JFS | ? | AIX | 8EB | 512TB to 4PB | Y |
| VxFS | 1991 | SVR4.0 | 16EB | ? | Y |
| ZFS | 2004 | Solaris 10 | 1YB | 16EB | N |

Journaling

Journaling adalah kemampuan untuk menyimpan journal atau log setiap perubahan yang dilakukan terhadap filesystem sehingga mudah untuk melakukan rekonstruksi bila terjadi system crash atau listrik mati. Level journaling dari filesystem dapat diatur menjadi sejumlah level logging, tergantung dari kebutuhan dan performansi system.

Kelemahan journaling terletak pada banyaknya log setiap kali terjadi disk write. Hal ini dapat diatasi, jika anda menjalankan sistem dimana performansi disk diatas rata-rata, sehingga delay akibat penulisan log menjadi rendah .

Tabel dibawah memuat beberapa filesystem yang ada di beberapa media fisik dan OS yang berbeda, mulai dari AIX, freeBSD, sampai dengan Tru64.

Tabel 2. Tipe file system pada Unix dalam beberapa media dan OS yang berbeda

| Use | AIX |
|---------------|---------------|
| Default local | jfs or jfs2 |
| NFS | nfs |
| CD-ROM | cdrfs |
| Swap | not needed |
| DOS | not supported |
| /proc | procfs |
| RAM-based | not supported |
| Other | |

Tabel 3. Tipe Filesystem Linux dan keterangannya

| Filesystem | Keterangan |
|---|---|
| ext2 Second Extended Filesystem | Dikembangkan dari versi extended, merupakan filesystem GNU/Linux yang terkenal stabil. Kelemahannya adalah tidak memiliki kemampuan journaling atau barrier. Karena tidak memiliki kemampuan journaling, akibatnya sering terjadi dataloss saat system crash atau saat listrik mati, Pengecekan filesystem ext2 berjalan lambat, karena itu tidak menyenangkan untuk directory / dan /home. Ext2 dapat dikonversi menjadi ext3. |
| ext3 — Third Extended Filesystem | Pengembangan dari filesystem ext2 dengan kemampuan journaling barrier. Tidak kompatibel dengan ext2, sangat stabil dan handal |
| ext4 — Fourth Extended Filesystem | Filesystem ext4 ini kompatibel dengan ext2 dan ext3. Mendukung partisi dengan ukuran 1 exabyte (i.e. 1,048,576 terabytes) dan ukuran file sampai 16 terabytes. Tiap partisi bisa memiliki ukuran sampai 32000 directory (jumlah maksimum directory dari ext3) sampai tidak terhingga. Ext4 juga mendukung kemampuan defragmentasi online. |
| Iso9660 | Filesytem standar pada CDROM. Memiliki kemampuan untuk menyimpan filename yang panjang secara otomatis. |
| smbfs | Filesystem berbasis jaringan yang memungkinkan OS Linux melakukan sharing file dengan OS MS Windows. Filesystem ini juga kompatibel sengan protokol filesharing dari Windows |
| /proc | Filesystem proc ini digunakan untuk memudahkan user mengakses struktur data kernel. Filesystem proc ini diakses lewat beberapa perintah yang biasa digunakan untuk file, misalnya cat |
| reiserfs | Filesystem yang paling tangguh. Kemampuan journaling membuat data loss semakin jarang terjadi. Setiap kali transaksi terjadi, record disimpan untuk transaksi tersebut dan transaksi sebelumnya, Akibatnya, jika terjadi kerusakan, filesystem mudah direkonstruksi dengan cepat. |
| VFAT — Virtual File Allocation Table | Perpanjangan dari filesystem FAT, yaitu FAT 32. Mengdukung ukuran partisi yang lebih besar dari FAT32. Hampir semua OS mendukung filesystem VFAT, karena itu banyak digunakan untuk sharing data dengan berbagai OS dan dipakai sebagai format filesystem pada solid-state memory cards. |
| nfs | Filesystem berbasis jaringan yang memungkinkan sharing file dengan beberapa komputer agar dapat saling berbagi file. |

