**Laporan Resmi**

**Operating System**

**Gabungan kelompok managemen disk**



Nama: Gagas Amukti Nandaka

Kelas: 1 D4 Teknik Informatika B

NRP: 3120600032

**PROGRAM STUDI TEKNIK INFORMATIKA**

**DEPARTEMEN TEKNIK INFORMATIKA DAN KOMPUTER POLITEKNIK ELEKTRONIKA NEGERI SURABAYA**

**2021**

**Bab 4**

**Dasar Teori**

Pada sistem operasi under UNIX seperti LINUX tidak mengenal adanya drive C:\ atau drive A:\, karena pada LINUX menggunakan sistem hirarki dan penyatuan (direktori di dalam direktori) dan memperlakuan file, direktori dan device driver (termasuk floppy dan cdrom) sebagai file. Sedangkan device driver untuk disk drive ditempatkan pada direktori /dev (direktori tempat semua device). Dan disk drive yang belum dikenal tersebut harus dikenalkan / dimuatkan / dipasangkan pada suatu direktori agar bisa dibaca / digunakan

Manajemen file adalah metode yang digunakan sistem operasi untuk mengatur dan mengorganisir file pada disk atau partisi. Pengertian manajemen file secara umum dapat juga diartikan sebagai kegiatan yang berhubungan dengan penyimpanan, penempatan, pengumpulan, pemeliharaan dan sejenisnya

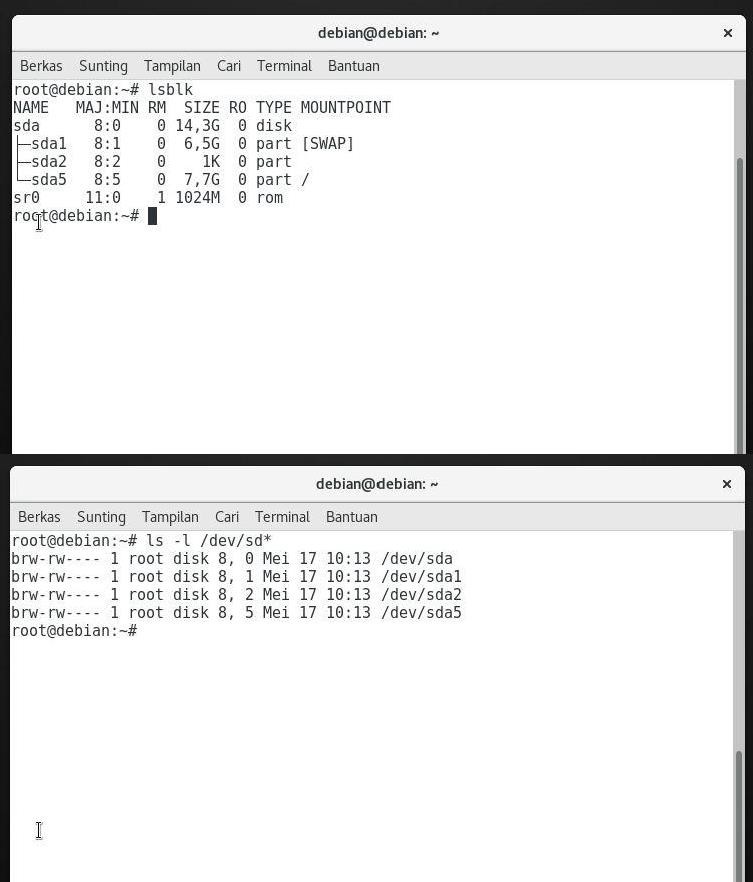
# Chapter 4. disk devices

Ada beberapa command yang tidak bisa mungkin dikarenakan command tersebut digunakan untuk centOS sehingga tidak bisa digunakan di Debian.

Selain itu juga ada juga perintah yang tidak bisa dijalankan karena tidak ada dalam partisi hardisk kita misalnya command **#badblock -ws /dev/sdb** bisa diganti dengan **#badblock -ws**

**/dev/sda** dan juga kita belum menginstall tool tersebut misalnya saja hdparm, bisa diinstall terlebih dahulu dengan perintah **apt-get install hdparm.**

##### Hasil Praktikum



|  |  |  |
| --- | --- | --- |
| **debian@ldebian:** - | **X** |  |
| Berkas Sunting Tampilan Cari Terminal Bantuan  root@debian:-# fdis,k -l I grep Disk  Disk /dev /sda· 14,3 GiB, 15301779456 bytes, 29886288 sectorsDisklabel type: dos  Disk ident ifier: Oxb43dc3aa root@debian:- # I |  |  |
|  |
| **debian@debian:** - | |  |
| Ber as Sunting Tampilan Cari Terminal Bantuarri  root@debian:-# fdisk -l I grep ' Disk /dev/sd' Disk /dev/sda: 14,3 GiB, 15301779456 bytes,  29886288 sectorsroot@debian:-# I | - |  |

**debian@ldebi**

Berkas Sunting Tampilan Cari Terminal Bantuan

roottdebian:-# fdis:k -l I grep Disk

Disk /dev /sda: 14,3 GiB, 15301779456 bytes, 29886288 sectorsDisklabel type: dos

Disk iden

**X**

**debian@ldebia**

Berk;is Sunting Tampil;in C;i,i Terminal B.antu;in

root@deb ian:-·# fdisk - l /dev/sda

**Dislc / de v/ s da : 1 4, 3 G,i B , 15301779456 bytes, 29886288 sectors**

Units: sectors of l \* 512 512 bytes Sector size (logical/physical):512 bytes/ 512 bytesI/0 size

**X**

/dev/sdal

**De vi Boot**

**End Sectors Si ze Id**

\*/dev/sd

13674494 29884415 16209922 7,7G 5

2048 13672447 13670400 6,5G 82 Linux s ap

root@debian:-

13674496 29884415 16209920 7,7G

|  |  |  |
| --- | --- | --- |
|  | **debian@)debian:**~ **X** |  |
|  | Be,kas Sunting Tampilan Cari Terminal Ba ntu an  root@debian: - # dmesg I g rep 'sd[ a-z]' I head  [ 2.407100] sd 0:0:0:0: [sda] 29886288 512-byte logical  blocks: (15.3 GB/14.3GiB)  [ 2. 407158] sd 0: 0: 0:0: [sda] Write Protect is off  [ 2. 4 07162]sd 0:0:0:0: [sda] Mode Sense: 00 3a 00 00  [ 2.407186] sd 0:0:0:0: [sda] Write cache: enabled, read cache: enabled, doesn't support DPO or FUA  [ 2. 482063]sda: sdal sda2 < sdaS *>*  [ 2.483353] sd 0:0:0:0: [sda] Attached SCSI disk  [ 3.624602] EXT4-fs (sda5): mounted filesystem with ordered data mode. Opts:(null)  [ 7.165834] EXT4-fs (sda5): re-mounted. Opts: errors=rem,ount- ro  [ 11.281187] Adding 6835196k swap on /dev /sdal . Priority:-1 extents:l across:6835196k IFS  root@debian:-# |  |
|  |  | |
|  | **debia111@debian:** .. **X** |  |
|  | Berkas Sunting Tampilan Cari Terminal Bantuan  root@debian:-# dmesg I grep - i "scsi disk" [ 2.483353] sd 0:0:0:0: [sda]  Attached SCSI disk root@deb ian:-# |  |

|  |  |  |
| --- | --- | --- |
|  |  | **debian@ldebian:** - |
|  |  | Berkas SLJnt ing Tampifan C.a, i Terminal BantLJan  root@debia:n-# lshw -class volume I grep -Al -B2 scsi description: Linux swap volume  X physical id: 1  bus info: scsi@2:0.0.0,l logical name:  /dev/sdal  description: Extended pa rtitionphysical id: 2  bus info: scsi@2:0.0.0,2 logical name:  /dev/sda2  root@debian:-#I |
|  |  |  |
|  |  | **debian@)debian:** - |
|  |  | Berkas Sunting Tampil,m Cari Terminal Bantuan  root@d ebian:-# lsscsi  [1:0:0:0] cd/dvd VBOX CD- ROM L 0  /dev/sr0 [2:0:0:0] disk ATA VBOX HARDDISK L 0 /dev/sda root@debian:-# |
|  |  | **debian@debian:** - |
|  |  | Berkas Sunting Tamp[lan Car[ Terminal |
|  |  | root@debian:-# lsscsi -c Attafhed devices:  Host: scsil Channel: 00 Target: 00 Lun: 00  Vendor: VBOX Model: CD - ROM Rev: **1 .** 0  Type: CD-ROM ANSI  SCSI revision: 05Host: scsi2 Channel: 00 Target: 00 Lun: 00  Vendor: ATA Model: VBOX HARDDISK Rev: 1.0 Type: Direct-Access ANSI  SCSI revision: 05root@debian:-# |

|  |  |  |
| --- | --- | --- |
|  | **debian@debian:** .. **X** |  |
|  | Berkas Sunting Tampilan Cari Terminal Bant LJan  root@debian;-# aptitude search foremost autopsy sleuthkit I tr -s ' ' p autopsy - graphical interface to SleuthKit  p foremost - forensic program to recover lost files  p sleuthkit - tools for forensics analysis on volume and filesystem data roit@debian:- # I |  |



|  |  |  |
| --- | --- | --- |
|  | **debian@debian:** .. **X** |  |
| Berkas Sunting Tamptlan C, ri Terminal Bant LJan  root@debian:-# badblocks - ws /dev/sda2 Merneriksa dengan pattern Oxaa: done  Membaca dan memband ingkan: done Memeriksa dengan pattern Ox55: done  Membaca dan mernbandingkan: done Memeriksa dengan pattern Oxff: done  Membaca dan memband ingkan: done Merneriksa dengan pattern OxOO: done  Membaca dan mernbandingkan: done rnot@deb ian:-# I |

Ber as

**debian@)debia**

Sunting Tampitan Ca ri Terminal Bantuan

)(

root@debian:-# dd if=/dev /zero of=/dev/sda2 dd: menulis ke '/dev/sda2': Tidak ruang lagi diperangkat3+0 catatan masuk

1r0o2o4t@dbeytbeisan(:1-,0 kB, 1,0 KiB) copied, 0,00258911 s,

**debian@)debia**

Berkas Sunti ng TampHan Cari Terminal

root@debian:-# hdparm /dev/sda2

/dev/sda2:

)(

limultcount

(on)IO suppo rt =

= 128

root@debian:-

geomet

= 1**1**8(603/22-55/63, sectors = 2, start

|  |  |  |
| --- | --- | --- |
| **debiar:i@ldebian:** - | |  |
| Berkas Sunting Tampfl,m Cari Terminal Bantuan |  |  |
| root@deb ian;-# hdparm /dev/sda5  /dev/sdaS:  multcount -  128 (on) IO support - **l** ( 32- bit) reidonly  - 0  (off)  readahead = 256 (on)  geometry - 1860/255/63, sectors= 16209920, start= 13674496root@deb ian:-# I | |
|  | |  |
| **debia111@1debian:** - | |  |
| Berk;is Sunting Tampif;in Cari Terminal Bantuan  root@debian:-# aptitude install sdparrn The following NEW packages will be  installed:sdpa r1m  0 packages upgraded, 1 newly installed, 0 to remove and 327 not upgraded.Need to get 113 kB of archives. After unpacking 324 kB will be used.  Get: l [http ://kambing.](http://kambing/) ui.ac.id/deb ian stretch/main i386 sdparrn i386 1. 08-l+bl [l  13 kB]  Fetcheq 113 kB in ls (78,8 kB/s)  Selectihg prev iously unselected package sdparm.  (Sedang mernbaca basis data .. . 136474 berkas atau direktori telah terpasang.)Preparing to unpack .../sdparm 1.08- l+bl 1386.deb . .. Unpacking sdpar m (1.08- l+bl) .  . -Sedang rnenata sdparrn ( 1.08- l+bl)  Processing triggers for man-db ( 2.7.6.1- 2) .. . root@debian;-# I | |  |

|  |  |  |  |
| --- | --- | --- | --- |
| d ebia ra@d eb ian: - | | **X** |  |
| Berkas Sunting Tampita n Cari Terminal Bantuan  root ebian:-# sdpar m /dev/sda2 I head -1  /dev/sda2: ATA VBOX HARDDISK root@deb ian:-# I | 1 . 0 |  |  |
|  |
| **debian@)debian:** - **X** | | |
| Berkas Sunt ing Tamp[lan Cari Terminal Bantua n  SDPARM(8) SDPARM **SDPARM(8)**  **NAME**  sdparm - access SCSI modes pages; read VPD pages; send simple SCSI com• E mands.  **SYNOPSIS**  **sdparm** [ -- all] [--clear=STR] [- - command=CMD] [ --dbd] [- -defaults]  [-- du mmy l [- -flexible] [--get=STR] [- -help] [ -- hex] [ --inguiryl  [--long] [ -- num-de sc] [--pageaa:PG[,SPG]] [--quiet] [ -- readonly] [- -save]  [ --setca:STR] [ - -six] [- -transport"'TN] [ --v endor:a=VN] [- -verbose] [ --ver• sion] DEVICE [ DEVICE...]  **sdpamr** - - enumerate [ - - all] [- - inguiryl [- - longl [- -page=PG[,SPG]l [ - -transpo1rt=TN] [ --vendo r=VN]  **sdpamr** - -wscan [ - -verbose]  **DESCR IPH ON**  This utility fetches and potent ially changes SCSI device (e.g. disk) mode pages. Inquiry data including Vital Product Data (VPD) pages can also be displayed. Commands associated with starting and stopping the medium; loading and unload ing the medium; and other housekeeping func-  Manual page sdparm(B) line **1** (p ress h for help or q to quit) | | |  |

|  |  |  |
| --- | --- | --- |
| **debian@de bian:** - | |  |
| Berkas Sunting Tampilari Cari Terminal Bantuan |  |  |
| root@debian:-# drnesg I grep - i disk  [ 0.000000]RAMIDISK: [ m,em Ox33a3d00O-Ox34af3fff]  [ 0.439629]VFS: Disk quotas dquot 6.6. 0  [ 2.577890] ata3.00: ATA-6: VBOX HARDDISK, 1.0, max UDMA/133  [ 2.589444]*scsi* 2:0:0:0: Direct- Access ATA VBOX HARD DISK  0 ANSI: 5  2.626094]sd 2:0:0:0: [sda] Attached SCSI disk [ 2.980014]PM: Starting manual resume from disk  root@debian:-# I | |  |

AktIvItas

ra <

Baobab -

Iii

*+* -

**n**

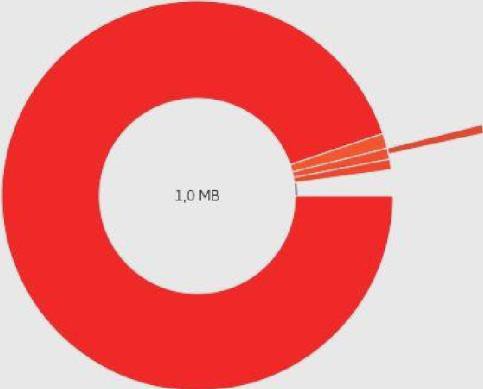
•>

0

Sen 13·48

root )(

- . sy n ap t ic 946,2 kB 4 butir 1 t ahun

* . co n f i g 16,4 kB 6 butir 2 tahun
* D .l o cal 12,3 kB 3 butir 2 tahun
* D . dbus 12,3 kB 3 butir 1 tahun

D .nano 4,1 kB l butir 2 tahun

D .cache 4,1 kB 1 butir 2 tahun

Berka s Sur:1t i ng Tampilan ntuan

**debian@ldeb**

C,ri Te rmi na l Ba

**X**

brw- rw- - - - 1 root disk 8, 0 Mei 17

root@debian:-# ls -1 /dev/sd\*

brw- rw- - - - 1 root disk 8, 5 Mei 17

root@deb

**debian@)debi**

Be,kas Su nt i ng Tamp ilan Ca ri Term ina l Bantua n

root@debian:-# fdisk -1 2>/dev/null I grep [MG

**X**

-

/dev/sdal

Devi

Boot

/dev/sd root@deb

13674494 29884415 16209922 7,7G 5

2048 13672447 13670400 6,5G 82 Linux

End Sectors Size Id

l

|  |  |  |  |
| --- | --- | --- | --- |
|  | **debian@debian:** - | **X** |  |
| Berkas lf"'Sunting Tamp ilan Cari Terminal Bantuan  root@deb ian:-# badblocks -ws  /dev/sda2Merneriksa dengan patte rn Oxaa: done  Membaca dan memband ingkan: done Merneriksa dengan pattern Ox55: done  Mernbaca dan memband ingkan: done Merneriksa dengan pattern Oxff: done  Mernbaca dan membandingkan: done Merneriksa dengan pattern OxOO: done  Membaca dan membandingkan:  doneroot@deb ian:-# I | | |  |
|  | **debian@debian:** - | **X** |
| Berkas Sunting Tampit;m Cari Terminal Bilntuan  root@debian:-# aptitude install lsscsi lshw  lsscsi is already installed at the requested version ( 0.27-3+bl)lshw is already installed at the requested version (02.18-0.1) lsscsi is already installed at the requested version (0.27-3+bl)lshw is already installed at the requested version (02.18- 0.1) No packages will be installed, upgraded, or removed.  0 packages upgraded, 0 newly installed, 0 to remove and 327 not upgraded.Need to get 0 B of archives. After unpacking 0 B will be used.  root@debian:-# I | | |  |

##### Analisa lsblk

**debian@debian:** -

**X**

[1:0:0:0) cd/dvd VBOX

[2:0:0:]0 disk ATA

root@debian:-#

Berkas Sunting Tampilan Cari Terminal Bantuan

root@debian:-# lsscis

CD-ROM

VBOX HARDDISK

1.0 /dev/sr0

1.0 /dev/sda

**r**

lsblk digunakan untuk melihat informasi media penyimpan yang terpasang di linux kamu,seperti flashdisk,hdd,ssd,dan sejenisnya

