

GNU/Linux

Fundamental

Pengantar

Assalamualaikum Warahmatullahi wabarakatu.

Alhamdulillah berkat rahmat Allah yang maha kuasa, kami sebagai tim penulis diberikan kesempatan, kesehatan untuk mengerjakan dan menyelesaikan pembuatan Buku panduan “Linux Fundamental”, ini dengan tepat waktu. Buku ini ditulis secara detail membahas tentang permasalahan-permasalahan dalam Linux tentunya secara mendasar (fundamen), dengan tujuan agar pengguna dapat memahami struktur & cara kerja Sistem Operasi berbasis Linux secara mendasar, dengan cara pandang yang mudah dimengerti.

Buku Panduan ini disusun dengan tujuan untuk mempermudah pembaca dalam memahami dan menggunakan sistem Linux secara keseluruhan dengan cara yang sederhana, menarik dan mudah. Sehingga dapat membantu masyarakat secara luas untuk bermigrasi untuk menggunakan aplikasi (software) Open Source secara menyeluruh. Buku panduan ini disusun sebagai panduan / pegangan bagi para pembaca yang baru mengenal dan mempelajari sistem operasi berbasis Linux, dengan harapan agar buku panduan ini dapat mempermudah pengguna dalam mempelajari Linux dengan cepat secara menyeluruh.

Harapan kami sebagai penulis buku ini dapat dimanfaatkan dengan sebaik-baiknya untuk mempermudah dan mempercepat proses pembelajaran dan migrasi masyarakat untuk menggunakan Software Open Source.

Gorontalo, Februari 2014

Tim Penulis

DAFTAR ISI

Sampul Depan	1
Pengantar	2
Daftar Isi	3
Bagian I Pengenalan Linux dan OSS	4
1.1 Pengenalan Open Source.....	5
1.2 Pengenalan GNU/Linux.....	7
1.3 Pengenalan Distribusi Linux.....	8
Bagian II Instalasi Linux	11
2.1 Kebutuhan Instalasi.....	12
2.2 Persiapan Instalasi.....	13
2.3 Tahapan Instalasi.....	14
Bagian III Konsep & Manajemen Partisi	23
3.1 Pengenalan Storage & Partisi	24
3.2 Primary & Logical Partisi.....	25
3.3 Filesystem Linux.....	26
3.4 Memory Virtual (Swap memory).....	28
3.5 Pembuatan Partisi Hardisk & Mounting.....	29
Bagian IV Manajemen Pemaketan	41
4.1 Sistem Pemaketan Distro.....	42
4.2 Sistem Server Repository.....	43
4.3 Alat Manajemen Pemaketan.....	44
Bagian V BASH (Bourne Again Shell)	48
5.1 Navigasi Shell.....	52
5.2 Manipulasi File & Direktori.....	57
5.3 Perintah Dasar Shell.....	68
Bagian VI Struktur Filesystem	75
Bagian VII Manajemen File	79
7.1 Mode Kepemilikan.....	80
7.2 Mode Akses.....	83
Bagian VIII Manajemen User	87
Bagian XI Manajemen Jaringan	97
Penutup	109
Referensi Pustaka	110

Bagian I

Pengenalan Linux & Open Source Software

1.1 Pengenalan Open Source

Open Source atau bisa disebutkan sebagai Sumber Terbuka dalam bahasa Indonesia, bisa dikatakan sebuah gagasan, filosofi, ataupun gerakan, untuk mengembangkan perangkat lunak (software) komputer, secara bebas tanpa adanya batasan-batasan yang mengikat / mengikuti software tersebut, sehingga secara bebas dapat digunakan, dikembangkan, dan didistribusikan tanpa adanya batasan-batasan yang mengikat software tersebut.

Open Source dapat dianalogikan sebagai sebuah cara untuk membuat sebuah Produk Kue secara terbuka, dalam hal ini yang dibuat terbuka adalah bahan & cara pembuatannya (resep) yang digunakan untuk membuat kue tersebut, sehingga sangat mungkin untuk dikembangkan kembali oleh orang lain, apabila kue tersebut memiliki sebuah kekurangan / terdapat sebuah kesalahan.