Untuk melihat lebih fitur ext2, ex3 dan ext dibuka link berikut :

di

detail bisa

<https://www.cyberciti.biz/tips/linux-convert-ext3-to-ext4-file-system.html>

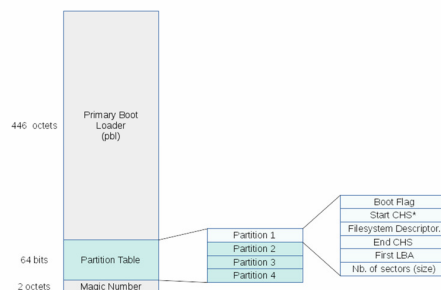
MBR dan GPT

Sebelum anda dapat menggunakan disk, anda harus melakukan partisi dan memformat disk tersebut. MBR dan GPT adalah cara untuk menyimpan informasi partisi pada disk. Informasi yang disimpan adalah dimana awal dari partisi, sehingga OS mengetahui nomor partisi dari sektor tertentu dan partisi mana yang bersifat bootable. Karena itu anda harus memilih antara MBR dan GPT sebelum melakukan partisi pada disk

Struktur MBR

MBR dipergunakan pertama kali oleh IBM PC DOS 2.0 pada tahun 1983. Disebut MBR karena MBR adalah sektor boot khusus yang berada di sektor pertama dari hard disk. MBR berisi : boot loader untuk OS yang diinstall & informasi mengenai tabel partisi logik dari hard disk. Boot loader berisi kode berukuran kecil (446 oktet) yang bertugas melakukan loading boot loader yang berukuran lebih besar dan berada di partisi lain dari hard disk. Jika anda memiliki Windows, boot loader dari Windows disimpan di MBR. Karena itu, jika MBR tidak sengaja terhapus dan Windows anda tidak muncul, maka MBR harus diinstall ulang. Jika anda menggunakan Linux, bootloader yang digunakan adalah GRUB yang disimpan di MBR.

MBR memiliki keterbatasan, yaitu hanya bekerja pada disk berukuran < 2TB. MBR hanya mendukung 4 partisi utama (primary partition). Jika anda ingin memiliki lebih dari 5 partisi primary, maka salah satu partisi primary harus diubah menjadi partisi extended, dan membuat partisi logik di dalam partisi extended



GPT

GPT singkatan dari GUID Partition Table. Ini adalah standar baru yang mulai banyak dipakai menggantikan MBR. GPT memanfaatkan fitur dari UEFI, yang mengganti BIOS menjadi lebih modern. GPT menggunakan GUID (globally Unique identifier) yang merupakan string random yang panjang sehingga tiap partisi memiliki identifier sendiri.

GPT dapat menangani disk dengan ukuran yg jauh lebih besar dari MBR. Ukuran maksimum yang dapat dihandle GPT tergantung dari OS dan file system yang digunakan. GPT juga memungkinkan anda memiliki banyak partisi, tergantung pada OS yang digunakan. Windows mendukung hingga 128 partisi dengan GPT. Dengan cara ini anda tidak perlu menggunakan partisi extended lagi.

FDISK

```
root # fdisk -l
Disk /dev/sda: 200.0 GB, 200049647616 bytes
255 heads, 63 sectors/track, 24321 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Disklabel type: dos
Disk identifier: 0x1c841c84

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1          1         574     4604008+    c   W95 FAT32 (LBA)
Partition 1 does not end on cylinder boundary.
/dev/sda2 *        575        8967     67416772+    7   HPFS/NTFS
/dev/sda3          8968       24321    123331005    f   W95 Ext'd (LBA)
/dev/sda5          8968       9229     2104483+    82   Linux swap / Solaris
/dev/sda6          9230      11840     20972826    83   Linux
/dev/sda7         11841      24321    100253601    83   Linux
```

- Partisi 1 adalah partisi yang digunakan untuk melakukan restorasi
- Partisi 2 berisi Windows XP
- Partisi 3 adalah partisi extended, yang berawal dari akhir /dev/sda1, sampai akhir dari disk. Partisi 3 berisi :
 - Partisi 5 berisi partisi swap
 - Partisi 6 berisi partisi /
 - Partisi 7 berisi partisi /home
- Disklabel type: dos, menunjukkan bahwa tipe MBR yang digunakan adalah DOS
- /dev/sda2 * menunjukkan bahwa /dev/sda2 mengandung partisi boot (tempat dimana boot manager disimpan)

ETC/FSTAB

```
# /etc/fstab: static file system information.
#
# <file system> <mount point> <type> <options>      <dump> <pass>
/dev/hda2      /           ext2     defaults      0        1
/dev/hda3      none        swap     sw            0        0
proc           /proc       proc     defaults      0        0
```