##### Analisa dmesg

dmesg digunakan untuk mencetak dan mengontrol kernel buffer ring di Linux dan sistem operasi Unix- like lainnya. Perintah ini berguna untuk memeriksa pesan-pesan terkait boot kernel

##### Analisa fdisk

digunakan untuk mengelola prtisi dari create,delete,mengatur jenis pertisi dll

##### Analisa lshw

lshw memberikan informasi yang cukup baik mengenai hardware yang digunakan sistem operasi Linux kita. Informasi yang tersedia berupa konfigurasi memori, versi firmware, konfigurasi main board, versi dan kecepatan CPU, konfigurasi cache, dan kecepatan bus

##### Analisa hdparm

Perintah ini digunakan untuk menampilkan dan mengatur parameter hard disk. hdparm dapat mendeteksi, tampilan dan pengaturan IDE atau SCSI parameter hard disk

##### Analisa sdparm

sdparm dapat digunakan untuk mengakses parameter pada perangkat apa pun yang menggunakan set perintah SCSI. Selain disk SCSI, perangkat tersebut termasuk drive CD / DVD (terlepas dari transportasi), tape drive SCSI dan ATAPI dan penutup SCSI. Sekumpulan kecil perintah yang terkait dengan memulai dan menghentikan media, memuat dan mengeluarkan media yang dapat dilepas dan beberapa fungsi lainnya

##### Analisa lsscsi

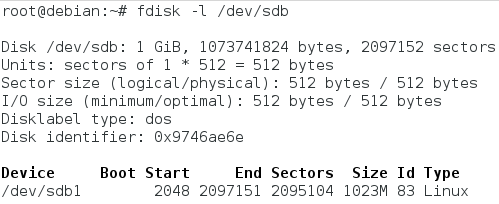
scsi drive mengikuti skema yang sama, tetapi semua mulai dengan / dev / sd. Ketika Anda kehabisan huruf (setelah / dev / sdz), Anda dapat melanjutkan dengan / dev / sdaa dan / dev / sdab dan sebagainya. (Kami akan lihat nanti bahwa volume LVM sering terlihat sebagai / dev/md0, / dev/md1 dll)

**Bab 5**

**Percobaan**

**5.2.1**

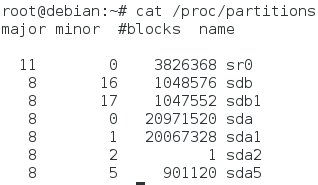
* Perintah: fdisk -l /dev/sdb



Analisa: Unruk melihat partisi apa aja yang ada di /dev/sdb

**5.2.2**

* Perintah: cat /proc/partitions



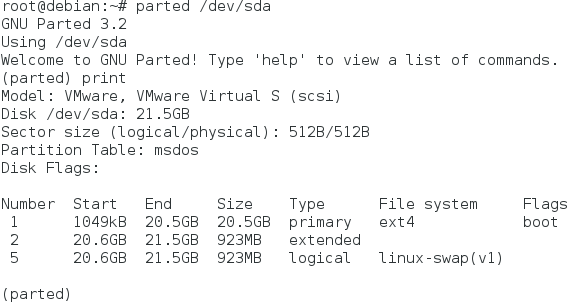
Analisa: File / proc / partisi berisi tabel dengan jumlah partisi mayor dan minor perangkat, jumlah bloknya dan nama perangkat di / dev.

**5.2.3**

* Perintah: rpm -q parted



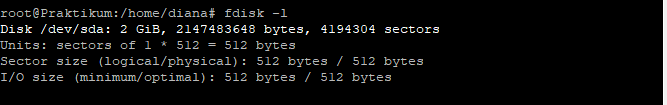
parted /dev/sda



Analisa: Untuk melihat partisi di /dev/sda namum menggunakan perintah parted

**5.3.1**

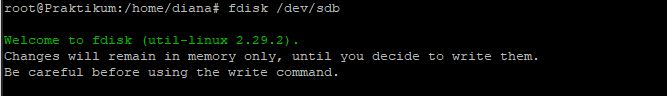
* Perintah: fdisk -l



Analisa: percobaan tersebut berfungsi untuk mengecheck partisi pada disk dan dapat menampilkan partisi serta rincian seperti jenis file system

**5.3.2**

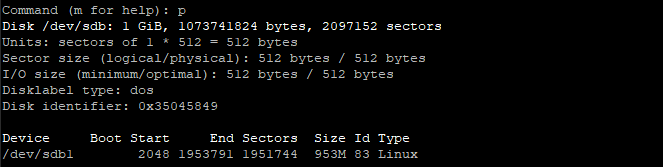
* Perintah: fdisk /dev/sdb



Analisa: percobaan tersebut berfungsi untuk melihat semua perintah yang terdapat pada fdisk

**5.3.3**

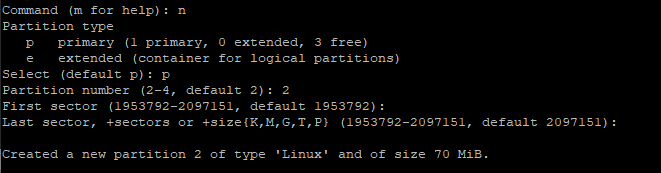
* Perintah: command p

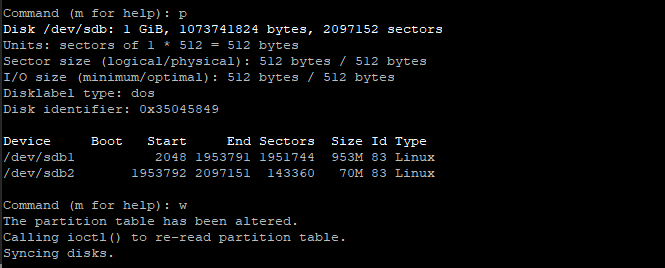


Analisa: opsi P berfungsi untuk mencetak tabel partisi yang ada

**5.3.4**

* Perintah: command n, p dan w

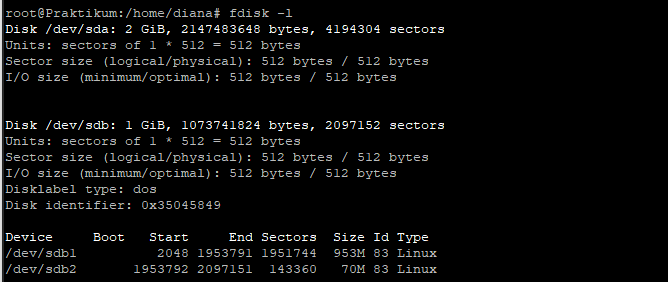




Analisa: opsi N berfungsi untuk membuat partisi baru kemudian opsi p untuk mengecek apakah sudah terdaftar, lalu untuk opsi w berfungsi untuk membuat partisi baru kemudian keluar dari fdisk

**5.3.5**

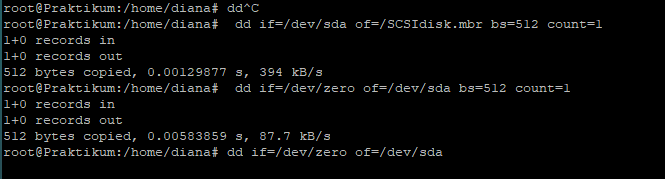
* Perintah: fdisk -l



Analisa: pada percobaan tersebut befungsi untuk mengecek dan memastikan apakah percobaan sebelumnya berhasil dan partisi baru sudah masuk

**5.4.1**

* Perintah: d if=/dev/sda of=/SCSIdisk.mbr bs=512 count=1, dd if=/dev/zero of=/dev/sda bs=512 count=1, dd if=/dev/zero of=/dev/sda



Analisa: pada perintah (dd if=/dev/sda of=/SCSIdisk.mbr bs=512 count=1) berfungsi untuk menyalin 512 MBR dari /dev/sda ke SCSI disk, (dd if=/dev/zero of=/dev/sda bs=512 count=1) berfungsi untuk menghapus semua informasi tentang partisi pada disk dan perintah (dd if=/dev/zero of=/dev/sda) fungsinya sama seperti sebelumnya yaitu dapat menghapus seluruh partisi.

**5.4.2**

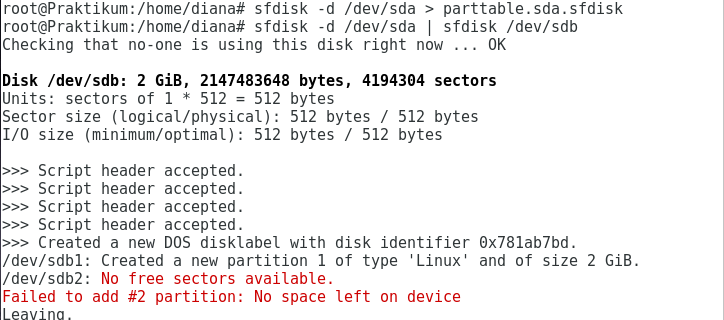
* Perintah: partprobe



Analisa: perintah tersebut berfungsi untuk membaca ulang tabel partisi setelah melakukan re-boot partisi dapat digunakan kembali

**5.4.3**

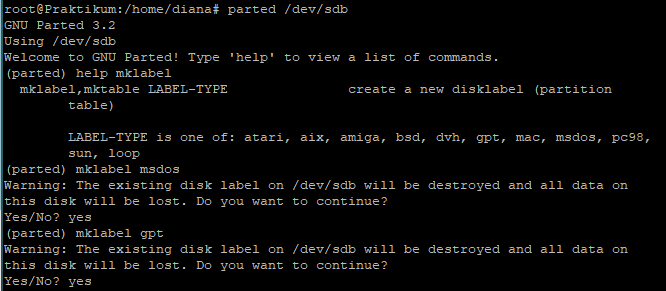
* Perintah: sfdisk -d /dev/sda > parttable.sda.sfdis, sfdisk -d /dev/sda | sfdisk /dev/sdb



Analisa: pada perintah sfdisk -d /dev/sda > parttable.sda.sfdis berfungsi untuk mencadangkan semua informasi partisi dan drive logis ke sebuah file dan perintah sfdisk -d /dev/sda | sfdisk /dev/sdb berfungsi untuk menyalin mbr dan semua info drive logis dari / dev / sda ke / dev / sdb

**5.6**

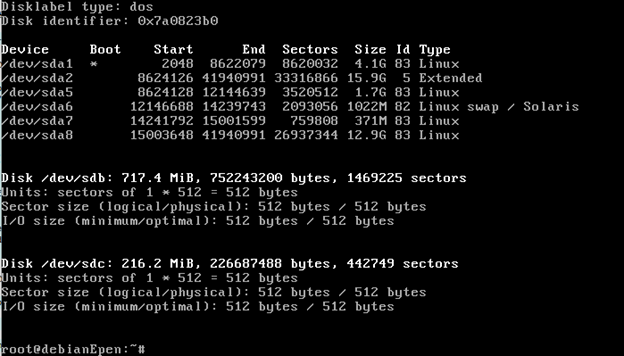
* Perintah: parted /dev/sdb



Analisa: pada perintah parted mirip seperti fungsi fdisk lalu pada perintah (parted /dev/sdb) berfungsi untuk memulai parted dalam mengelola partisi di /dev/sda, untuk opsi help mklabel berfungsi untuk menampilkan semua label yang didukung, opsi mklabel msdos berfungsi untuk memberi label pada hardisk dan opsi mklabel gpt berfungsi untuk mempartisi hard disk non-SSD dengan *parted* karena saya ingin menggunakan tabel partisi GPT.

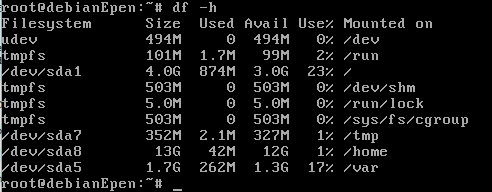
**5.7-5.8**

Perintah fdisk -l

****

Analisa : fdisk -l untuk melihat partisi partis yang ada di dalam linux debian kita beserta ukuran ukuran partisi nya.

Perintah df -h

****

Analisa : Untuk melihat partisi partisi yang sudah digunakan atau di mounted pada direktori direktori yang ada.

Perintah fdisk /dev/sdc dan membuat partisi 200MB

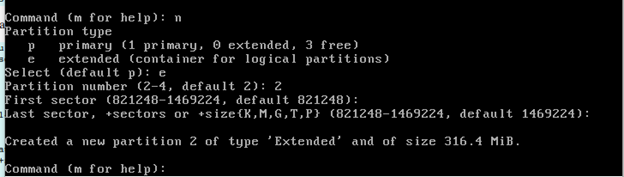
****

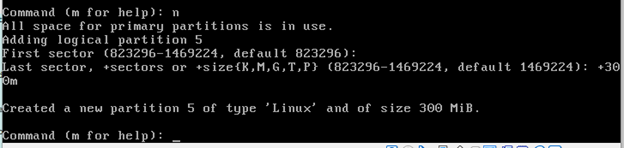
Analisa : disini akan membuat partisi 200MB primary pada partisi sdc untuk membuat partisi tersebut menggunakan perintah fdisk /dev/sdc maka akan tampil seperti diatas lalu masukan huruf n lalu enter untuk membuat partisi baru , masukan huruf p untuk partisi primary dan pilih angka 1 sebagai nomor partisi nya dan untuk first sector dibiarkan default , yang terakhir pada last sector atur ukuran yang akan dipartisi dengan memasukan +200m yang artinya akan membuat partisi 200MB pada partisi sdc yang pertama dan untuk keluar dari fdisk ini menggunakan huruf w lalu enter

Perintah fdisk /dev/sdb dan untuk membuat partisi 400 MB dan 300 MB

****

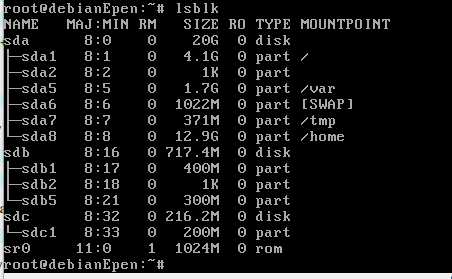
Analisa : disini akan membuat 2 partisi yaitu 400MB sebagai primary disk dan 300 MB sebagai logical. untuk membuat partisi tersebut disini saya menggunakan partisi sdb yang akan dibuat 2 partisi diatas. menggunakan perintah fdisk /dev/sdb untuk masuk pada fdisk partisi sdb nya lalu masukan huruf n untuk membuat partisi baru , masukan p untuk menentukan partisi tersebut sebagai primaty disk dan pilih angka paritisi nya pilih angka 1 dan untuk first sector dikosongkan agar mengikuti default dan untuk last sector isi +400m untuk menambahkan ukuran pada partisi tersebut yaitu 400MB

****

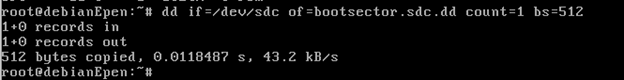
****

Analisa: Diatas untuk membuat partisi kedua nya yaitu 300 MB pada logical disk , langkah tetap sama dengan partisi yang sebelumnya huruf n untuk membuat partisi baru , lalu memasukan e untuk extended atau menjadikan partisi sebagai logical disk dan pilih number partisinya lalu first sector dibiarkan default dan pada last sector masukan ukurannya yaitu 300MB maka dengan perintah +300m dan untuk keluar dari fdisk ini menggunakan huruf lalu enter

Perintah lsblk

****

Analisa : untuk mengecek apakah partisi yang telah kita buat sebelumnya pada sdb dan sdc sudah berhasil dibuat. dan diatas sudah berhasil dibuat sehinnga tampil partisi beserta ukurannya.

****

Analisa : dengan perintah dd disini untuk membuat cadangan pada partisi sdc dari mbr yang berisi partisi yang telah kita buat yaitu partisi primary tadi. cadangan tersebut berada pada bootsector.sdc.dd

****

Analisa : membuat cadangan pada partisi sdb dimana yang berisi 2 partisi yaitu primary dengan ukuran 400MB dan logical nya 300 MB. disini mencadangkan partisi sdb nya kedalam partable.sdb.sfdisk dengan menggunakan perintah sfdisk.

**Kesimpulan**

Pada bab ini kami melakukan banyak percobaan tentang partisi pada perangkat hard disk yaitu bagaimana cara membuat partisi baru menggunakan fungsi command yang sudah tersedia pada sistem, mengecek sebuah partisi dan membuat partisi dengan ukuran tertentu seperti pada percobaan 5.7-5.8

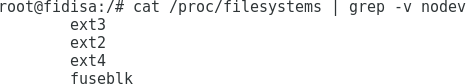
**Bab 6**

##### man fs



Analisa : perintah diatas digunakan untuk menampilkan guide atau dokumentasi yang berhubungan dengan filesystem.

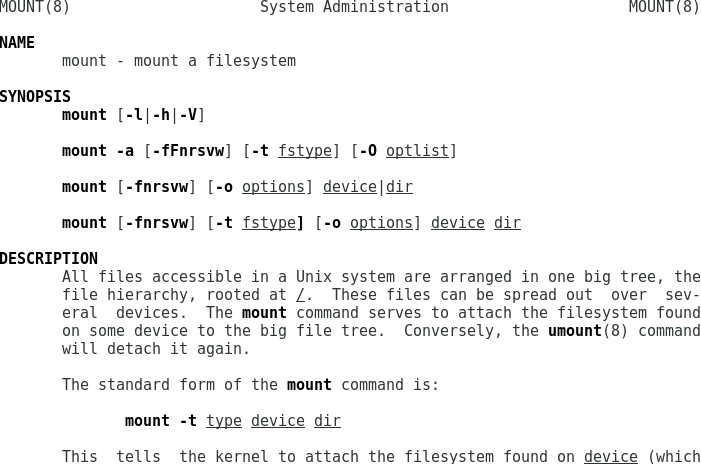
##### /proc/filesystem



Analisa : perintah diatas digunakan untuk menampilkan filesystem yang sedang dimuat pada

/proc/filesystem

##### /etc/filesystems



Analisa : perintah diatas digunakan untuk menampilkan guide atau dokumentasi yang berhubungan dengan mount.