Apabila analogi diatas lebih diperjelas, yaitu bagaimana cara untuk mengembangkan sebuah perangkat lunak komputer secara terbuka, agar bebas didistribusikan bersama dengan Source Code (kode sumber / cara pembuatannya), sehingga bisa dikembangkan kembali secara bebas, tanpa diikat oleh hak dari pembuat awal maupun pengembang. Apabila berbicara tentang sebuah software akan sangat lekat dengan istilah CopyRight dan Copyleft.

Copyright merupakan sebuah pengembangan produk (dalam hal ini perangkat lunak), dimana didalamnya berisi ikatan terhadap produk dan pembuatnya secara utuh. Sedangkan **Copyleft** pada dasarnya mengadopsi prinsip copyright, namun prinsip tersebut digunakan untuk menjamin kebebasan berkreasi. Jaminan tersebut berbentuk pelampiran kode sumber (source code), serta pernyataan bahwa perangkat lunak tersebut boleh dimodifikasi asalkan tetap mengikuti prinsip awalnya yaitu copyleft.

Sejak tahun 2004, Indonesia sebenarnya sudah mengadopsi gaya pengembangan perangkat lunak (software) secara terbuka dengan konsep OpenSource (Copyleft), hal tersebut dibuktikan dengan adanya penerbitan Deklarasi 5 Menteri terhadap Himbuan penggunaan Open Source, seperti yang dilampirkan secara jelas pada gambar berikut :



Dengan adanya Deklarasi diatas, kemudian dibentuklah gerakan kecil seperti FOSS-ID (Free Open Source Software Indonesia) dan Indonesia Goes Open Source 2013.



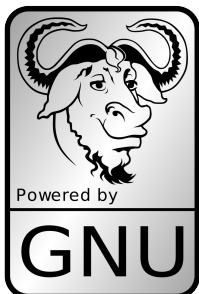
Gerakan-gerakan diatas sudah berjalan sejak deklarasi Open Source, sampai dengan sekarang ini, akan tetapi perkembangan Open Source di Indonesia, banyak terhambat dengan adanya kepentingan-kepentingan lainnya, sehingga membuat perkembangan opensource tersendat, untuk mengadopsi Open Source secara merata & menyeluruh di Indonesia perlu dukungan dan gerakan dari semua lapisan masyarakat, karena dengan Open Source kita bisa belajar untuk kreatif, dan mandiri.

1.2 Pengenalan GNU/Linux

Jika berbicara tentang Sistem Operasi berbasis Linux, erat kaitannya dengan GNU (Genuine not Unix). Kedua hal tersebut merupakan dua komponen utama yang membuat perkembangan Open Source berkembang pesat sampai dengan saat ini, tanpa adanya peran aktif dari proyek GNU, Linux tidak mungkin bisa berkembang menjadi sekarang ini, begitupun sebaliknya.

Untuk itu selain mengenal tentang Linux, kita perlu mengenal lebih detail tentang GNU (Genuine Not Unix), yang apabila kita translasikan, akan berarti "*Bukan Mesin unix*", istilah konsep tersebut digagaskan pertama kali oleh Richard Matius Stallman pada pertengahan Tahun 1980, berawal dari keresahan pembatasan hak (lisensi) penggunaan sebuah perangkat lunak.

Unix merupakan sebuah perangkat lunak yang tersusun dari bagian-bagian yang berbeda-beda dan dikembangkan secara komersil. Stallman merintis pengembangan perangkat lunak yang berorientasi pada kebebasan pengguna. GNU bertujuan untuk memberdayakan kebebasan para pengguna dengan menggunakan, membagikan, mendistribusikan, dan mengembangkan kembali sebuah perangkat lunak.



Maskot GNU

Konsep dari proyek GNU ini lebih dikenal dengan istilah "**Free Software**". Konsep GNU yang dirintis oleh Stallman, konsep tersebut banyak diadopsi dan digunakan oleh banyak kalangan, baik kelompok, perusahaan maupun individu yang bergerak dibidang pengembangan perangkat lunak (software)

Pada awal tahun 1991 seorang Mahasiswa Finlandia bernama Linus Torvalds, telah merilis sebuah kernel sistem operasi yang diberi nama Linux (Like Unix). Linux adalah sebuah kernel yang didesain secara unik yang didesain mirip dengan Unix, Linux mengutamakan kecepatan pengaksesan data, dan menggunakan prinsip kerja penggunaan kernel monolithic.