- Filesystem : berisi devais yang memuat file system tersebut
- Mount point : berisi direktori dimana filesystem di-mount
- Filesystem type : berisi tipe filesystem.
- Options :

berisi opsi yang diberikan ketika filesystem di-mount. Jika ada beberapa opsi, maka masing-masing opsi dipisahkan koma(,) tanpa ada spasi. Beberapa opsi yang biasa dipakai :

- defaults : opsi standar yang dipakai oleh kebanyakan filesystem
 - errors=remount-ro : berarti jika terjadi error saat filesystem dicek, maka filesystem akan dimount dengan mode readonly. Dengan cara ini, administrator dapat menganalisa error tanpa menimbulkan kerusakan lebih jauh
 - sw : filesystem dimount sebagai partisi swap
 - ro: filesystem dimount sebagai partisi read-only. biasanya digunakan pada CD ROM atau devai lainnya.
 - noauto : filesystem tidak akan di-mount secara otomatis saat system startup
 - user : memungkinkan user lain untuk melakukan mount filesystem
-
- Dump flag : menunjukkan apakah perintah dump dapat dipakai untuk menciptakan backup filesystem. Filesystem dengan dump flag 0 atau kosong, tidak akan melakukan backup
 - Pass : menunjukkan urutan filesystem mana yang dicek saat booting. Nilai pass kosong atau 0 berarti filesystem tidak akan dicek.

UUID

UUID adalah sejumlah string yang digunakan untuk mengidentifikasi partisi pada block device. UUID disimpan dalam partisi yang diidentifikasi. Keuntungan menggunakan UUID adalah UUID bersifat unik, tidak mungkin tertukar.

Percobaan 1. Perintah du dan df

1. Untuk melihat space yang dibutuhkan oleh tiap file dan direktory, kita dapat menggunakan du -h

```
#du -h /
```

Untuk melihat total keseluruhan space dari file dan direktory, kita dapat menggunakan du -s

```
#du -sh /
```

Untuk melihat space dari seluruh file dan direktory, kita dapat menggunakan du -a. Untuk mengurutkan ukurannya dari terkecil sampai terbesar, kita dapat menggunakan sort -n

```
#du -a / | sort -n
```

Untuk mengetahui isi dari subfolder /, maka digunakan --max-depth=1

```
#cd /
```

```
#du -h --max-depth=1 | sort -hr
```

Untuk memberitahu berapa ukuran file dan directory terbesar

```
#du -sh -- * | sort -hr
```

2. Untuk melihat informasi mengenai pemakaian disk, kita dapat menggunakan perintah `df -h`, opsi `-h` berarti menggunakan format yang dapat dibaca manusia (human readable format)

```
#df -h
```

Untuk melihat informasi mengenai seluruh filesystem yang ada

```
#df -a
```

Untuk melihat informasi mengenai filesystem yang digunakan

```
#df -T
```

Untuk melihat informasi mengenai filesystem tertentu

```
#df -t ext4
```

Untuk melihat informasi mengenai filesystem kecuali filesystem ext4

```
#df -x ext4
```

Untuk menentukan informasi yang hendak ditampilkan, bisa digunakan opsi `--output` dan `-x`

```
$ df -h --output=source,fstype,size,used,avail,pcent,target -x tmpfs -x devtmpfs
```

Untuk melihat informasi mengenai filesystem yang sesungguhnya pada `/dev`

```
$ df -h | grep ^/dev
```

Percobaan 2. Utilitas untuk mengecek hard disk

1. `fdisk`

```
#fdisk -l
```

2. `sfdisk`

```
#sfdisk -l
```

3. `lsblk`. `lsblk` digunakan untuk mendisplay informasi mengenai block device yang available pada sistem

```
#lsblk  
#lsblk -n  
#lsblk -m  
#lsblk -ln
```

4. `blkid`. `blkid` adalah perintah yang digunakan untuk mengecek UUID dari block device

```
#blkid /dev/sda  
#blkid -po udev /dev/sda1
```