**6.2.1. ext2 and ext3**

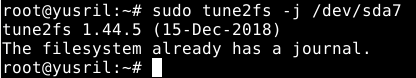
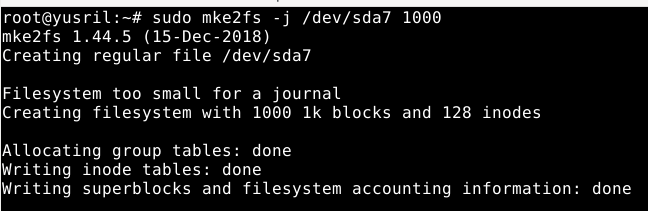
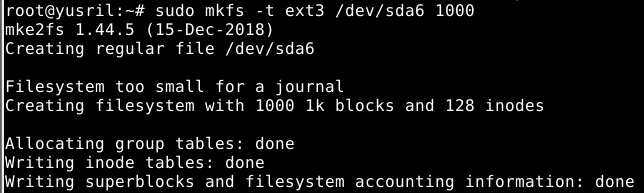
Ext 2 adalah system windows 95 SP 2 pengembangann lebih dari FAT16. NTFS di kenalkan pada Windows NT dan merupakan file system yang berbeda dengan teknologi FATdan ext3 ini adalah upgrade dari ext2 disetiap filesystem terdapat kelemahan dan kelebihan masing masing seperti ext2 mampu menyokong file dari unix sedangkan ext3 tidak mendukung proses pengecekan file system.

Ext2 pertama kali dikembangkan dan diintegrasikan pada kernel Linux, dan sekarang ini sedang dikembangkan juga penggunaannya pada sistem operasi lainnya.

Ext3 merupakan suatu journalled file system, journalled file system didesain untuk membantu melindungi data yang ada di dalamnya. Dengan adanya journalled filesystem, maka kita tidak perlu lagi untuk melakukan pengecekan kekonsistensian data, yang akan memakan waktu sangat lama bagi harddisk yang berkapasitas besar

Setelah sistem file Linux yang paling umum adalah sistem file ext2 (diperpanjang kedua). Sebuah Kerugiannya adalah pemeriksaan sistem file pada ext2 bisa memakan waktu lama. ext2 digantikan oleh ext3 pada kebanyakan mesin Linux. Mereka pada dasarnya sama, kecuali untuk penjurnalan yang hanya ada di ext3. Penjurnalan berarti bahwa perubahan pertama kali ditulis ke jurnal di disk. Jurnal itu memerah secara teratur, menulis perubahan dalam sistem file. Penjurnalan menjaga sistem file dalam keadaan yang konsisten, jadi Anda tidak memerlukan pemeriksaan sistem file setelah pematian yang tidak bersih atau Masalah listrik

* + 1. **Creating ext2 and ext3**



**Analisis :** Dalam membuat ext3 diperlukan akses root dan menggunakan -t ext3 /dev/sda6 dan jumlahsize yang diinginkan agar dapat membuat ext3 dan juga bisa convert ext2 ke ext3 dengan cara tune2fs -j dan alamat fs dan jangan lupa mkinitrd untuk booting device

##### ext4

1. Telah dinyatakan stabil dan didukung sejak kernel linux 2.6.28.
2. Didesign untuk memberikan performance yang lebih baik dan peningkatan kemampuan.
3. Dapat meningkatkan daya tampung maksimal filesystem ke 1 Exa Byte (1,048,576 Tera Byte), dengan ukuran maksimum filesystem dengan 16 TB untuk maksimum file size nya, Fast fsck, Journal checksumming, Defragmentation support.
4. Mengurangi waktu yang diperlukan untuk melakukan pengecekan hardisk (fsck yang mana pada Filesystem Ext3, setiap 2030 kali mount).
5. Berdasarkan test benchmark yang dilakukan oleh beberapa benchmarker, Filesystem Ext4 memiliki keunggulan performance yang significant dalam menulis dan membaca file berukuran besar.
6. Filesystem Ext4 menyisihkan filesystem lain seperti xfs, jfs, Reiserfs dan Ext3. Dalam kasus Ubuntu 9.04, filesystem Ext4 di curigai sebagai faktor utama yang mempercepat waktu boot Ubuntu 9.04. Filesystem Ext4 juga meningkatkan umur hidup media flash seperti SSD. Karena filesystem Ext4 tidak melakukan penulisan data layaknya Filesystem Ext3 yang menulis beberapa kali.

Ext4 ialah Inkarnasi terbaru dari sistem file ext bernama ext4 dan tersedia di Linux kernel sejak 2008. ext4 mendukung file yang lebih besar (hingga 16 terabyte) dan sistem file yang lebih besar dari ext3 (dan banyak lagi fitur lainnya). Pengembangan dimulai dengan membuat ext3 berkemampuan penuh untuk 64- bit. Saat itu ternyata terjadi perubahan signifikan, pengembang memutuskan untuk menamainya ext4. Intinya sebelumnya tidak ada ext4 karena perkembangan linux dengan file yang lebih besar maka dibuatlah ext4

##### xfs

XFS : Sistem file 64-bit performa tinggi yang dibuat oleh Silicon Graphics. Unggul secara paralel operasi I/O dan konsistensi data.

XFS adalah sangat scalable , sistem file kinerja tinggi yang awalnya dirancang di Silicon Graphics , Inc diciptakan untuk mendukung filesystem yang sangat besar ( hingga 16 exabyte ) , file ( 8 exabyte ) dan struktur direktori ( puluhan juta entri ) .

Fitur

utama XFS mendukung journal metadata , yang memfasilitasi pemulihan kecelakaan lebih cepat . The XFS file system juga dapat defragmented dan diperbesar sementara dipasang dan aktif . Selain itu, Red Hat Enterprise Linux 6 mendukung cadangan dan memulihkan utilitas khusus untuk XFS .

Fitur alokasi

XFS fitur skema alokasi berikut :

Alokasi berbasis Luas Kebijakan alokasi Stripe -aware alokasi tertunda

Ruang pra-alokasi

Alokasi tertunda dan optimasi kinerja lainnya mempengaruhi XFS dengan cara yang sama yang mereka lakukan ext4 . Yakni, sebuah program menulis ke sistem file XFS tidak dijamin untuk berada di -disk kecuali isu program yang fsync ( ) panggilan sesudahnya.

Redhat Enterprise Linux 7 akan memiliki XFS sebagai sistem file default. Ini sangat skalabel sistem file berkinerja tinggi. xfs dibuat untuk Irix dan selama beberapa tahun juga digunakan di FreeBSD. Itu

didukung oleh kernel Linux, tetapi jarang digunakan dalam distribusi di luar ranah Redhat / CentOS

##### vfat

VFAT adalah sebuah variasi sistem berkas FAT16 yang mendukung nama berkas panjang, hingga 255 karakter. Sistem berkas ini diintegrasikan ke dalam sistem operasi Windows 95 dan Windows NT 3.51. Meskipun mendukung nama berkas panjang, sebenarnya dalam struktur sistem berkas ini tidak ada perubahan yang signifikan. Bahkan nama berkas panjang akan memakai beberapa entri direktori secara sekaligus

Sistem file vfat ada dalam beberapa bentuk: fat12 untuk floppy disk, fat16 pada ms-dos, dan FAT32 untuk disk yang lebih besar. Implementasi vfat Linux mendukung semua ini, tetapi vfat kekurangan file banyak fitur seperti keamanan dan tautan. fatdisk dapat dibaca oleh setiap sistem operasi, dan banyak digunakan untuk kamera digital, stik usb dan untuk bertukar data antara OS yang berbeda di komputer pengguna rumahan.

##### 6.2.6 ISO 9660

iso 9660 adalah format standar untuk cdroms. Kemungkinan Anda akan menemukan sistem file ini juga di hard disk Anda dalam bentuk gambar cdrom (sering kali dengan ekstensi .iso). Itu standar iso 9660 membatasi nama file ke format 8.3. Dunia Unix tidak menyukai ini, dan karenanya menambahkan ekstensi punggungan rock, yang memungkinkan nama file hingga 255 karakter dan mode file Unixstyle, kepemilikan, dan tautan simbolis. Ekstensi lain untuk iso 9660 adalah joliet, yang menambahkan 64 karakter unicode ke nama file. Standar el torito meluas iso 9660 untuk dapat melakukan booting dari CD- ROM.

##### udf

Kebanyakan media optik saat ini (termasuk cd dan dvd) menggunakan udf, Format Disk Universal

##### swap

Swap adalah sebuah ruang pada harddisk yang dijadikan ruang virtual memory yang digunakan ketika komputer/laptop membutuhkan lebih banyak memory. Dalam artian partisi dengan filesystem Swap ini bekerja sebagai cadangan , apabila RAM yang digunakan penuh Semua hal dipertimbangkan, swap b partisi sebagai partisi swap itu harus diformat dan d

ukanlah sistem file. Tetapi untuk menggunakan ipasang sebagai ruang swap.

**.2.9 gfs**

luster Linux sering kali menggunakan sistem file kl

**.2.10 banyak lagi**

Masih banyak reiserf di linux model lama mungkin b

**.2.11 /proc/filesystems**

##### 6

K uster khusus seperti GFS, GFS2, ClusterFS, ..

##### 6

isa zfs atau os btrfs dan lainnya

##### 6

**Analisis :** Perintah ini menampilkan list yang support dengan file systems. Ketika kita mount sebuah file system tanpa mendefinisikan. Pertama file system akan mendefinisikan ke etc lalu jika tidak bisa bar uke proc sehingga disini terlihat ext 2 tanpa label nodev

##### putting a file system on a partition # Ls –ls /sbin/mk\*

diatas digunakan untuk melihat data dari filesystems

##### tuning a file system

# tune2fs –l /dev/sdal | grep –I “block count”

Analisis :

perintah di atas digunakan untuk menampilka n filesystems pada sbin

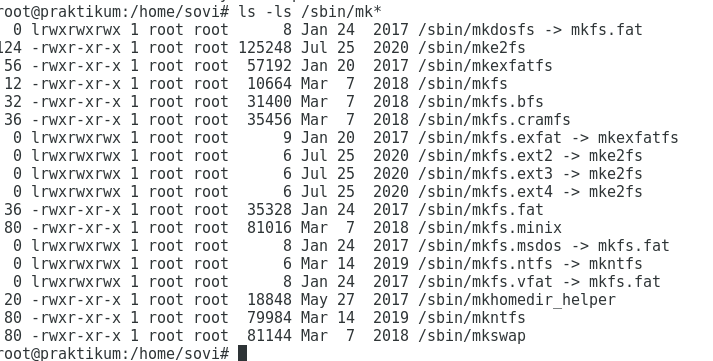
##### # mke2fs

**/dev/sdbl**

Analisis : perintah

# tune2fs -l

/dev/sdal | grep -I “block count”

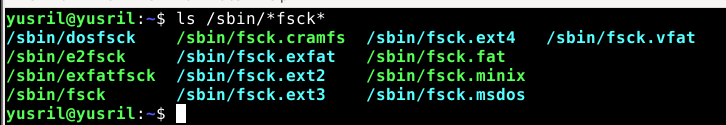


# tune2fs -

m10

atas digunakan untuk me list dan menyetel file system

##### Mengecek file system

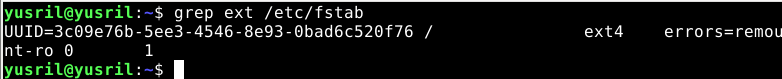


Analisis : fsck berfungsi untuk mengecek file system untuk eror

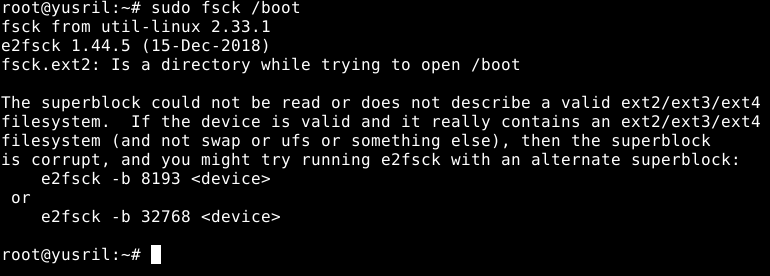
Analisis :

perintah di

kolom pada /etc/fstab untuk determinasi sebuah file system yang harus dicek dan bootup



Mengecek manual file yang telah di mounted sebua file system dan menghasilkan warning dari fsck



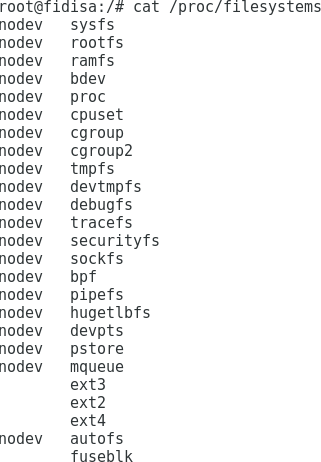
Disini tidak bisa di read karena tidak bisa mendeteksi valid ext2 atau ext3 karena tadi pembuatan ext 2 dan 3 tidak bisa sehingga proses fsck boot tidak bisa dieksekusi karena file ext belum terbuat

##### Practice & 6.7 Solution : file systems

1. List the filesystems that are known by your system.

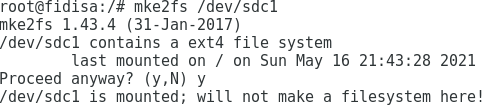


Analisis : perintah diatas digunakan untuk menampilkan guide atau dokumentasi yang berhubungan dengan filesystem.



Analisis : perintah diatas digunakan untuk menampilkan list yang support dengan filesystem. Ketika kita mount sebuah system tanpa mendefinisikan. Pertama fue system akan mendefinisikan ke etc lalu jika tidak bisa baru ke proc sehingga disini terlhat ext2 tanpa label nodev.

1. Create an ext2 filesystem on the 200MB partition



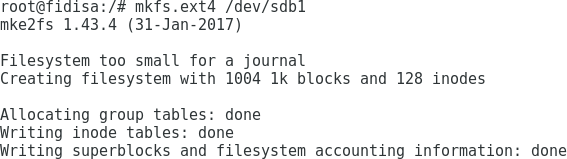
Analisa : perintah diatas digunakan untuk membuat filesystem pada partisi /dev/sdc1. Hal tersebut tidak berhasil dikarenakan partisi /dev/sdc1 sedang termount.

1. Create an ext3 filesystem on one of the 300MB logical drives



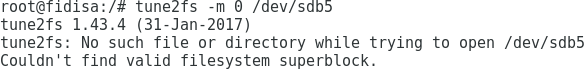
Analisa : perintah diatas digunakan untuk membuat filesystem pada partisi /dev/sdb5. Hal tersebut tidak berhasil dikarenakan partisi /dev/sdb5 tidak ditemukan.

1. Create an ext4 on the 400MB partition.



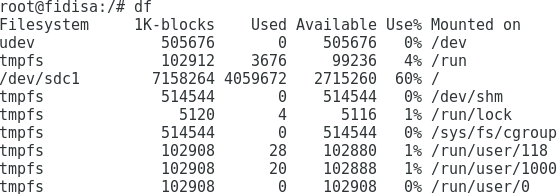
Analisa : perintah diatas digunakan untuk membuat filesystem pada partisi /dev/sdb1.

1. Set the reserved space for root on the ext3 filesystem to 0 percent.



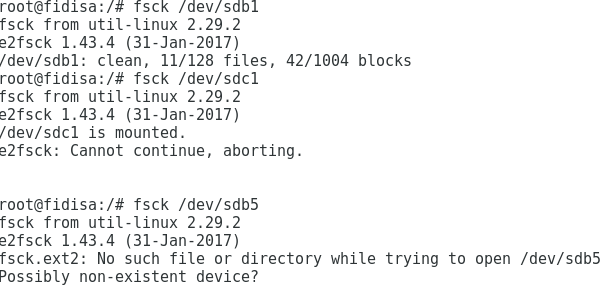
Analisa : perintah diatas digunakan untuk mengset ruang yang dicadangkan untuk root pada filesystem ext3 menjadi 0 persen.

1. Verify your work with **fdisk** and **df**.



Analisa : perintah diatas digunakan untuk menampilkan jumlah ruang disk yang tersedia pada sistem file.

1. Perform a file system check on all the new file systems.



Analisa : perintah diatas digunakan untuk melakukan pemeriksaan pada setiap filesystem baru. untuk

/dev/sdc1 tidak dapat dilakukan pemeriksaan dikarenakan sedang termount. Sedangkan untuk /dev/sdb5 juga tidak dapat dilakukan pemeriksaan dikarenakan partisi tersebut tidak ditemukan.

**Bab 7**

##### Chapter 7. mounting

* 1. **mounting local file systems**
     1. mkdir

This example shows how to create a new **mount point** with **mkdir**.

root@RHELv4u2:~# mkdir /home/project42



Analisa : Perintah di atas digunakan untuk membuat direktori baru yang berada home, bernama project42 menggunakan perintah mkdir

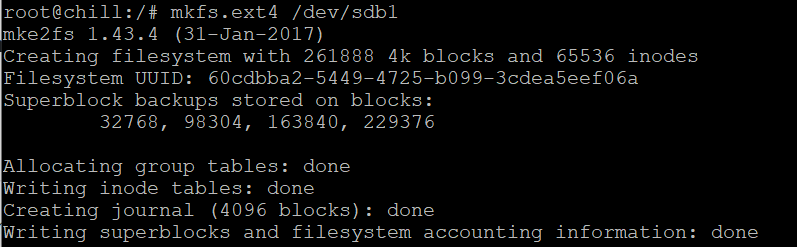
* + 1. mount

When the **mount point** is created, and a **file system** is present on the partition, then **mount**

can **mount** the **file system** on the **mount point directory**.

root@RHELv4u2:~# mount -t ext2 /dev/sdb1 /home/project42/

Once mounted, the new file system is accessible to users.



Analisa : Ketika kita sudah membuat point/file yang akan dilakukan mount dan partisi sudah siap, maka kita bisa melanjutkan mount dengan menambahkan posisi ext nya. Kalau saya sendiri menggunakan ext4.