Linus Torvalds

Linux dikembangkan dengan lisensi publik (umum) GNU/GPL (Copyleft), sehingga Linux bisa berkembang secara pesat sampai dengan saat ini dengan banyaknya varian Distribusi Linux yang dikembangkan oleh orang dari seluruh dunia, termasuk di Indonesia.

1.3 Pengenalan Distribusi Sistem Operasi Linux

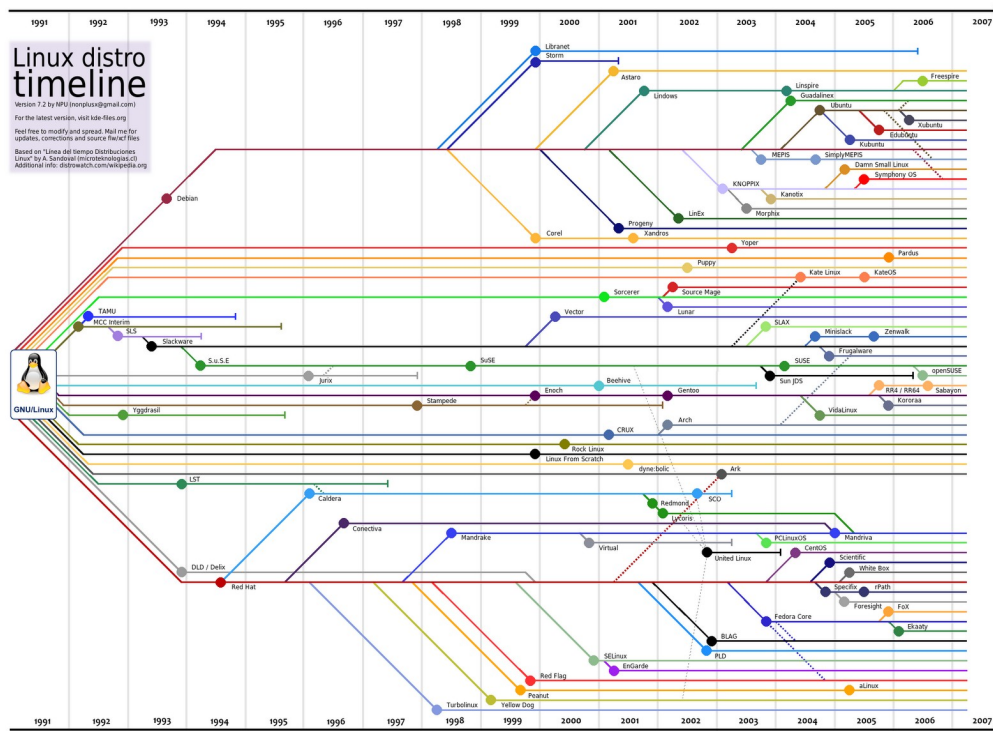
Linux dikembangkan dengan basis pengembangan terbuka (GNU/GPL), sehingga hal tersebut memungkinkan Linux dikembangkan oleh banyak orang sesuai dengan keinginan dan kebutuhan, Varian / hasil pengembangan Sistem Operasi berbasis GNU/Linux dapat kita kenal dengan Distro (Distribusi Sistem Operasi). Linux bisa berkembang sangat pesat sampai dengan saat ini dikarenakan Linux mengutamakan pengembangan secara terbuka. Sehingga Linux bisa berkembang dengan sangat cepat dalam kurun waktu yang sangat Singkat.

Linux memiliki distribusi awal yang dikembangkan oleh beberapa komunitas, individu maupun perusahaan pengembang software. Distribusi tersebut adalah Debian, Redhat dan Slackware.