5. `hwdm`

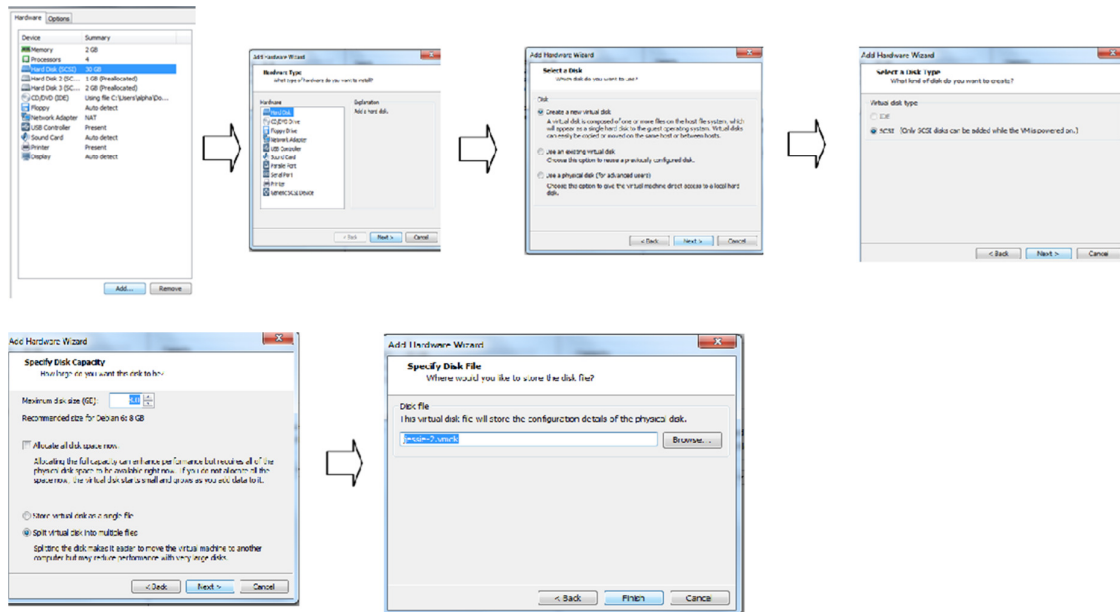
```
$ hwdm --block --short
```

Percobaan 3. Membuat partisi disk

1. Jika anda menggunakan vmware, maka klik virtual machine settings.

- Klik Hard Disk (SCSI), klik Add.
- Pada Add Hardware Wizard, klik Hard Disk dan Next.
- Pada Select Disk, klik Create a New Virtual Disk.

- Pada Select disk type, secara default menggunakan SCSI.
- Pada Specify disk capacity, masukkan ukuran disk anda pada Maximum Disk Space. Klik Allocate Disk Space Now. Pilih Split Disk into multiple files. Klik Next.
- Masukkan nama disk jika anda menginginkan nama yang berbeda, jika tidak, biarkan tetap seperti semula. Klik Finish



Buatlah dua partisi baru dengan ukuran 1 G dan 2 G

2. Jika sudah booting linux anda.
3. Cek apakah partisi baru anda sudah dikenali dengan

```
#fdisk -l
```

4. Sekarang buatlah partisi baru dengan fdisk.

Jika anda menggunakan virtual system, nama device anda adalah /dev/sda, /dev/sdb, /dev/sdc, /dev/sdd, dan seterusnya. Jika anda menggunakan dual boot, nama device anda adalah /dev/hda, /dev/hdb, /dev/hdc, /dev/hdd, dan seterusnya.

```
#fdisk /dev/sdb
```

- Pada baris Command (m for help) , pilih n untuk membuat partisi baru.
- Pada partition type pilih : p untuk primary
- Pada partition number, pilih 1. Karena pada /dev/hdb, ini adalah partisi pertama. Kita boleh memiliki sampai 4 buah partisi primary dalam /dev/sdb.
- Pilih blok awal dari partisi, yaitu 2048
- Pilih blok akhir dari partisi. Karena kita menggunakan seluruh /dev/sdb dalam satu partisi. Tulis saja blok terakhir dari /dev/sdb

- Pada baris Command (m for help), pilih w untuk menuliskan partisi tersebut

Lakukan hal yang sama untuk /dev/sdc yang berukuran 2 G. Bagi menjadi 2 buah partisi, yang masing-masing berukuran 1 GB. Pada partisi 1 /dev/sdc1, sektor pertama adalah 2048 dan sektor akhir adalah +1G. Pada partisi 2 /dev/sdc2, sektor pertama dan terakhir gunakan nilai default.

5. Reboot system anda

6. Lakukan fdisk -l . Partisi kita telah terbentuk

7. Sekarang kita akan memformat partisi dengan perintah mkfs

```
#mkfs -t ext2 /dev/sdb
#mkfs -t ext4 /dev/sdc1
#mkfs -t ext3 /dev/sdc2
```

Cek filesystemnya

```
# df -T
```

Walaupun telah terformat sebagai ext2,ext3 dan ext4 namun tipe file akan tertulis tmpfs.