* + 1. /etc/filesystems

Actually the explicit **-t ext2** option to set the file system is not always necessary. The **mount**

command is able to automatically detect a lot of file systems.

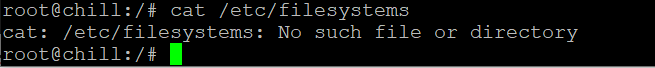
When mounting a file system without specifying explicitly the file system, then **mount** will first probe **/etc/filesystems**. Mount will skip lines with the **nodev** directive.

paul@RHELv4u4:~$ cat /etc/filesystems ext3

ext2

nodev proc nodev devpts iso9660 vfat

hfs



Analisa : Disini terlihat file /etc/filesystems tidak ada. Sebenarnya opsi eksplisit -t ext2 untuk mengatur sistem file tidak selalu diperlukan. Tunggangannya perintah dapat secara otomatis mendeteksi banyak sistem file. Saat memasang sistem file tanpa menentukan secara eksplisit sistem file, maka mount akan melakukannya probe pertama / etc / filesystems. Mount akan melewati baris dengan direktif nodev.

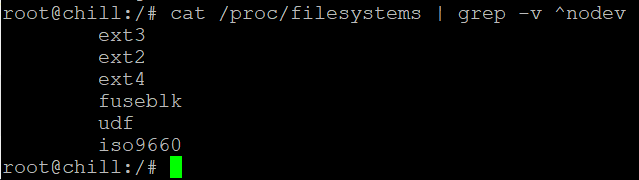
* + 1. /proc/filesystems

When **/etc/filesystems** does not exist, or ends with a single \* on the last line, then **mount**

will read **/proc/filesystems**.

[root@RHEL52 ~]# cat /proc/filesystems | grep -v ^nodev ext2

iso9660 ext3



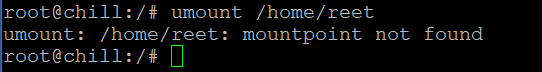
Analisa : Karena mount tidak dapat dilihat pada e/etc/filesystems, maka petunjuk lainnya yakni bisa menggunakan

cat /proc/filesystems

* + 1. umount

You can **unmount** a mounted file system using the **umount** command.

root@pasha:~# umount /home/reet



Analisa : Kita bisa melakukan unmount dengan menggunakan perintah umount.

##### displaying mounted file systems

To display all mounted file systems, issue the **mount** command. Or look at the files **/proc/ mounts** and **/etc/mtab**.

* + 1. mount

The simplest and most common way to view all mounts is by issuing the **mount** command without any arguments.

root@RHELv4u2:~# mount | grep /dev/sdb

/dev/sdb1 on /home/project42 type ext2 (rw)



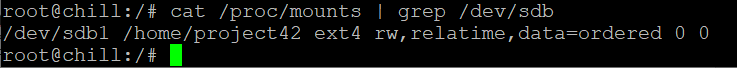
Analisa : Untuk melihat semua mount yang dimiliki pada /dev/sdb adalah dengan mengeluarkan perintah mount tanpa argumen lebih lanjut dan menggunakan grep untuk melihat nya.

* + 1. /proc/mounts

The kernel provides the info in **/proc/mounts** in file form, but **/proc/mounts** does not exist as a file on any hard disk. Looking at **/proc/mounts** is looking at information that comes directly from the kernel.

root@RHELv4u2:~# cat /proc/mounts | grep /dev/sdb

/dev/sdb1 /home/project42 ext2 rw 0 0



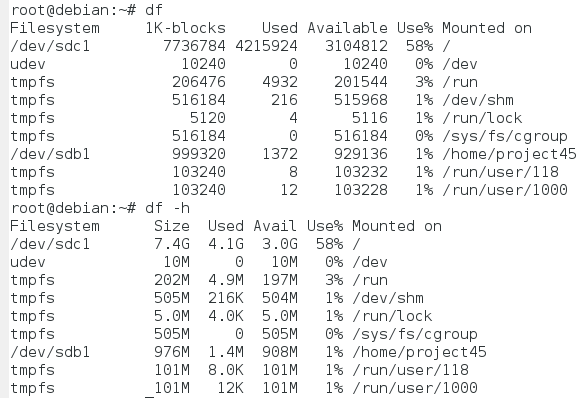
Analisa : Pada perintah ini digunakan hampir sama dengan perintah mount yaitu digunakan untuk melihat mount yang ada pada /dev/sdb namun disini dijelaskan lebih singkat tidak ada variabel seperti on dan type yang dimiliki pada perintah mount.

* + 1. /etc/mtab



Analisa : Pada perintah ini juga hampir sama seperti contoh diatas, namun perbedaannya perintah ini tidak pernah diperbarui lagi oleh kernel. Sehingga jangan sampai mengedit dengan menggunakan perintah ini.

* + 1. df



Analisa : Ada juga cara yang lebih efektif untuk melihat sistem file yang di-mount yaitu perintah df(diskfree). Perintah ini memiliki keunggulan yaitu menunjukkan kepada pengguna ruang kosong disetiap disk yang dimiliki. Selain perintah df terdapat opsi lebih lanjut yaitu perintah df –h, perintah ini digunakan sama seperti df namun lebih bisa dibaca manusia.

* + 1. df –h



Analisa : Untuk perintah ini yaitu digunakan untuk melihat disk yang ada pada sdb1.

* + 1. du



Analisa : Perintah du digunakan untuk meringkas dari keseluruhan penggunaan disk untuk file dan direktori. Opsi –s akan memberikan pengguna untuk melihat total ringkasan untuk direktori utama. Biasanya opsi –s digabung dengan opsi –h yang kesuluruhan digunakan pada titik mount memberikan jumlah total yang digunakan oleh sistem file di partisi itu.

##### from start to finish

Below is a screenshot that show a summary roadmap starting with detection of the hardware (/dev/sdb) up until mounting on **/mnt**.

[root@centos65 ~]# **dmesg | grep '\[sdb\]'**

sd 3:0:0:0: [sdb] 150994944 512-byte logical blocks: (77.3 GB/72.0 GiB) sd 3:0:0:0: [sdb] Write Protect is off

sd 3:0:0:0: [sdb] Mode Sense: 00 3a 00 00

sd 3:0:0:0: [sdb] Write cache: enabled, read cache: enabled, doesn't support \ DPO or FUA

sd 3:0:0:0: [sdb] Attached SCSI disk [root@centos65 ~]# **parted /dev/sdb** (parted) **mklabel msdos**

(parted) **mkpart primary ext4 1 77000**

(parted) **print**

Model: ATA VBOX HARDDISK (scsi)

Disk /dev/sdb: 77.3GB

Sector size (logical/physical): 512B/512B Partition Table: msdos

Number Start End Size Type File system Flags

1 1049kB 77.0GB 77.0GB primary (parted) **quit**

[root@centos65 ~]# **mkfs.ext4 /dev/sdb1**

mke2fs 1.41.12 (17-May-2010)

Filesystem label= OS type: Linux

Block size=4096 (log=2) Fragment size=4096 (log=2)

Stride=0 blocks, Stripe width=0 blocks

4702208 inodes, 18798592 blocks

939929 blocks (5.00%) reserved for the super user First data block=0

Maximum filesystem blocks=4294967296

574 block groups

32768 blocks per group, 32768 fragments per group 8192 inodes per group

( output truncated )

...

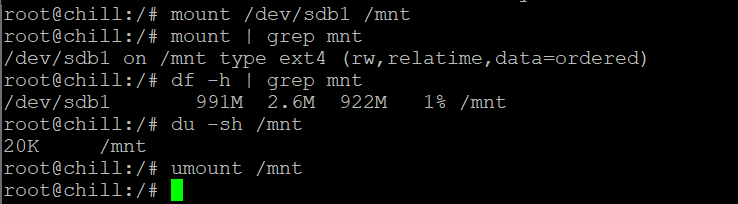
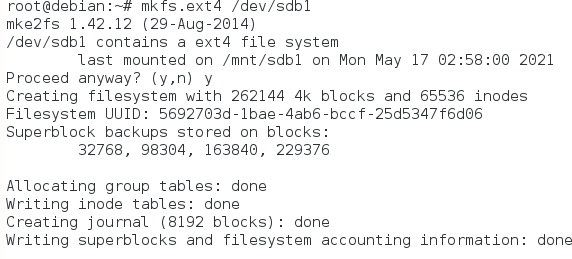
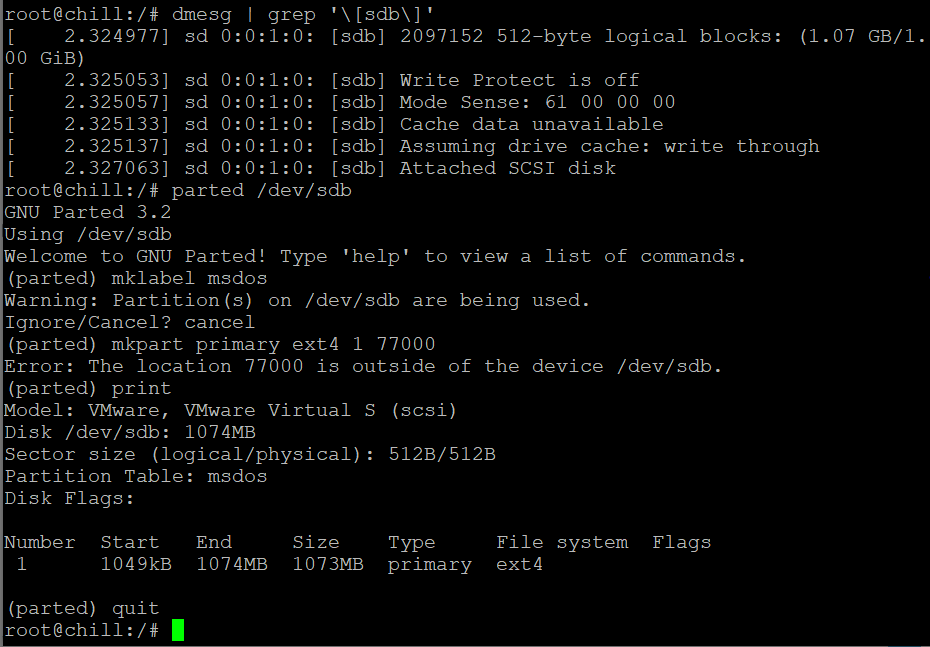
[root@centos65 ~]# **mount /dev/sdb1 /mnt**

[root@centos65 ~]# **mount | grep mnt**

/dev/sdb1 on /mnt type ext4 (rw) [root@centos65 ~]# **df -h | grep mnt**

/dev/sdb1 71G 180M 67G 1% /mnt [root@centos65 ~]# **du -sh /mnt** 20K /mnt

[root@centos65 ~]# **umount /mnt**



Analisa : Pada proses ini menunjukkan jalan ringkasan yang dimulai dengan deteksi perangkat keras yaitu pada

/dev/sdb hingga pemasangan di direktori /mnt.

##### permanent mounts

Until now, we performed all mounts manually. This works nice, until the next reboot. Luckily there is a way to tell your computer to automatically mount certain file systems during boot.

* + 1. /etc/fstab

The file system table located in **/etc/fstab** contains a list of file systems, with an option to automtically mount each of them at boot time.

Below is a sample **/etc/fstab** file.

root@RHELv4u2:~# cat /etc/fstab

/dev/VolGroup00/LogVol00 / ext3 defaults 1 1

LABEL=/boot /boot ext3 defaults 1 2

none /dev/pts devpts gid=5,mode=620 0 0

none /dev/shm tmpfs defaults 0 0

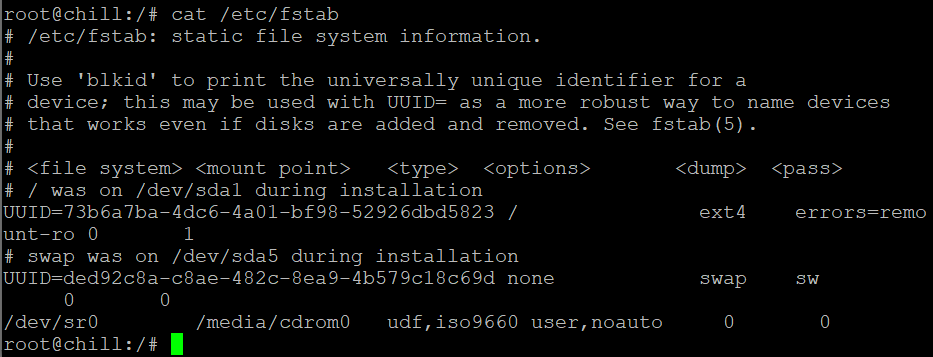
none /proc proc defaults 0 0

none /sys sysfs defaults 0 0

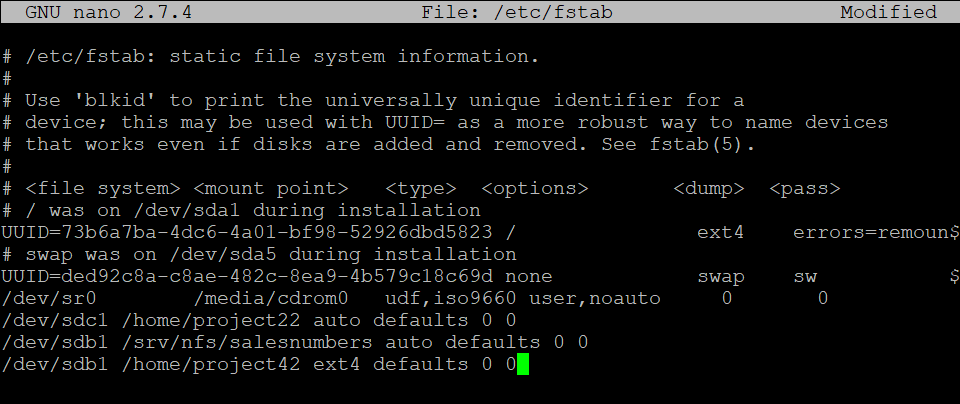
/dev/VolGroup00/LogVol01 swap swap defaults 0 0

By adding the following line, we can automate the mounting of a file system.

/dev/sdb1 /home/project42 ext2 defaults 0 0



nano /etc/fstab



Analisa : Sampai saat ini, kebanyakan orang mengaplikasikan seluruh pemasangan (mounting) secara manual. Ini berfungsi dengan baik, sampai reboot berikutnya. Tetapi untungnya terdapat metode untuk memerintahkan linux agar memasang sistem file tertentu secara otomatis selama boot. Tabel sistem file yang terletak di / etc / fstab berisi daftar sistem file, dengan opsi untuk secara otomatis me-mount masing-masing saat boot.

* + 1. mount /mountpoint

Adding an entry to **/etc/fstab** has the added advantage that you can simplify the **mount** command. The command in the screenshot below forces **mount** to look for the partition info in **/etc/fstab**.

root@rhel65:~# mount /home/project42



Analisa : Menambahkan entri ke / etc / fstab memiliki keuntungan tambahan yaitu kita dapat menyederhanakan proses perintah mount. Perintah pada gambar di atas memaksa mount untuk mencari partisi info di / etc / fstab.

##### securing mounts

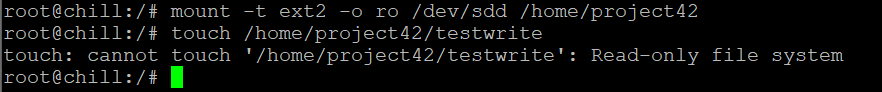
Sistem file dapat diamankan dengan beberapa opsi pemasangan. Berikut ini beberapa contohnya.

* + 1. ro

Opsi ro akan memasang sistem file sebagai hanya baca, mencegah siapa pun untuk menulis.

root@rhel53 ~# mount -t ext2 -o ro /dev/hdb1 /home/project42 root@rhel53 ~# touch /home/project42/testwrite

touch: cannot touch `/home/project42/testwrite': Read-only file system



Analisa : Opsi ini menjelaskan bahwa file perangkat tidak diperbolehkan, seperti blok atau perangkat karakter. Biasanya ini hanya ditemukan di bawah / dev dan tidak terlihat di mount point lainnya. Kebanyakan titik mount akan bekerja dengan benar ketika ini dinonaktifkan, dengan sistem file root sebagai pengecualian.

* + 1. noexec

Opsi noexec akan mencegah eksekusi binari dan skrip pada file yang dipasang sistem.

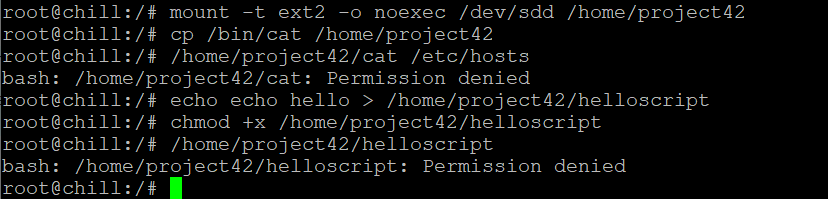
root@rhel53 ~# mount -t ext2 -o noexec /dev/hdb1 /home/project42 root@rhel53 ~# cp /bin/cat /home/project42

root@rhel53 ~# /home/project42/cat /etc/hosts

-bash: /home/project42/cat: Permission denied

root@rhel53 ~# echo echo hello > /home/project42/helloscript root@rhel53 ~# chmod +x /home/project42/helloscript root@rhel53 ~# /home/project42/helloscript

-bash: /home/project42/helloscript: Permission denied



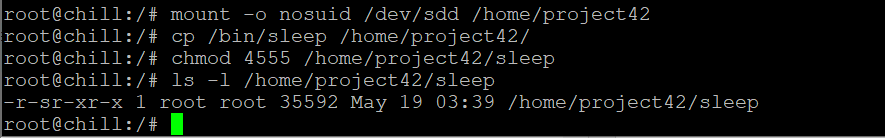
Analisa : Dengan kumpulan opsi ini, biner tidak dapat dieksekusi secara langsung. Berguna untuk: / boot / dev / shm / var dan partisi data. Tidak cocok untuk: root (/), / home (saat menggunakan steam, wine atau development) dan / tmp (misalnya aplikasi kompilasi dapat rusak)

* + 1. nosuid

Opsi nosuid akan mengabaikan binari setuid bit set pada sistem file yang dipasang. Perhatikan bahwa Anda masih dapat menyetel bit setuid pada file.

root@rhel53 ~# mount -o nosuid /dev/hdb1 /home/project42 root@rhel53 ~# cp /bin/sleep /home/project42/ root@rhel53 ~# chmod 4555 /home/project42/sleep root@rhel53 ~# ls -l /home/project42/sleep

-r-sr-xr-x 1 root root 19564 Jun 24 17:57 /home/project42/sleep



Analisa : Jangan gunakan bit set-user-identifier (SETUID) atau set-group- identifier (SETGID) untuk diterapkan. Bit ini disetel dengan chmod (u + s, g +

s) atau unset (u-s, g-s) untuk memungkinkan biner berjalan di bawah pengguna tertentu, yang bukan pengguna aktif itu sendiri. Misalnya, untuk mengizinkan

pengguna biasa menjalankan perintah ping dengan hak akses root. Ini diperlukan untuk memungkinkan membuka soket.