Masing-masing distro Linux memiliki kelebihan dan kekurangan masing-masing, kesemua Distro Linux dimewariskan sifat dasar dari kernel Linux, yaitu keamanan, kecepatan pemrosesan, dan lisensi yang sama yaitu GNU/GPL.

Dari ketiga distro Linux tersebut kemudian pengembangan Linux menjadi sangat pesat sampai dengan saat ini. Berikut adalah timeline pengembang distro Linux yang (terdata secara resmi) sejak tahun 1991 sampai dengan 2007.



Perkembangan Distro Linux (1991-2007)

Dari ketiga Distro Linux diatas, kemudian linux berkembang menjadi banyak turunan, banyak ragam, yang dikembangkan oleh banyak orang sesuai dengan keinginan, kebutuhan dan selera orang-orang disemua negara.

Seperti Indonesia, Indonesia sudah konsisten untuk mengembangkan Distro Linux dengan nama **Blankon**, **IGN** (IGOS Nusantara), dan masih banyak lagi yang telah terdaftar maupun tidak terdaftar. Tiap harinya distro linux dikembangkan dengan banyak varian sesuai dengan kebutuhan dan keinginan masing-masing pengembang. Pengembang (Developer) Distro Linux bisa siapa saja, baik Individu, Instansi, Kelompok, maupun Komunitas.



Distro Linux karya anak Indonesia

Hal tersebut yang membuat GNU/Linux berkembang dengan sangat pesat, Puluhan Distro Linux dikembangkan tiap harinya untuk kebutuhan yang beragam. Berbagi pengembangan distro linux secara resmi dapat anda akses melalui website *distrowatch.com*,



Website Distrowatch.com

Pada website distrowatch.com telah terdaftar ratusan distro linux yang dikembangkan secara yang sudah dibuatkan rating penggunaan / kepopuleran distribusi linux tersebut, adapun rating tersebut akan menampilkan distro linux yang paling banyak digunakan oleh pengguna komputer.

Peringkat Kunjungan Halaman		
Data span:		
Last 6 months <input type="button" value="Segarkan"/>		
Peringkat	Distribusi	H.P.D*
1	Mint	3593▲
2	Debian	1907▲
3	Ubuntu	1901▲
4	Mageia	1532▲
5	Fedora	1399▲
6	openSUSE	1338▲
7	PCLinuxOS	1093▼
8	Manjaro	1000▲
9	Arch	951▲
10	Puppy	884▲

Bagian II

Instalasi Linux

Panduan Instalasi pada bagian ke II ini, kami akan menggunakan Distribusi Linux Ubuntu, kenapa memilih Ubuntu ???

“Ya, karena Ubuntu merupakan distribusi Linux paling banyak digunakan di Indonesia, bisa dibilang populer, mudah, dan memiliki komunitas yang besar di Indonesia (<http://ubuntu-indonesia.com>). Selain itu Ubuntu juga banyak digunakan oleh Distribusi Linux lokal (Blankon), dan banyak distribusi Linux lainnya.”



Ubuntu merupakan salah satu distribusi Linux yang berbasis Debian dan didistribusikan sebagai perangkat lunak bebas. Nama Ubuntu berasal dari filosofi dari Afrika Selatan yang berarti "*kepedulian kepada sesama*". Ubuntu dirancang untuk kepentingan penggunaan pribadi, namun tim pengembang ubuntu juga menyediakan versi Server.



Proyek Ubuntu resmi disponsori oleh *Canonical Ltd.* yang merupakan sebuah perusahaan yang dimiliki oleh pengusaha Afrika Selatan *Mark Shuttleworth*.

Tujuan dari distribusi Linux Ubuntu adalah membawa semangat yang terkandung di dalam filosofi Ubuntu ke dalam dunia perangkat lunak.