Tmpfs adalah nama filesystem sementara karena filesystem belum dimount ke system

8. Coba /dev/sdb, /dev/sdc1, /dev/sdc2. Berhasilkah ? pasti gagal karena /dev/sdb, /dev/sdc1 dan /dev/sdc2 tidak bisa diakses langsung oleh user.

```
#cd /dev/sdb
#cd /dev/sdc1
#cd /dev/sdc2
```

9. Untuk dapat mengakses /dev/sdb, /dev/sdc1 dan /dev/sdc2, perlu dibuat mount point. Buat mount point di direktory puncak, yaitu: /

```
#cd /
#mkdir /mnt/sdb
#mkdir /mnt/sdc1
#mkdir /mnt/sdc2
```

10. Lakukan mount filesystem

```
# mount -t ext2 /dev/sdb /mnt/sdb
# mount -t ext4 /dev/sdc1 /mnt/sdc1
# mount -t ext3 /dev/sdc2 /mnt/sdc2
```

11. Cek apakah filesystem baru telah ter-mount

```
# df -k
```

Cek filesystemnya

```
# df -T
```

12. Coba buat file dan direktori di /mnt/sdb, /mnt/sdc1, /mnt/sdc2. Berhasilkah ?

```
#cd /mnt/sdb
#touch file-sdb
#mkdir new-sdb
```

13. Masukkan filesystem baru di /etc/fstab.

```
#nano /etc/fstab
```

Masukkan baris berikut di bagian paling bawah

```
/dev/sdb /mnt/sdb ext2 defaults 0 0
/dev/sdc1 /mnt/sdc1 ext4 defaults 0 0
/dev/sdc2 /mnt/sdc2 ext3 defaults 0 0
```

15. Reboot linux

16. Buka terminal, ketik

```
# df -k
```

Lihat apakah /dev/sdb, /dev/sdc1 dan /dev/sdc2 telah termount oleh Linux secara otomatis

Percobaan 4. Mengubah tipe filesystem

1. Mengecek file system. Masuklah sebagai user root dan ketik perintah dibawah

```
# df -hT | awk '{print $1,$2,$NF}' | grep "^/dev"
```

Pada /dev/sdb, /dev/sdc1, /dev/sdc2. Buatlah file dan direktori. Lakukan hal yang sama untuk /dev/sdc1 dan /dev/sdc2

```
#cd /mnt/sdb
#touch sdb
#mkdir sdb-dir
```

2. Sekarang kita mengubah /dev/sdb dari ext2 menjadi ext3. Ketika berpindah dari ext2 ke ext3, ditambahkan fasilitas journaling pada filesystem. Ini ditunjukkan opsi j pada tune2fs

```
#tune2fs -j /dev/sdb
```

Cek tipe partisi baru. Adakah perubahan dari ext2 ke ext3 ?

```
# mount | grep "/dev"
```

Jika belum, sekarang lakukan unmount dan mount kembali /dev/sdb. Haini digunakan untuk merefresh tabel partisi system.

```
#umount /dev/sdb
#mount /dev/sdb /mnt/sdb
```

Cek tipe partisi baru dengan perintah df diatas. Adakah perubahan dari ext2 ke ext3 ?

```
# fsck -N /dev/sdb
```

Sekarang lakukan cek filesystem untuk memastikan bahwa filesistem anda OK. Untuk melakukan ini pastikan bahwa file system anda dalam keadaan unmount. Opsi -p berarti jika ada error e2fsck akan memberitahu administrator, Opsi -f berarti e2fsck akan tetap melakukan cek walaupun filesistem dalam keadaan baik. Jika e2fsck menunjukkan jumlah file dan sektor yang dicek dan tidak memberikan laporan error apapun, berarti file sistem anda dalam keadaan baik

```
#umount /dev/sdb
#e2fsck -pf /dev/sda1
```

Lakukan mount kembali

```
#mount /dev/sdb /mnt/sdb
```

Cek apakah file dan direktori anda masih ada ?