Tetapi pengguna tidak dapat mengeksploitasi fitur setuid.

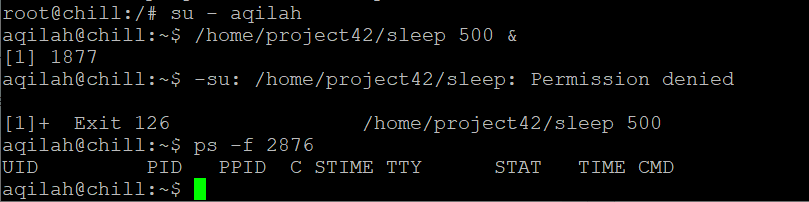
root@rhel53 ~# su - paul

[paul@rhel53 ~]$ /home/project42/sleep 500 & [1] 2876

[paul@rhel53 ~]$ ps -f 2876

UID PID PPID C STIME TTY STAT TIME CMD

paul 2876 2853 0 17:58 pts/0 S 0:00 /home/project42/sleep 500 [paul@rhel53 ~]$



* + 1. noacl

Untuk mencegah perizinan yang berantakan dengan acl, gunakan opsi noacl.

root@rhel53 ~# mount -o noacl /dev/hdb1 /home/project42

Opsi pemasangan lainnya dapat ditemukan di halaman manual pemasangan.



Analisa : Untuk mencegah hak akses cluttering dengan acl maka gunakan perintah noacl. Banyak pilihan untuk menemukan opsi mount dengan menggunakan manual mount.

##### mounting remote file systems

**(Terdapat kendala, karena menggunakan centOS)**

##### practice: mounting file systems

**(Terjawab pada percobaan 7.8)**

1. Mount the small 200MB partition on /home/project22.
2. Mount the big 400MB primary partition on /mnt, the copy some files to it (everything in / etc). Then umount, and mount the file system as read only on /srv/nfs/salesnumbers. Where are the files you copied ?
3. Verify your work with **fdisk**, **df** and **mount**. Also look in **/etc/mtab** and **/proc/mounts**.
4. Make both mounts permanent, test that it works.
5. What happens when you mount a file system on a directory that contains some files ?
6. What happens when you mount two file systems on the same mount point ?
7. (optional) Describe the difference between these commands: find, locate, updatedb, makewhatis, whereis, apropos, which and type.
8. (optional) Perform a file system check on the partition mounted at /srv/nfs/salesnumbers.

##### solution: mounting file systems

1. Mount the small 200MB partition on /home/project22.

mkdir /home/project22

mount /dev/sdc1 /home/project22



Analisa : Perintah di atas digunakan untuk membuat direktori baru yang berada home, bernama project42 menggunakan perintah mkdir. Kemudian melkukan mount pada mount point yang sudah dibuat sebelumnya

1. Mount the big 400MB primary partition on /mnt, the copy some files to it (everything in / etc). Then umount, and mount the file system as read only on /srv/nfs/salesnumbers. Where are the files you copied ?

mount /dev/sdb1 /mnt cp -r /etc /mnt

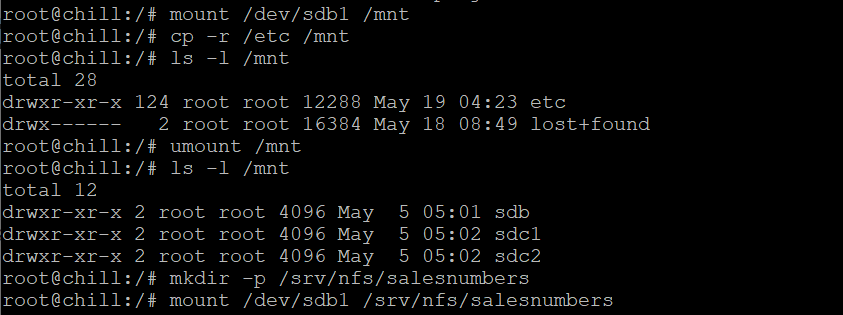
ls -l /mnt umount /mnt ls -l /mnt

mkdir -p /srv/nfs/salesnumbers

mount /dev/sdb1 /srv/nfs/salesnumbers

You see the files in /srv/nfs/salenumbers now...

But physically they are on ext3 on partition /dev/sdb1



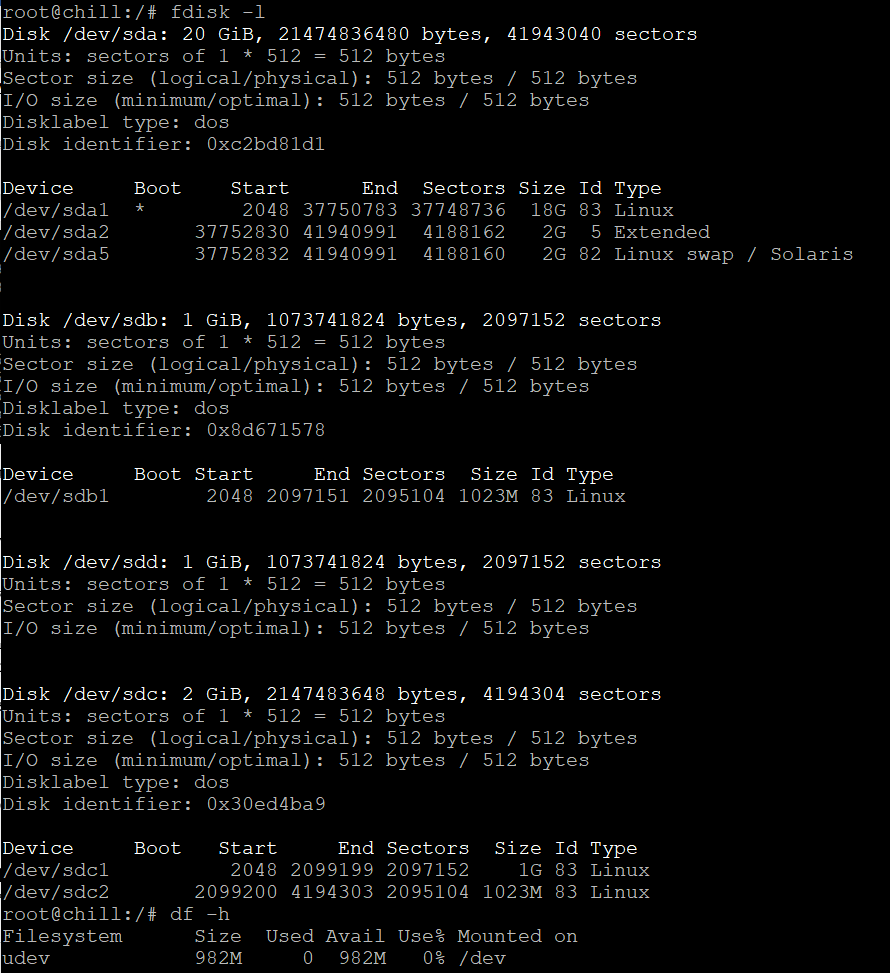
Analisa : Pasang partisi primer 400MB besar di / mnt, salin beberapa file ke situ (semua yang ada di /dll). Kemudian umount, dan mount sistem file sebagai read only di / srv / nfs / salesnumbers.

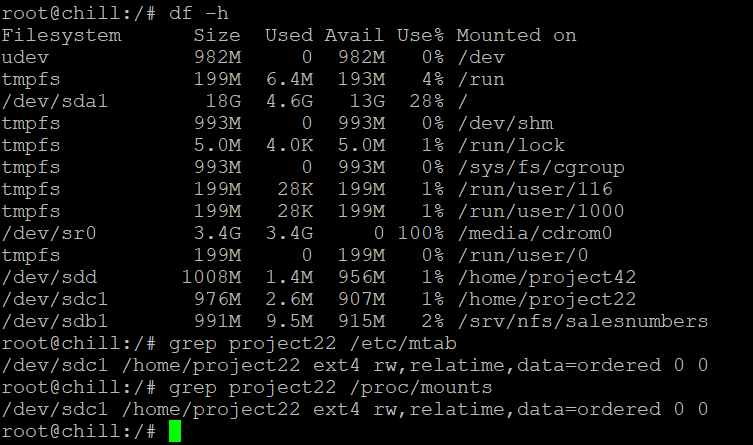
1. Verify your work with **fdisk**, **df** and **mount**. Also look in **/etc/mtab** and **/proc/mounts**.

fdisk -l df -h mount

All three the above commands should show your mounted partitions. grep project22 /etc/mtab

grep project22 /proc/mounts





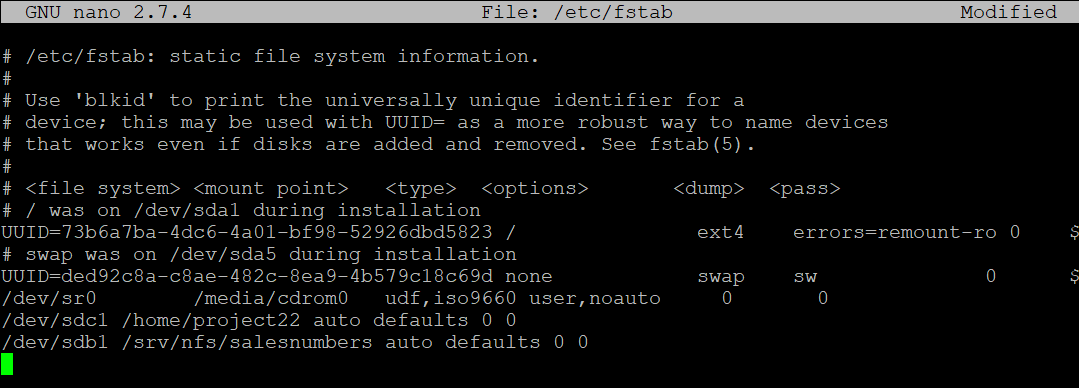
Analisa : Gambar 1 sampai 3 adalah tampilan dari perintah fdisk,df -h dan mount, setelah itu kita grep salesnumbers ke dalam /etc/mtab, Nah perhatikan gambar di atas file sistemnya berada di dalam /etc/mtab.

1. Make both mounts permanent, test that it works.

add the following lines to /etc/fstab

/dev/sdc1 /home/project22 auto defaults 0 0

/dev/sdb1 /srv/nfs/salesnumbers auto defaults 0 0



Analisa : Disini saya menambahkan 2 line baru /dev/sdc1 /home/project22 auto defaults 0 0

/dev/sdb1 /srv/nfs/salesnumbers auto defaults 0 0 pada /etc/fstab dengan menggunakan perintah nano /etc/fstab untuk membuat mounth permanen

1. What happens when you mount a file system on a directory that contains some files ?

The files are hidden until **umount**.

Analisa : Yang terjadi adalah filenya tersembunyi atau tidak dapat di akses dan dapat di akses kembali setelah di umount.

1. What happens when you mount two file systems on the same mount point ?

Only the last mounted fs is visible.

Analisa : Hanya file sistem yang terakhir di pasang yang dapat dilihat.

1. (optional) Describe the difference between these commands: find, locate, updatedb, makewhatis, whereis, apropos, which and type.

man find man locate

...

Analisa : – Whereis = Menampilkan lokasi atau tempat file yang ingin kita gunakan Contoh whereis app (menampilkan tempat dimana aplikasi itu berada)

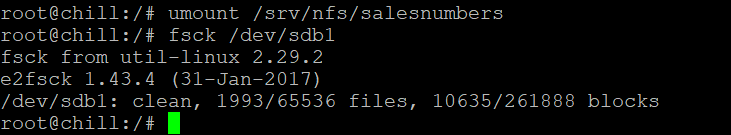
* Locate = Menemukan semua file yang kita inginkan contoh locate file
* Find = contoh find /nama namafile ( mencari semua file yang bernama

namafile)

* Apropos = daftar pengguna halaman untuk subyek.
* Which = Daftar pengguna halaman untuk halaman manual subjectList untuk subjek.

1. (optional) Perform a file system check on the partition mounted at /srv/nfs/salesnumbers.

# umount /srv/nfs/salesnumbers (optional but recommended) # fsck /dev/sdb1



Analisa : Selanjutnya kita melakukan pemeriksaan sistem file pada partisi yang dipasang di / srv / nfs / salesnumbers.

##### Bab 8

**Landasan Teori**

**Troubleshooting Tools**

Sebuah program akan berjalan benar jika *prekondisi* dipenuhi. Contoh: program konversi citra bisa berjalan jika ada input yang valid, ada *disk space* yang cukup untuk output. Beberapa program berasumsi bahwa ini pasti benar, Sehingga jika *disk space* habis, kasus tersebut tidak ditangani, dan hanya muncul *error* generik. Jadi langkah pertama adalah: cek prekondisi dasar sebuah system.

Prekondisi pertama adalah mengecek hal-hal yang bersifat jelas seperti koneksi internet, hardware, dan sebagainya. Prekondisi berikutnya adalah: disk tidak error. Hal ini bisa dicek dengan dmesg, apakah ada pesan *error* dari kernel. Hal itu penting karena jika ada hardware bermasalah maka akan muncul *error* di sini. Jika ada *error*, sebuah file system akan di- *remount* menjadi readonly, artinya log file tidak bisa diandalkan (tidak valid).

Lalu cek *disk space* sistem. Jika *disk space* penuh, maka log mungkin akan gagal ditulis jadi tidak mendapatkan *error message* yang benar. *Disk space* bisa dicek dengan **[df](https://en.wikipedia.org/wiki/Df_(Unix))** [(disk free)](https://en.wikipedia.org/wiki/Df_(Unix)). Jika ternyata memang penuh, kita bisa memakai program [**du** (disk usage)](https://en.wikipedia.org/wiki/Du_(Unix)) untuk mencari tahu direktori / file mana yang memakan space banyak. Masalah disk ini juga kadang bisa dilihat dari “wa” (berapa banyak I/O yang menunggu) di program “[top](https://en.wikipedia.org/wiki/Top_(software))“. Secara umum, pelajarilah berbagai nilai yang muncul di program top, karena ini sangat berguna.

Beberapa program terkait adalah lsof, fuser, chroot, iostat, iotop, dan vmstat.

1. **lsof**

Ketika dipanggil tanpa opsi, lsof akan menampilkan semua file yang terbuka. Melihat perintah (pada kasus init), PID-nya (1) dan pengguna (root) telah membuka direktori root dan /sbin /init. Kolom FD (deskriptor file) menunjukkan bahwa direktori root (rtd) dan direktori yang bekerja saat ini (cwd) dengan perintah /sbin/init. Kolom FD menampilkan **rtd** untuk direktori root, **cwd** untuk direktori saat ini dan txt untuk teks (termasuk data dan kode).

1. **fuser**

perintah fuser akan menampilkan file system dari user. Dalam penggunaannya, dapat menggunakan perintah vi utnuk menjalankan proses pada *background* dan fuser untuk mencari proses yang berjalan.

1. **chroot**

Perintah chroot membuat shell dengan direktori root sebagai alternatif. Hal tersebut secara efektif dapat menyembunyikan apa pun di luar direktori ini. Dalam contoh beberapa kasus dapat diasumsikan bahwa sistem menolak untuk memulai (mungkin karena ada file yang bermasalah dengan /etc/fstab atau pemasangan sistem file root).

1. **Iostat**

iostat melaporkan statistik IO setiap periode waktu tertentu. Hal ini juga termasuk penggunaan total CPU yang kecil. Contoh ini menunjukkan iostat berjalan setiap sepuluh detik dengan /dev/sdc dan /dev/sde menunjukkan banyak aktivitas menulis.

1. **Iotop**

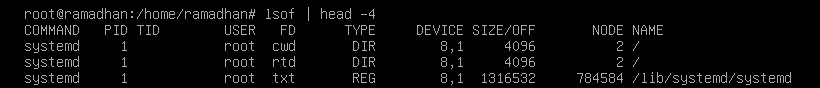
iotop bekerja seperti perintah top tetapi proses perintah dengan input/output bukan dari CPU. Secara default iotop akan menampilkan semua proses. Sebagai contoh penggunaan iotop –o yang berguna hanya untuk menampilkan proses secara actual I/O.

1. **vmstat**

vmstat pada dasarnya adalah alat pemantauan memori, perlu disebutkan di sini untuk pelaporan ringkasan data I / O untuk perangkat blok dan ruang swap.