2.1 Kebutuhan Instalasi

Berikut adalah beberapa kebutuhan dasar spesifikasi yang dibutuhkan untuk melakukan instalasi Ubuntu :

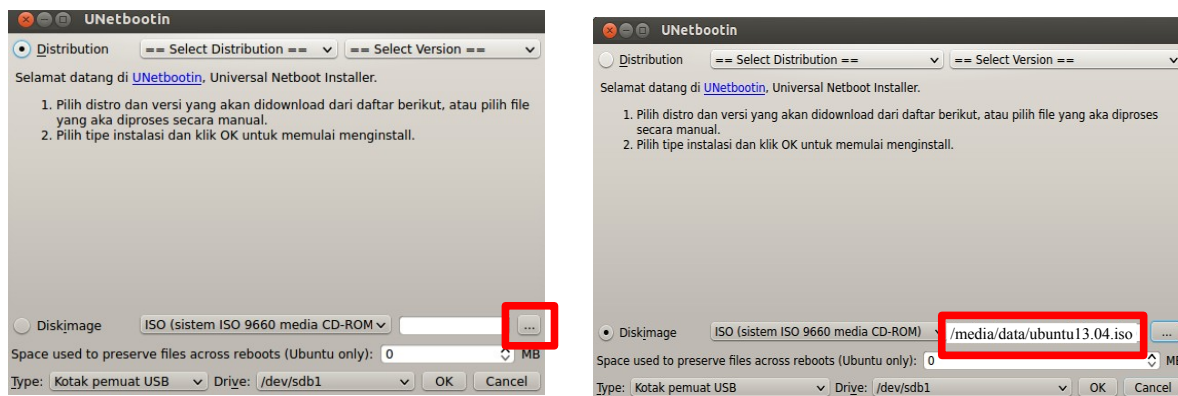
KEBUTUHAN MINIMAL	SERVER	DESKTOP
PROCESSOR	300 MHz	700 MHz
MEMORY (RAM)	128 MiB	384 MiB
HARDISK DRIVE	1 GB	5 GB
MONITOR RESOLUSI	640 x 480	640 x 480

2.2 Persiapan Instalasi

Pastikan anda sudah memiliki installer distribusi Linux Ubuntu, anda bisa mendapatkan melalui mirror download terdekat yaitu melalui tautan berikut : <http://repo.ugm.ac.id/iso/ubuntu>, <http://kambing.ui.ac.id/iso/ubuntu>, dll.

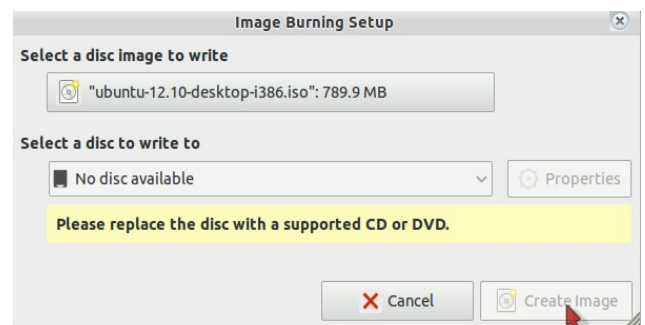
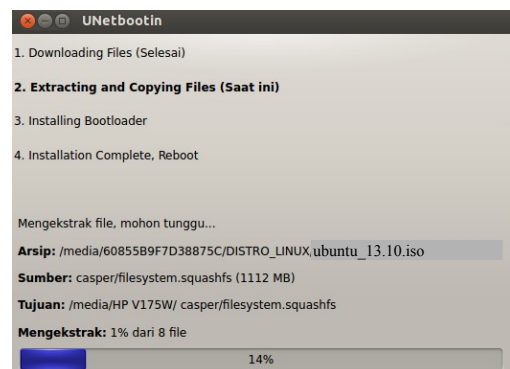
Setelah itu siapkan media instalasi yang dapat berbentuk USB Flashdrive atau CD/DVD. Jika anda menggunakan flashdisc sebagai media instalasi, gunakan aplikasi Unetbootin, yang dapat anda download melalui tautan berikut : <http://unetbootin.sourceforge.net/>.

Setelah proses pengunduhan selesai selanjutnya siapkan flashdisc dengan ukuran minimal **1 GB**, karena ukuran file instalasi Linux Ubuntu 13.10 berukuran kurang lebih 950 MB. Selanjutnya hubungkan media instalasi (flashdrive) yang akan digunakan, selanjutnya buka aplikasi Unetbootin, dan selanjutnya ikuti tahapan berikut :



Setelah itu tekan tombol **[OK]** untuk melanjutkan sehingga akan ditampilkan pada gambar disamping.