3. Sekarang, kita akan mengubah /dev/sdc2 dari tipe filesystem ext3 menjadi ext4.

```
#tune2fs -O extents,uninit_bg,dir_index /dev/sdc2
```

Dengan menggunakan fsck, kita dapat mengecek filesystem. Jangan lupa lakukan unmount dulu.

```
#umount /dev/sdc2
```

```
#fsck -pf /dev/sdc2
```

Lakukan mount filesystem

```
# mount -t ext4 /dev/sdc2 /mnt/sdc2
```

Cek tipe partisi baru dengan perintah df diatas. Adakah perubahan dari ext3 ke ext4 ?

```
# df -hT
```

Cek apakah file dan direktori anda masih ada ?

4. Sebelumnya kita ubah kembali /dev/sdb dari ext3 ke ext2

```
# tune2fs -O ^has_journal /dev/hda5
```

Pasti error, karena filesystem harusnya diunmount terlebih dulu. Unmount filesystem

Jika sudah, lakukan perintah diatas kembali. Berpindah dari ext3 ke ext2 berarti

menghilangkan fitur journaling pada ext3. Cek apakah filesystem OK.

```
#fsck -pf /dev/sdb
```

Lakukan mount dan cek apakah filesystem sudah berubah dari ext3 ke ext2.

Cek apakah file dan direktori anda masih ada ?

5. Sekarang kita ubah dari ext2 ke ext4.

```
#tune2fs -O extents,uninit_bg,dir_index,has_journal /dev/sdb
```

Cek apakah filesystem OK. Pastikan filesistem dalam keadaan unmount

```
#fsck -pf /dev/sdb
```

Sekarang cek apakah filesystem sudah berubah dari ext2 ke ext4.

Lakukan mount kembali dan cek tipe filesystem

```
#mount /dev/sdb /mnt/sdb
```

```
#file -sL /dev/sdb
```

Cek apakah file dan direktori anda masih ada ?

Percobaan 5. Membuat Partisi Swap

1. Cek pemakaian memory dan swap. Catat berapa total, used dan free space swap

```
#free
```

2. Sekarang kita membuat partisi swap. Gunakan partisi /dev/sdc2 yang telah anda buat sebelumnya.

Untuk menformat sebagai swap, anda cukup menggunakan perintah. Cek kembali dengan

perintah free, apakah sudah ada perubahan

```
#mkswap /dev/sdc2
```

```
#free
```

3. Aktifkan partisi swap. Cek kembali dengan perintah free, apakah sudah ada perubahan

```
#swapon /dev/sdc2
```

```
#free
```

4. Untuk mengecek summary dari seluruh partisi swap, lakukan

```
#swapon -s
```

Apa berapa partisi swap yang anda miliki ?

5. Sekarang reboot. Cek kembali apakah partisi swap sudah diaktifkan system.

```
#swapon -s
```

Secara otomatis, saat sistem reboot, partisi swap yang kita buat tidak aktif walaupun masih ada.

Ada dapat melakukan swapon /dev/sdc2 untuk mengaktifkannya kembali. Namun jika anda ingin partisi tersebut tetap aktif walaupun system di reboot, maka tambahkan partisi tersebut di /etc/fstab

6. Entry dalam /etc/fstab menggunakan format <filename> <mount-point> <type> <options> <dump> <pass>. Kali ini kita tidak menggunakan /dev/sdc2, namun menggunakan UUIDnya. Cek dan catat UUID /dev/sdc2

```
#blkid /dev/sdc2
```

7. Masukkan UUID di file /etc/fstab. Simpan.

```
UUID=1d46ef32-ac3b-44c3-a8e0-7960a8ce961d swap swap defaults 0 0
```

Cek apakah entry /etc/fstab sudah benar dengan

```
#mount -a
```

8. Reboot dan cek apakah swap sudah aktif

```
#swapon -s  
#free
```

9. Menonaktifkan swapoff

```
#swapoff /dev/sdc2
```

10. Hapus entry tersebut dari /etc/fstab

11. Reboot

3. Reboot kembali linux anda

4. Cek ukuran hard disk anda pada /dev/sda

```
#fdisk -l
```

- 5.

Referensi

<http://community.linuxmint.com/tutorial/view/454>

<http://www.mattpopper.com/2006/05/adding-a-new-disk-to-a-vmware-virtual-machine-in-linux/comment-page-5/>

<http://bocahnewbie212.blogspot.co.id/2014/06/cara-instalasi-dan-konfigurasi-quota-di.html>

<https://www.computernetworkingnotes.com/rhce-study-guide/how-to-create-swap-partition-in-linux.html>