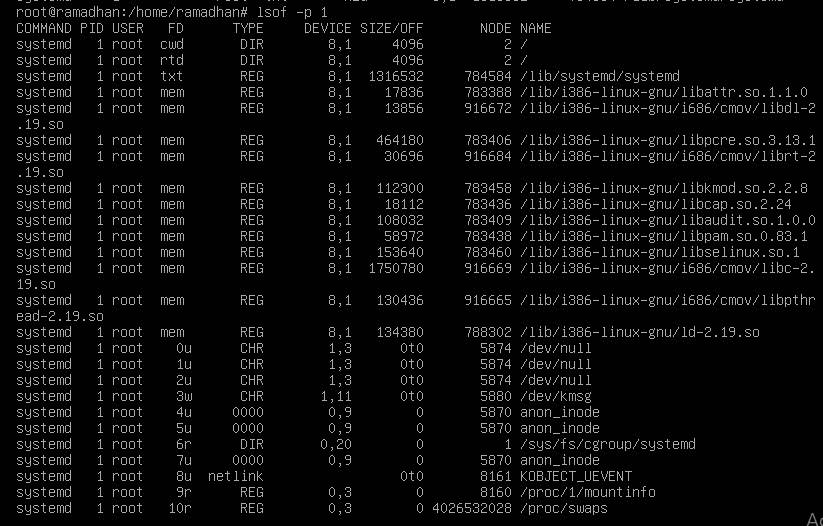
##### Output

* 1. lsof
     1. perintal lsof | head -4



lsof berfungsi me list semua file yang sedang open kemudian ditambah dengan perintah head -4yang membatasi jumlah baris yang ditampilkan hanya 4 baris pada perintah lsof.

* + 1. perintal lsof –p 1



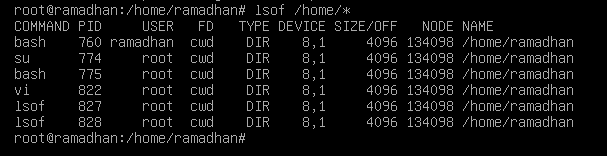
Perintah ini berfungsi melihat file yang terbuka berdasarkan id proses.

* + 1. perintah df –h | grep sda



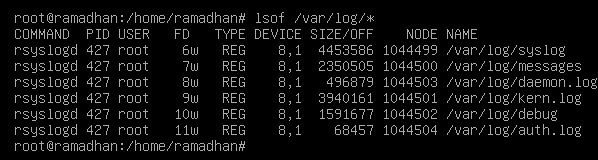
Secara default, perintah df menunjukkan ruang disk dalam 1 kilobyte blocks dan ukuran ruangdisk yang digunakan dan tersedia dalam kilobyte. Untuk melihat informasi format yang mudahuntuk dibaca oleh manusia (megabytes dan gigabytes), berikan opsi –h. Kemudian ditabahperintah untuk mencari kata sda, sehihingga akan muncul filesystem yang ada nama sda nya Bacaan : <https://www.linuxid.net/26115/cara-melihat-penggunaan-hard-disk-dengan-perintah-df/>

* + 1. perintah lsof /<nama dir>/\*



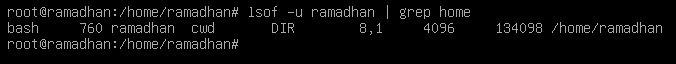
Perintah lsof diatas berfungsi me list semua file yang sedang terbuka pada direktori home.

* + 1. perintah lsof /var/log/\*



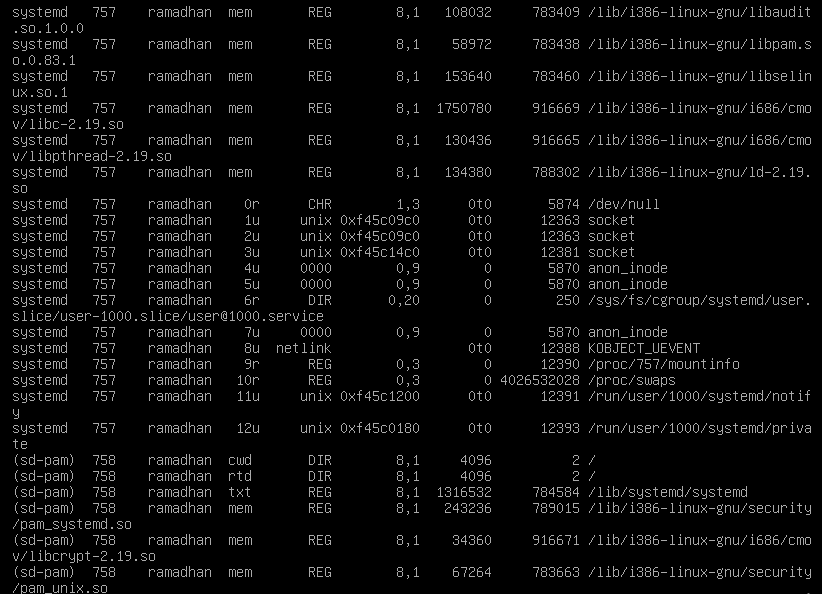
Perintah lsof diatas berfungsi me list semua file yang sedang terbuka pada direktori var bagiansubdirektori log.

* + 1. perintah lsof –u <user> | grep home



Perintah lsof –u diatas berfungsi me list semua file yang sedang terbuka dengan menentukan user tertentu. Pada perintah tersebut juga menggunakan perintah grep home yang berfungsi menjalankan perintah lsof pada direktori home

* + 1. perintah lsof –u^root



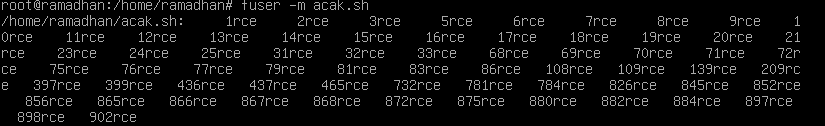
Tambahan perintah -u dari lsof juga mendukung karakter ^ yang artinya 'tidak'. Fungsinya jugasama, yaitu ntuk melihat semua file yang terbuka, tetapi bukan yang dibuka oleh root:

* 1. fuser

1. Perintah jobs

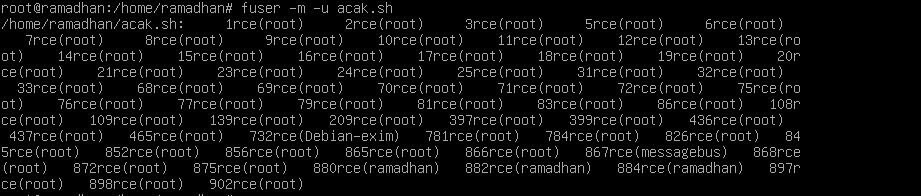
Perintah **jobs** akan menampilkan semua jobs saat ini beserta dengan statusnya.

1. Perintah fuser –m <nama file>



Fuser –m diatas berungsi untuk menemukan id proses dari file systemnya.

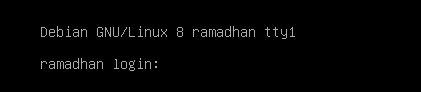
1. Perintah fuser –m –u <nama file>



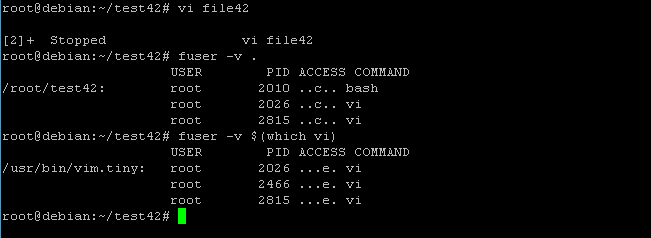
Dengan menambahkan opsi –u pada perintah diatas , berfungsi menampilkan nama pengguna.

1. Perintah fuser –m -k –u <nama file>

Saar menjalankan perita diatas dengan tambahan ospsi –k yang berfungsi untuk menghentikanproses tertentu. Pada saat memasukkan perintah diatas, langsung menuju login



1. Perintah **fuser -v .** & **fuser -v $(which vi)**



Menampilkan semua proses yang menggunakan direktori tersebut (bash & vi)

1. Perintah pd –fp <PID>



Pada contoh ini menunjukkan bagaimana menemukan proses yang mengakses file tertentudengan menggunakan PID nya.

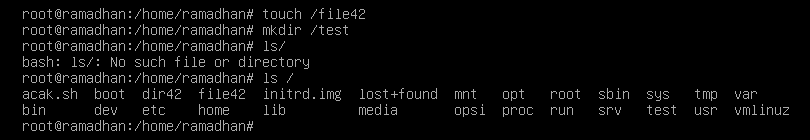
* 1. chroot

1. Perintah df -h | grep root



1. Perintah #touch /<namafile>

#mkdir /<namafile>#ls /



1. Perintah mount /dev/mapper/packer--debian--7-root /mnt



1. Perintah # cd /mnt

# chroot /mnt# ls /

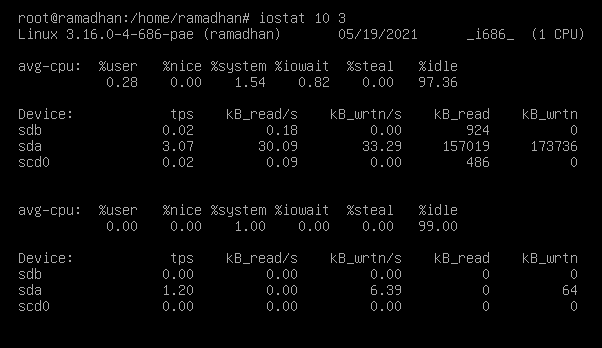


1. Perintah # exit

#ls /



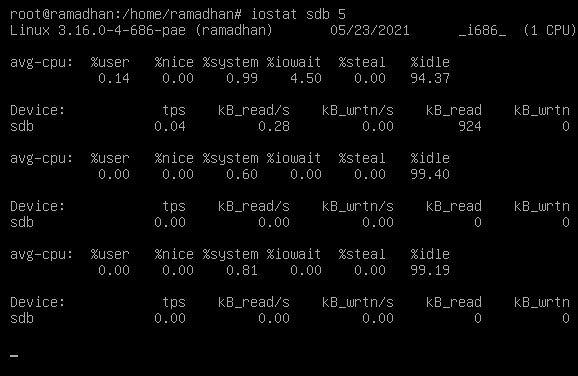
* 1. iostat

1. Perintah iostat 10 3

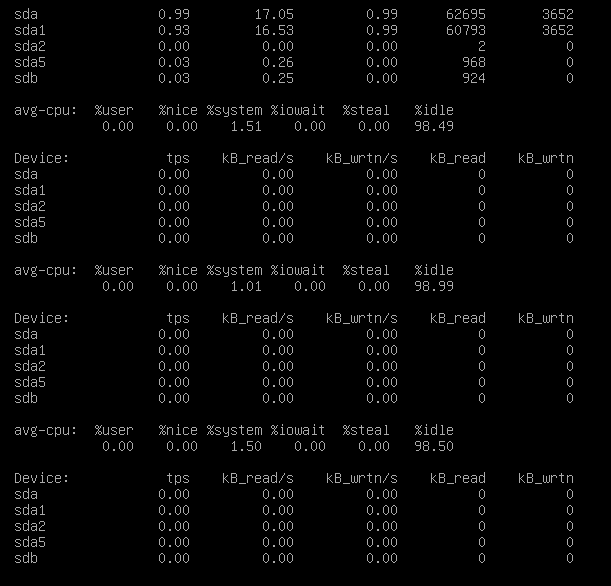
Perintah **iostat digunakan untuk memantau beban perangkat input / output sistem**, melihat berapa lama perangkat aktif dalam kaitannya dengan kecepatan transfer rata-ratanya. Ini juga dapat digunakan untuk membandingkan aktivitas antar disk.Perintah iostat 10 3 berfungsi menampilkan pemantauan sebanyak 3 kali per

5 detik.

1. Perintah iostat sdb 5



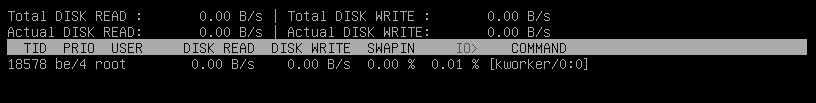
Perintah iostat juga dapa lebih menspesifikasi disk mana yang mau dipantau oleh user. Comtohnya dengan menggunakan perintah iostas sdb 5, dimana memantau disk sdb dan hasilpemantauan akan keluar per 5 detik.

1. Perintah iostat -p sda –p sda 5

Menampilkan statistik dari iostat per partisi

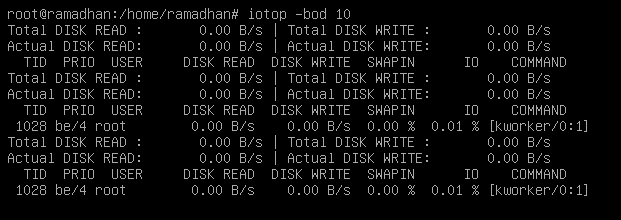
* 1. iotop

1. Perintah iotop –o



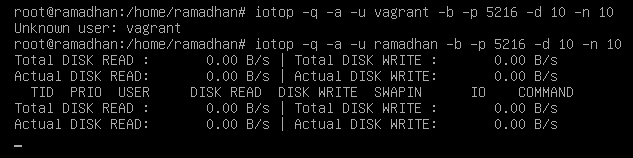
Perintah Iotop berfungsi untuk melihat tampilan semua proses dan jumlah aktivitas disk yang terjadi. Perintah iotop -0 berfungsi untuk melihat/memantau proses atau thread yang sedang melakukan proses input/output

1. Perintah iotop –bod 10



Menggunakaun opsi –b atau –bod pada iostat akan membuat tampilannnya menjadi non interaktif ntuk mengaktifkan penggunaan I / O logging.

1. Perintah iotop –q –a –u <user> –b –p 5216 –d 10 –n 10

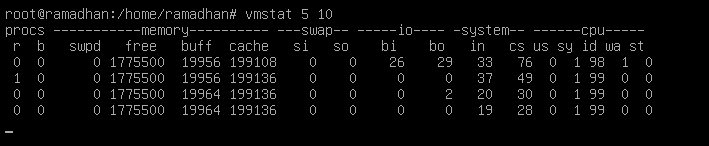


Ini adalah contoh iotop untuk melacak I / O disk setiap sepuluh detik untuk satu user dengan nama ramadhan (dan hanya satu proses pengguna ini, tetapi ini dapat dihilangkan). Opsi -a mengakumulasi I / O dari waktu ke waktu

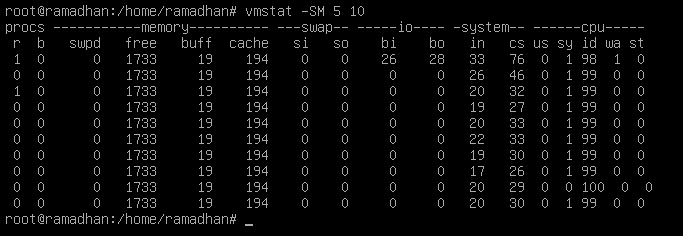
* 1. vmstat

1. Perintah vmstat 5 10

Vmstat berfungsi untuk menampilkan penggunaan memori, swap, memberikan informasi mengenai interrupt sistem, kecepatan I/O, dan statistik CPU secara real time, dan proses yang sedang berjalan. Disitu terlihat pada perintah akan ditampilkan data per 5 detik sebanyak 10 kali



1. Perintah vmstat –SM 5 10



Vmstat dapat digunakan untuk untuk menampilkan memori dalam kilobyte, megabyte atauevenkibibyte dan mebibyte dengan menggunakan -S (diikuti oleh kK m atau M).

##### Kendala

Pada subbab 8.3 tentang perintah chroot, tidak dapat menjalankan percobaan sesuai denganmodul karena tidak dapat menemukan direktori **rootfs.**



# Bab 9

# Landasan Teori UUID

UUID itu singkatan dari Universally Unique Identifier yang dijadikan sebagai standar identifier sama Open Software Foundation (OSF) sebagai bagian dari Distributed Computing Environment (DCE).UUID juga terdapat pada direktori /etc/fstab pada direktori tersebut anda juga mungkin melihat entry seperti UUID=62fa5eac-3df4-448d-a576-916dd5b432f2. Entry ini bukan sebagai penunjuk device name seperti, /dev/hda1, tetapi entry ini disebut identifier unik (UUID) atau lebih jelasnya merupakan kode identifikasi unik yang diberikan oleh sistem. UUID juga dapat digunakan untuk mengidentifikasi DVD drive, Removable media (USB drive) dan seluruh partisi pada hard drive anda. Maksud dari UUID adalah untuk memungkinkan sistem terdistribusi untuk secara unik mengidentifikasi informasi tanpa koordinasi pusat signifikan. Dalam konteks ini kata unik harus diartikan “praktis unik” bukan “unik dijamin”.

Sejak pengidentifikasi memiliki ukuran terbatas adalah mungkin untuk dua item yang berbeda untuk berbagi identifier yang sama. Ukuran pengenal dan proses generasi harus dipilih sehingga membuat ini cukup mustahil dalam praktek. Siapapun dapat membuat UUID dan menggunakannya untuk mengidentifikasi sesuatu dengan keyakinan memadai bahwa identifier yang sama tidak akan pernah tidak sengaja dibuat oleh siapapun untuk mengidentifikasi sesuatu yang lain. Informasi diberi label dengan UUIDs karena itu dapat kemudian digabungkan menjadi satu database tanpa perlu menyelesaikan identifier (ID) konflik. Salah satu penggunaan luas dari standar ini adalah dalam pengidentifikasi Microsoft secara global unik (GUID). Kegunaan penting lainnya termasuk ext2/ext3 UUIDs filesystem, LUKS partisi terenkripsi, GNOME, KDE, dan Mac OS X, yang semuanya menggunakan implementasi yang berasal dari perpustakaan uuid ditemukan dalam paket e2fsprogs

### about unique objects

**Output**

root@debian5:~# vol\_id --uuid /dev/sda1 → blkid /dev/sda1



analisa : vol\_id pada contoh praktikum diganti menggunakan blkid dikarenakan pada versi vol\_id digunakan pada rhell 5 dan pada debian menggunakan blkid. setelah mengetikkan blkid /dev/sda1 maka akan keluar UUID dari sda1.