Apabila menggunakan media instalasi CD/DVD, anda bisa menggunakan perangkat lunak disc burner seperti **brasero** (untuk Linux) atau **UltraISO** dan Nero Burner (untuk Windows). Seperti yang ditunjukkan pada gambar berikut :



2.3 Tahapan Instalasi

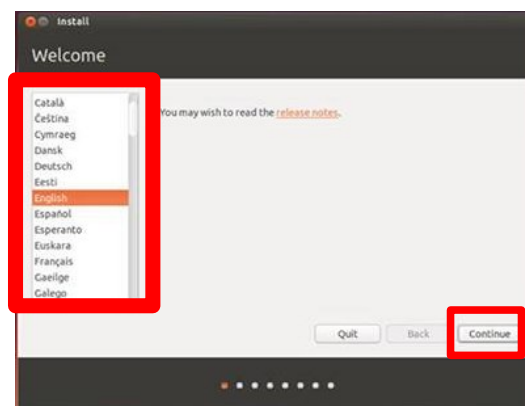
Proses instalasi kali ini, menggunakan distribusi Ubuntu dengan versi 13.10, untuk memulai proses instalasi hubungkan media instalasi dengan perangkat komputer (laptop), selanjutnya lakukan reboot dan lakukan perubahan pengaturan “**boot priority / 1st first boot**” pada pengaturan BIOS (Basic Input Output System) (untuk masuk ke pengaturan BIOS anda bisa menggunakan beberapa tombol key seperti F2, F11, F12, Esc, atau Del). Sehingga proses booting akan berjalan dan masuk ke desktop Ubuntu 13.10 seperti yang ditunjukkan pada gambar berikut :



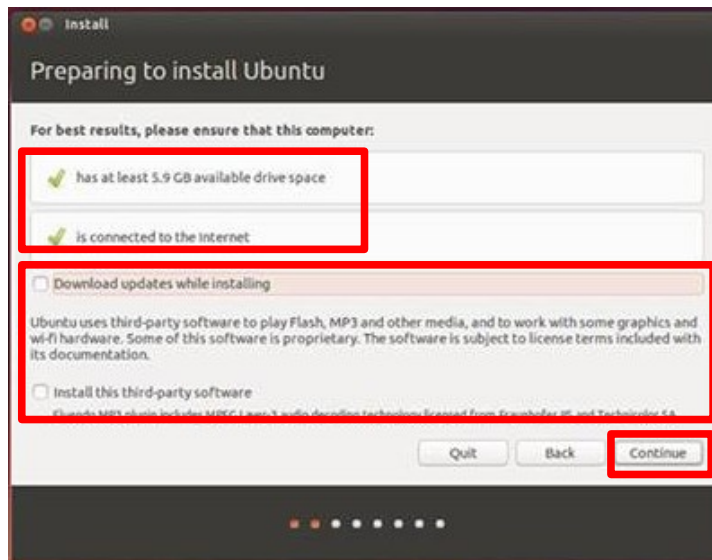
Untuk melakukan instalasi Ubuntu 13.10 dapat dengan melakukan klik ganda pada icon desktop “Install Ubuntu 13.10”, seperti yang ditunjukkan pada gambar disamping. Selanjutnya akan ditampilkan window berikut, yang merupakan tahapan awal instalasi yaitu, pemilihan penggunaan bahasa yang