##### root@rhel53 ~# /lib/udev/vol\_id -u /dev/hda1



**analisa : /lib/udev/vol\_id -u /dev/hda1 pada contoh praktikum diganti menggunakan blkid -s UUID -o value dikarenakan mengingat pada percobaan awal sudah di ganti menggunakan blkid, maka seterusnya juga menggunakan blkid. pada percobaan kali ini gagal dikarenakan tidak adanya partisi hda1**

* 1. ***tune2fs***

[root@RHEL5 ~]# tune2fs -l /dev/sda1 | grep UUID



analisa : Perintah tune2fs berguna untuk memanipulasi parameter sistem file dari sistem file tipe ext2, atau ext3, atau ext4. Ada beberapa opsi yang tersedia dengan perintah tune2fs yang dapat Anda gunakan untuk melihat parameter saat ini, atau mengubah beberapa nilai parameter. pada perintah diatas, setelah dijalankan akan keluar output UUID dari sda1. sebelumnya juga dituliskan grep untuk memfilternya

[root@RHEL5 ~]# /lib/udev/vol\_id -u /dev/sda1 → blkid -s UUID -o value /dev/sda1



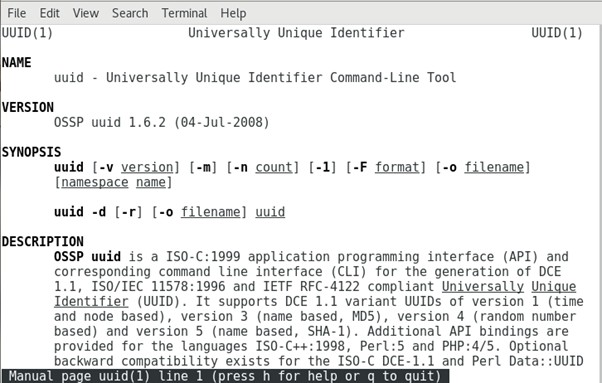
analisa : /lib/udev/vol\_id -u /dev/hda1 pada contoh praktikum diganti menggunakan blkid -s UUID -o value dikarenakan mengingat pada percobaan awal sudah di ganti menggunakan blkid, maka seterusnya juga menggunakan blkid. pada percobaan ini menghasilkan output berupa UUID dari sda1

### uuid

[root@rhel65 ~]# yum install uuid



analisa : yum pada debian diganti menggunakan apt-get install, dikarenakan yum hanya digunakan pada centOS saja,

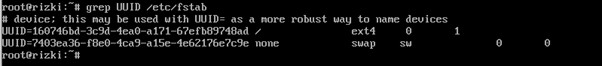
[root@rhel65 ~]# man uuid

analisa : Perintah man di Linux digunakan untuk menampilkan manual pengguna dari setiap perintah yang dapat kita jalankan di terminal. Ini memberikan tampilan rinci dari perintah. setelah menginstall uuid, kemudian menjalankan perintah man, akan terliat rincian dari uuid tersebut

### uuid in /etc/fstab

[root@RHEL5 ~]# tune2fs -l /dev/sdc1 | grep UUID

analisis : Kemudian kami memeriksa apakah itu ditambahkan dengan benar ke / etc / fstab, uuid menggantikan variabel nama perangkat / dev / sdc1. Perintah tune2fs berguna untuk memanipulasi parameter sistem file dari sistem file tipe ext2, atau ext3, atau ext4. Ada beberapa opsi yang tersedia dengan perintah tune2fs yang dapat Anda gunakan untuk melihat parameter saat ini, atau mengubah beberapa nilai parameter. pada perintah diatas, setelah dijalankan akan keluar output UUID dari sdc1. sebelumnya juga dituliskan grep untuk memfilternya [root@RHEL5 ~]# grep UUID /etc/fstab



analisa : perintah diatas digunakan untuk memfilter agar hanya menampilkan UUID pada file

/etc/fstab

[root@RHEL5 ~]# mount /home/pro42 → mount /dev/sdc1

analisa : Perintah mount menginstruksikan sistem operasi bahwa sistem file siap digunakan, dan mengaitkannya dengan titik tertentu dalam keseluruhan hierarki sistem file (titik pemasangannya) dan menetapkan opsi yang berkaitan dengan aksesnya. pada kali ini perintah mount /home/pro42 diganti menggunakan mount /dev/sdc1, dikarenakan pro42 tidak ada di dalam debian.

[root@RHEL5 ~]# df -h | grep sdc1

analisa : df -h untuk melihat informasi pada disk dengan menggunakan format yang dapat dibaca manusia, kemudian | grep sdc1 untuk memfilter, bahwa hanya sdc 1 saja yang akan ditampilkan.

[root@RHEL5 ~]# tune2fs -l /dev/sdb1 | grep UUID



analisa : Perintah tune2fs berguna untuk memanipulasi parameter sistem file dari sistem file tipe ext2, atau ext3, atau ext4. Ada beberapa opsi yang tersedia dengan perintah tune2fs yang dapat Anda gunakan untuk melihat parameter saat ini, atau mengubah beberapa nilai parameter. pada perintah diatas, setelah dijalankan akan keluar output UUID dari sdb1 sebelumnya juga dituliskan grep untuk memfilternya

[root@RHEL5 ~]# df -h | grep sdb

analisa : df -h untuk melihat informasi pada disk dengan menggunakan format yang dapat dibaca manusia, kemudian | grep sdb untuk memfilter, bahwa hanya sdb saja yang akan ditampilkan.

### uuid as a boot device

**root@rhel53 ~# root=/dev/sda1**



analisa : debian tidak bisa melakukan booting dari LVM setelah penginstalan default hanya bisa di RHEL dan CentOS jadi Garis yang dimulai dengan root= adalah kelanjutan dari baris kernel yang akan menghasilkan pergantian /dev/sda1 digantikan UUID tidak bisa dijankan

### - 9.7 practice and solution: uuid and filesystems

##### root@ubu1004:~# blkid /dev/sda2

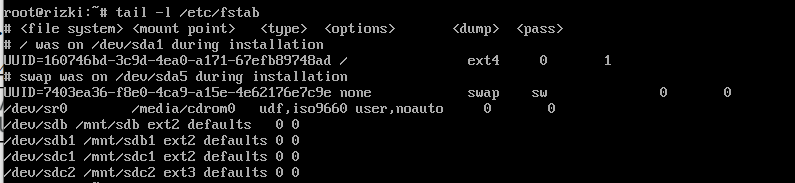


**root@ubu1004:~# tune2fs -l /dev/sda2 | grep UUID**



analisa : setelah melakukan blkid, akan tampil UUID besrta typenya, jika tune2fs dikarenakan sudah difilter, maka yang muncul hanyalah UUID nya, tanpa diikuti typenya

root@ubu1004:~# tail -1 /etc/fstab



analisa : Perintah tail menampilkan bagian terakhir (10 baris secara default) dari satu atau lebih file atau data yang disalurkan. Ini juga dapat digunakan untuk memantau perubahan file secara real time. pada percobaan kali ini, perintah tail menampilkan isi dari /etc/fstab

paul@deb503:~$ grep ^[^#] /boot/grub/menu.lst | grep root=

analisa : tidak bisa dijalankan dikarenakan menu.lst tidak ada pada direktori /boot/grub. saat dibuat secara manual pun tidak bisa, sehingga file tidak bisa di coba pada percobaan kali ini.

**Bab 10**

**10.1 Perangkat Keras atau Perangkat Lunak**

Redundant Array of Independent atau RAID dapat diatur menggunakan perangkat keras atau perangkat lunak. RAID perangkat keras lebih mahal, tetapi menawarkan kinerja yang lebih baik. Software RAID lebih murah dan mudah dikelola, tetapi menggunakan CPU dan memori Beban kerja pada cpu untuk perangkat lunak RAID dulu tinggi, tetapi cpu sudah banyak danlebih cepat.

**10.2 RAID Level**

**10.2.1. RAID 0**

raid 0 menggunakan dua atau lebih disk, dan sering disebut striping (atau stripe set, atau striped volume). Data dibagi dalam beberapa bagian, potongan tersebut tersebar merata di setiap disk dalam array. Keuntungan utama raid 0 adalah Anda dapat membuat drive yang lebih besar. raid 0 adalah satu-satunya raid tanpa redundansi.

**10.2.2. JBOD**

jbod menggunakan dua atau lebih disk, dan sering disebut penggabungan (spanning, spanned set, atau volume rentang). Data ditulis ke disk pertama, hingga penuh. Kemudian data ditulis ke disk kedua .Keuntungan utama dari jbod (Just a Bunch of Disks) adalah Anda dapat membuatnya drive yang lebih besar. JBOD tidak menawarkan redundansi. Jbod menggunakan dua atau lebih disk, dan sering disebut penggabungan (spanning, spanned set, atau volume rentang). Data ditulis ke disk pertama, hingga penuh. Kemudian data ditulis ke

disk kedua .Keuntungan utama dari jbod (Just a Bunch of Disks) adalah Anda dapat membuatnya drive yang lebih besar. JBOD tidak menawarkan redundansi.

**10.2.3. RAID 1**

raid 1 menggunakan tepat dua disk, dan sering disebut mirroring (atau mirror set, atau mirror

volume). Semua data yang ditulis ke array ditulis di setiap disk. Keuntungan utama dari raid 1 adalah redundansi. Kerugian utamanya adalah Anda kehilangan setidaknya setengah dari disk yang tersedia ruang (dengan kata lain, Anda setidaknya menggandakan biaya).

**10.2.4. RAID 2, 3 dan 4**

raid 2 menggunakan bit level striping, raid 3 byte level, dan raid 4 sama dengan raid 5, tetapi dengan a disk paritas khusus. Ini sebenarnya lebih lambat dari raid 5, karena setiap menulis akan melakukannya untuk menulis paritas ke disk yang satu (bottleneck) ini. Kecil kemungkinan Anda akan melihat serangan ini tingkat produksi.

**10.2.5. RAID 5**

raid 5 menggunakan tiga atau lebih disk, masing-masing dibagi menjadi beberapa bagian. Setiap potongan waktu ditulis ke array, salah satu disk akan menerima potongan paritas. Tidak seperti raid 4, parity chunk akan bergantian di antara semua disk. Keuntungan utama dari ini adalah bahwa raid 5 akan memungkinkan untuk penuh pemulihan data jika terjadi satu kegagalan hard disk.

**10.2.6. RAID 6**

raid 6 sangat mirip dengan raid 5, tetapi menggunakan dua potongan paritas. raid 6 melindungi dari dua hard kegagalan disk. Oracle Solaris zfs menyebut raidz2 ini (dan juga memiliki raidz3 dengan tiga paritas).

**10.2.7. RAID 0 + 1**

serangan 0 + 1 adalah cermin (1) dari garis-garis (0). Ini berarti Anda pertama kali membuat dua set serangan 0 stripe, dan lalu Anda mengaturnya sebagai satu set cermin. Misalnya, jika Anda memiliki enam disk 100GB, maka set garis masing-masing 300GB. Dikombinasikan dalam sebuah cermin, ini menghasilkan total 300GB. serangan 0 + 1 akan bertahan satu kegagalan disk. Ini hanya akan bertahan dari kegagalan disk kedua jika disk ini ada di file

set garis yang sama dengan disk yang gagal sebelumnya.

**10.2.8. RAID 1 + 0**

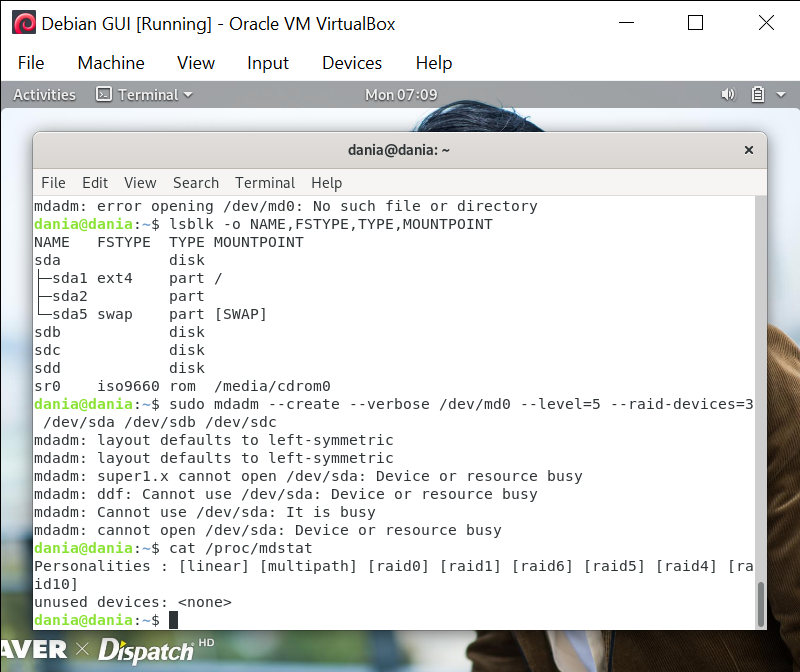
serangan 1 + 0 adalah garis (0) dari cermin (1). Misalnya, jika Anda memiliki enam disk 100GB, maka Anda terlebih dahulu membuat tiga mirror masing-masing 100GB. Anda kemudian menyatukannya menjadi 300GB mendorong. Dalam contoh ini, selama tidak semua disk di mirror yang sama gagal, disk dapat bertahan hingga tiga kegagalan hard disk.

**10.2.9. RAID 50**

raid 5 + 0 adalah stripe (0) dari raid 5 array. Misalkan Anda memiliki sembilan disk 100GB, maka Anda dapat membuat tiga serangan 5 array masing-masing 200GB. Anda kemudian dapat menggabungkannya menjadi satu yang besar set garis.

**10.3 Building a Software raid5 array**

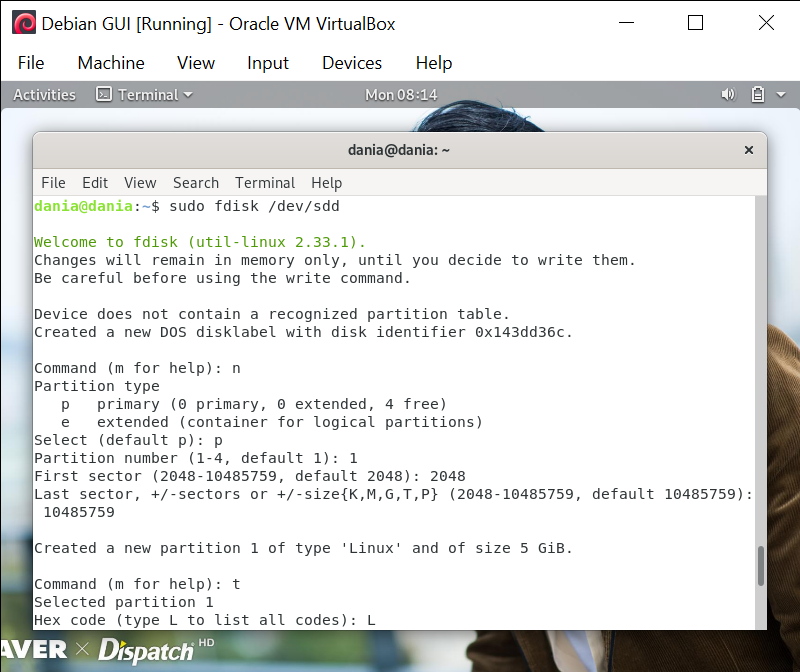
**10.3.1 do we have three disks**

****

as you can see, we have three disks without file system. These will be the raw components we will use to build the array.

**10.3.2 fd partition type**

The next step is to create a partition of type **fd** on every disk.

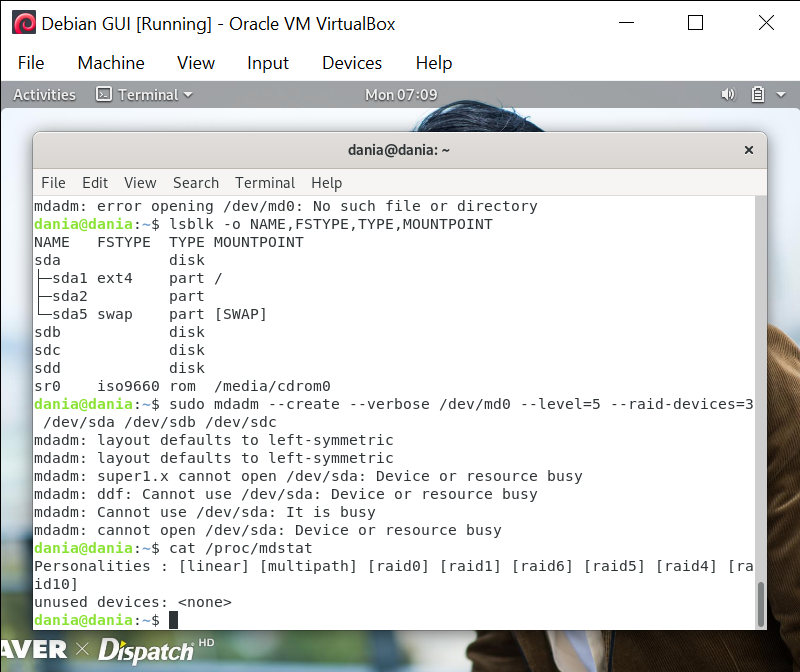
****

**10.3.3 verify all the partitions**

Now all three disks are ready for **raid 5**, so we have to tell the system what to do with these disks.



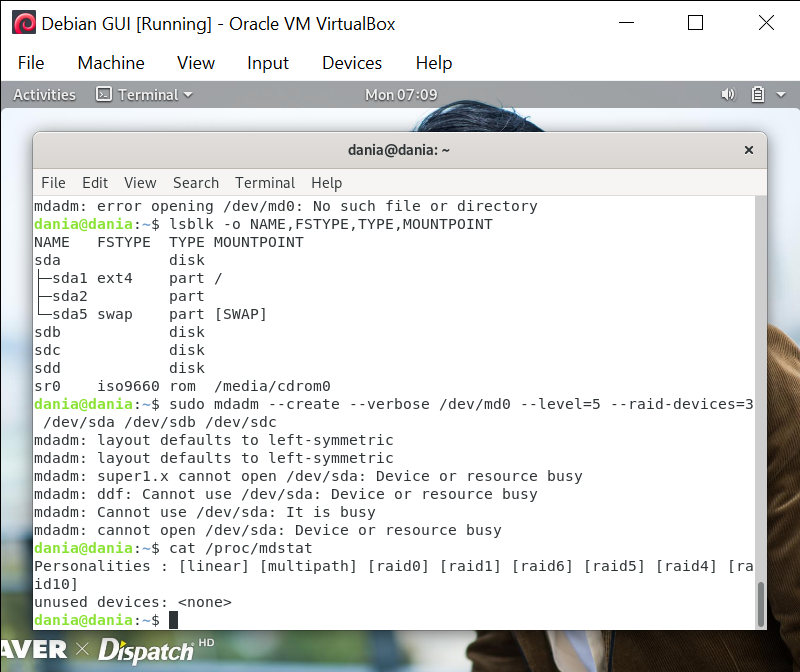
**10.3.4 create the raid 5**

****

the mdadm tool will start to configure the array. This can take some time to complete, but array can be used this time.

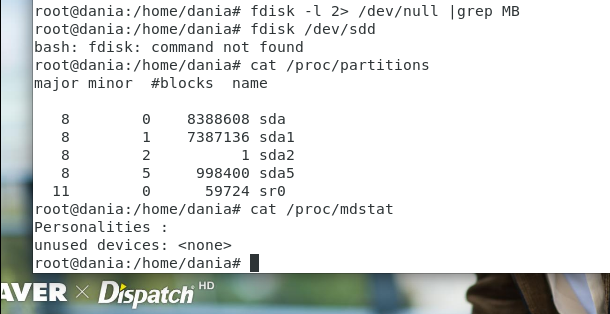
**10.3.5 proc/mdstat**

monitoring the progress of the mirroring

****

**10.3.6 Mdadm Detail**

use mdadm --detail to get information

****

**Bab 12**

**ISCSI DEVICES**

iSCSI (Internet Small Computer System Interface) adalah protokol yang memungkinkan SCSI melalui IP. Ini berarti Anda dapat memiliki SCSI lokal perangkat (seperti / dev / sdb) tanpa memiliki perangkat keras penyimpanan di komputer lokal.

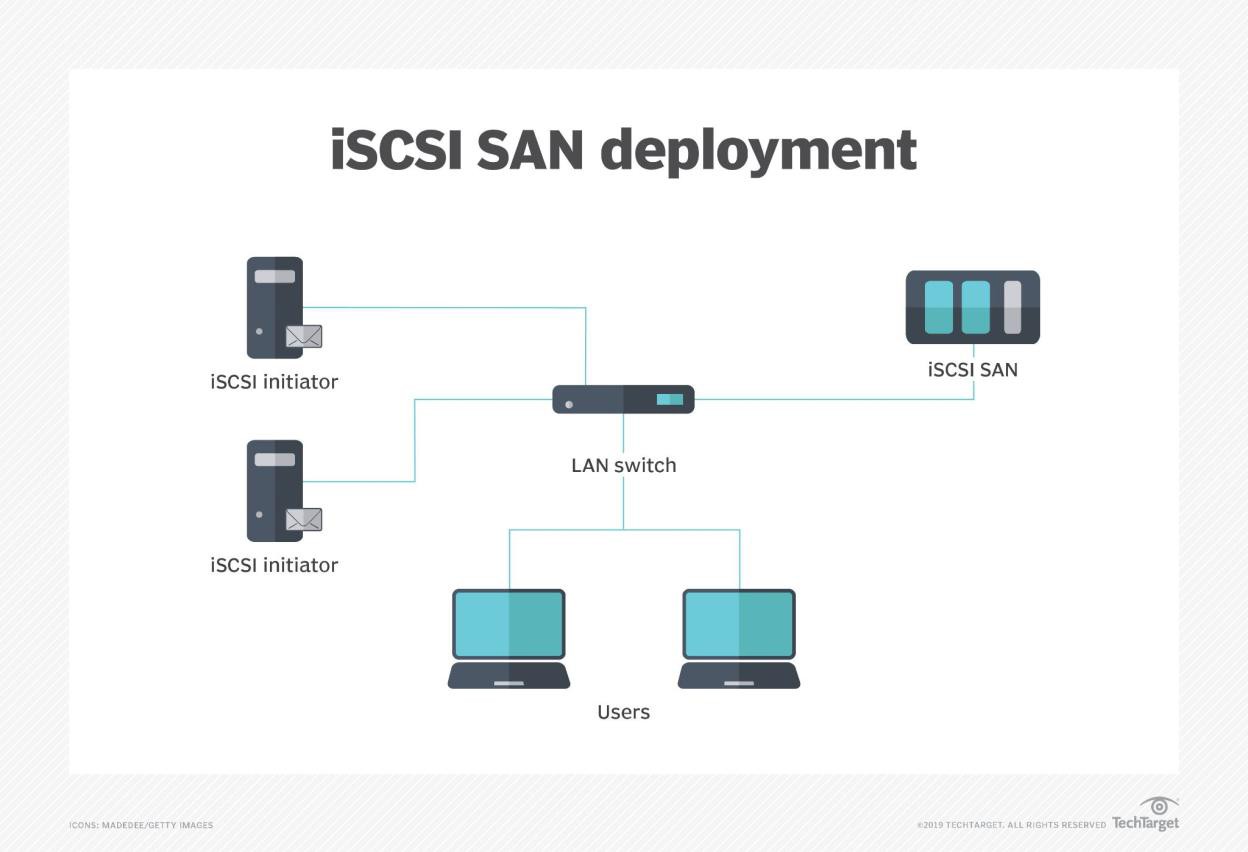
Selain itu ISCSI berperan untuk mengangkut data pada block level antara iscsi inisiator pada mesin klient dengan iscsi target pada mesin penyimpanan pada server.

Komputer yang memegang perangkat keras penyimpanan fisik disebut Target iSCSI. Setiap perangkat iSCSI yang dapat dialamatkan individu pada server target akan mendapatkan nomor LAN.

Komputer klien iSCSI yang terhubung ke server Target disebut Inisiator. Sebuah inisiator akan mengirimkan perintah SCSI melalui IP, bukan langsung ke perangkat keras. Inisiator akan terhubung ke Target.

#### **ISCSI Target** : Target iSCSI adalah perangkat server / penyimpanan yang menerima perintah iSCSI

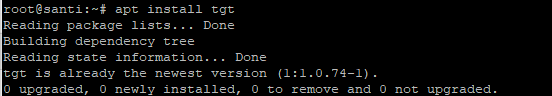
**Initiator** : Klien atau host iSCSI (juga dikenal sebagai inisiator iSCSI) adalah driver perangkat lunak yang berjalan di komputer yang terhubung ke jaringan Ethernet. Inisiator iSCSI menyediakan konektivitas ke perangkat target iSCSI melalui koneksi Ethernet. Ini memungkinkan untuk mengakses penyimpanan yang berada di lokasi berbeda dari komputer Anda. Saat komputer Anda tersambung ke disk pada perangkat target iSCSI, disk tersebut muncul sebagai hard disk lokal yang dapat dikelola dan diakses seperti hard disk lokal lainnya.



**iSCSI target on Debian**

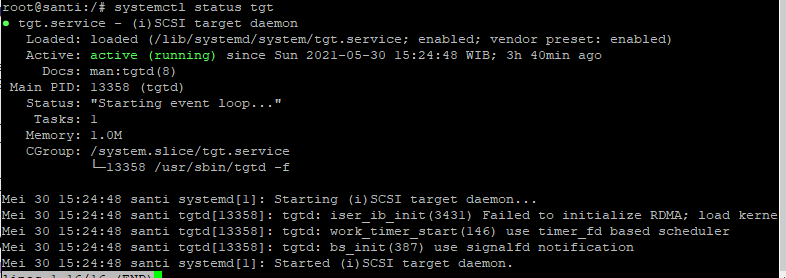
Installing the software for the target server requires **iscsitarget** on Ubuntu and Debian, and an extra

**iscsitarget-dkms** for the kernel modules only on Debian.



##### Penjelesan:

Perintah di atas menunjukkan bahwa sebelumnya saya sudah pernah menginstall tgt, namun karena belum sempat saya screenshot maka saya jalankan Kembali programnya.

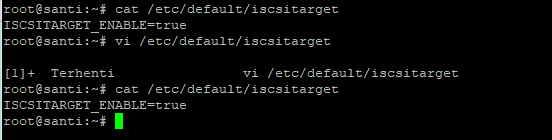


##### Penjelesan:

Perintah di atas digunakan untuk memastikan bahwa status tgt sedang aktiv dan tengah berjalan sesuai dengan tulisan berwarna hijau di atas.

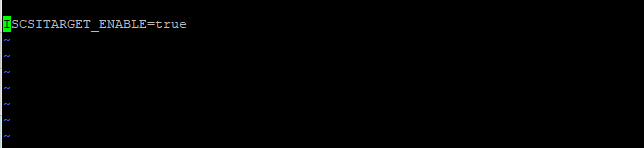
On Debian 6 you will also need **aptitude install iscsitarget-dkms** for the kernel modules, on Debian 5 this is **aptitude install iscsitarget-modules-`uname -a`**. Ubuntu includes the kernel modules in the main package.

The iSCSI target server is disabled by default, so we enable it.



##### Penjelesan:

Perintah di atas untuk menampilkan bahwa iscsitarget dalam /etc/default/iscsitarget berisi ISCSITARGET\_ENABLE=true atau iscsi target memiliki kemungkinan untuk dijalankan.

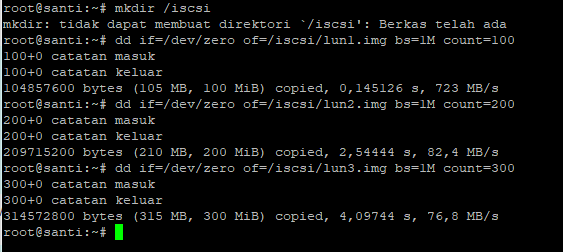


##### Penjelesan:

Isi dari vi /etc/default/iscsitarget yang sama dengan pemakaian chat di atas namun hany berbeda cara menampikannya dengan perintah vi.

**iSCSI target setup with dd files**

You can use LVM volumes (/dev/md0/lvol0), physical partitions (/dev/sda) ,raid devices (/dev/md0) or just plain files for storage. In this demo, we use files created with **dd**. This screenshot shows how to create three small files (100MB, 200MB and 300MB).



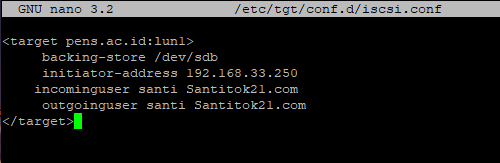
##### Penjelesan:

Membuat sebuah direktori dengan nama iscsi. Lalu dapat menggunakan volume logical yang tidak digunakan, file image, harddisk (misalnya, /dev/sdb), partisi harddisk (misalnya, /dev/sdb1). Disini membuat image file /iscsi/lun1.img yang besarnya 100MB, lun2 200MB, lun3 300MB.



##### Penjelesan:

Masuk kedalam /etc/tgt/conf.d/iscsi.conf untuk menyeting konfigurasi target ISCSI.



##### Penjelesan:

* + - Baris pertama mendefinisikan nama LUN.
    - Baris kedua mendefinisikan lokasi dan nama perangkat penyimpanan di server Target iSCSI.
    - Baris ketiga mendefinisikan alamat IP dari inisiator iSCSI.
    - Baris keempat mendefinisikan nama pengguna/kata sandi yang masuk.
    - Baris kelima mendefinisikan nama pengguna / kata sandi yang akan diberikan target kepada pemrakarsa untuk memungkinkan otentikasi CHAP bersama terjadi.



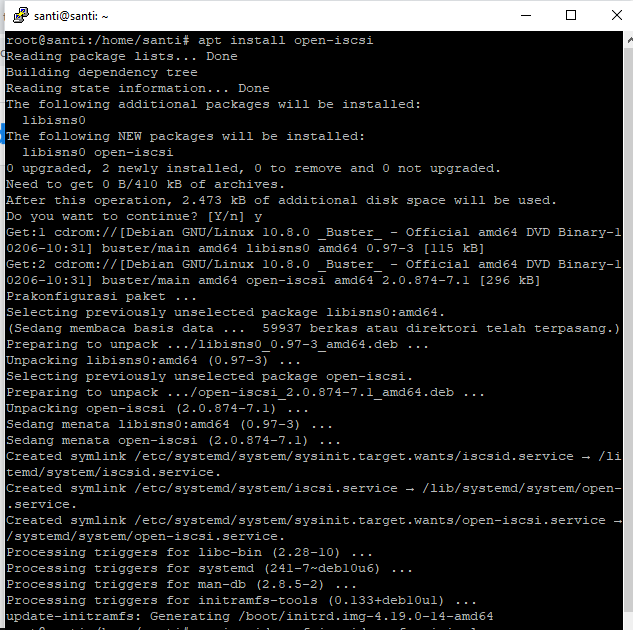
##### Penjelesan:

Selanjutnya, mulai ulang layanan iSCSI untuk menerapkan perubahan.

**I initiator on ubuntu**

First we install the iSCSi client software (on another computer than the target).

root@ubu1104:~# aptitude install open-iscsi

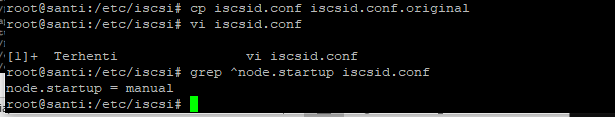


##### Penjelesan:

Melakukan instalasi dengan apt install open-iscsi

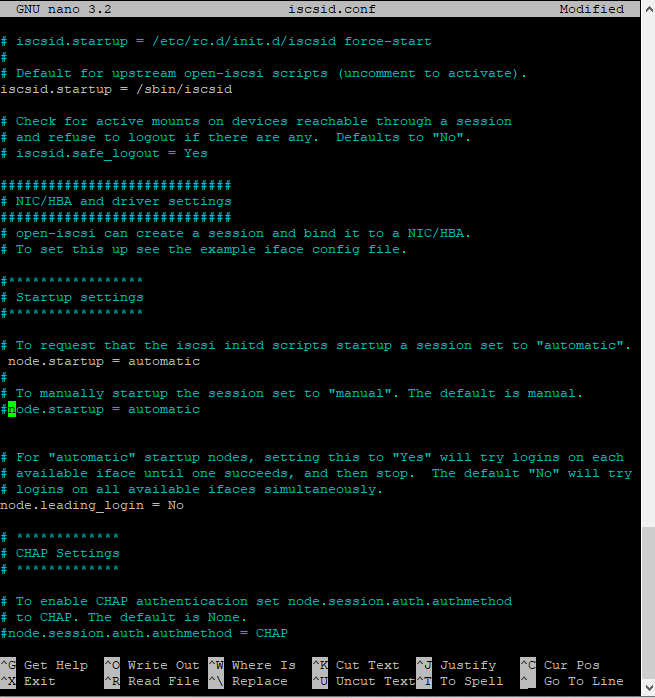
Then we set the iSCSI client to start automatically.

root@ubu1104:/etc/iscsi# cp iscsid.conf iscsid.conf.original root@ubu1104:/etc/iscsi# vi iscsid.conf root@ubu1104:/etc/iscsi# grep ^node.startup iscsid.conf node.startup = automatic



##### Penjelesan:

Melakukan copy file iscsid.conf ke file iscsid.conf.original lalu selanjutnya menampilkan isi dari iscsid.conf dengan perintah vi.



##### Penjelesan:

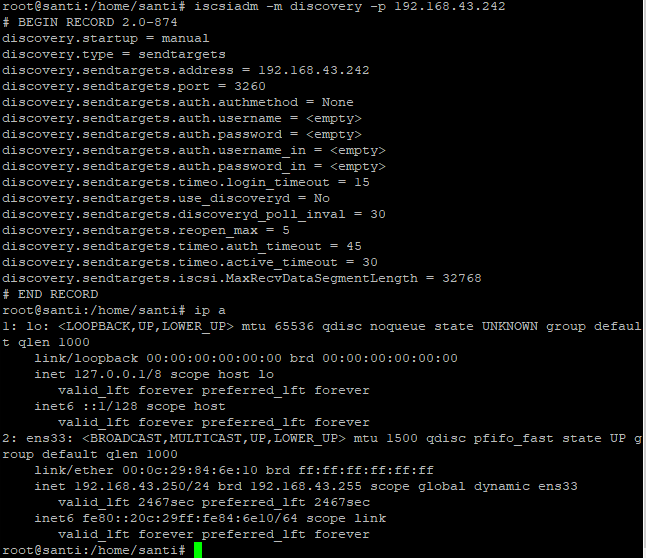
Tampilan vi dari file iscsid.conf iscsid.conf disini mengganti node.startuo yang awalnya manual diganti dengan node.startuo=automatic.



##### Penjelesan:

Menampilkan Kembali dan memastikan bahwa sudah automatic.

Now we can connect to the Target server and use **iscsiadm** to discover the devices it offers:



##### Penjelesan:

Menyambungkan dari inisiator dengan target. Disini ip address dari target adalah 192.68.43.242, sedangkan ip address dari inisiator adalah 192.68.43.250 yang sudah bisa terhubung dengan target.

Kesimpulan

Pada Praktikum Disk System kali ini kita belajar lebih dalam tentang disk drive hard drive dari bab 4 – 12, dari melakukan partisi bab 5. Bab 6 kita melakukan percobaan mengambil file dari dalam partisi yang sudah dibuat. Bab 7 kita belajar melakukan mount dan. Bab 8 mengenai troubleshooting tools untuk melakukan troubleshooting partisi. Bab 9 kita belajar tentang UUID (Universallly unique Identifier). Bab 10 kita dikenalkan dalam macam macam Raid atau Redundant

Array of Independent Disk yang dapat diatur menggunakan perangkat keras atau perangkat lunak. Bab 11 masih belum selesai karena masih banyak error . Bab 12 (bab saya) terakhir kita belajar tentang iSCSI server target dan juga iSCSI klien initiator, iSCSI berguna untuk melakukan hubungan antar device melalu ip address yang sudah di config sebelumnya iSCSI initiator