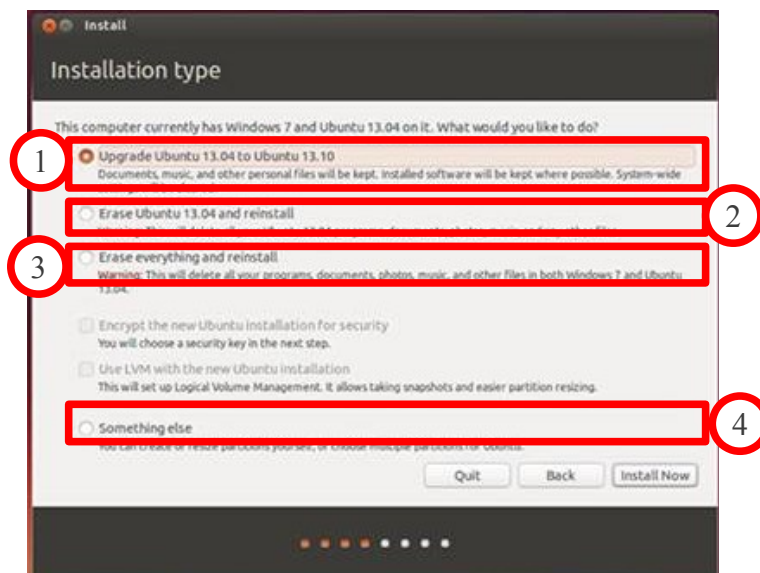
akan digunakan selama proses instalasi berjalan. Apabila sudah melakukan pemilihan bahasa, selanjutnya tekan tombol “Continue”.



Setelah menekan tombol “Continue / Lanjutkan”, selanjutnya akan ditampilkan halaman berikut :



Seperti yang ditunjukkan pada gambar diatas, pada tahapan ini, sistem akan melakukan pengecekan untuk persiapan instalasi, beberapa komponen yang akan dicek adalah : Ketersediaan Space Hardisk, dan Akses / Koneksi ke Internet. Selanjutnya apabila kita terkoneksi dengan internet kita bisa menjalankan proses download dan update perangkat lunak tambahan dan perangkat lunak pihak ketiga. Untuk Melanjutkan ke tahapan selanjutnya, tekan “Continue”.



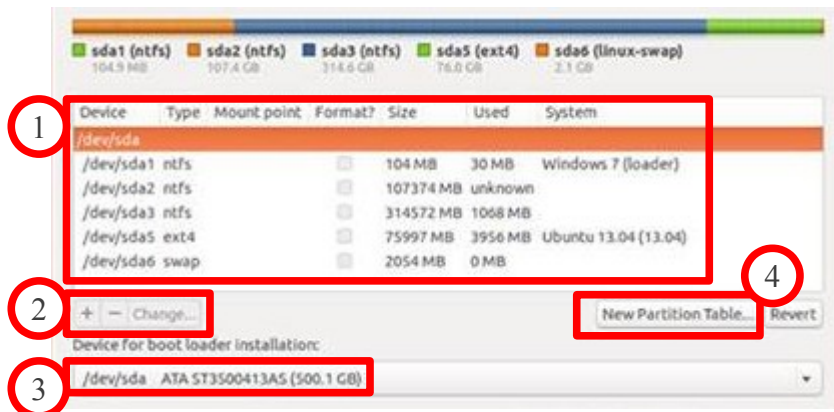
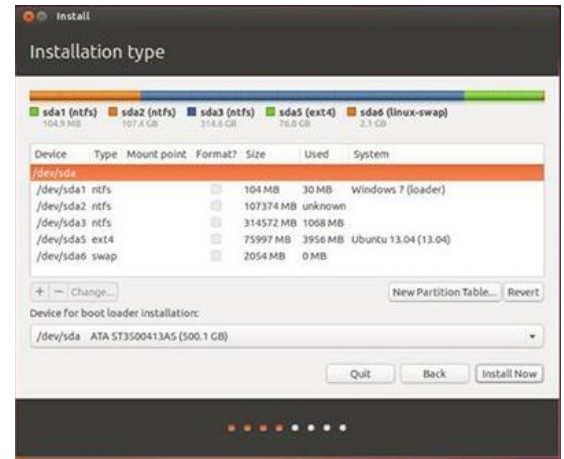
Tahapan diatas merupakan tahapan pemilihan tipe instalasi yang akan digunakan, yang terbagi menjadi beberapa tipe instalasi diantaranya adalah :

1. **Upgrade Ubuntu 13.04 to Ubuntu 13.04.** Opsi ini berarti kita akan melakukan upgrade Ubuntu 13.04 (yang sebelumnya terinstall di perangkat komputer) dengan Ubuntu 13.10, sehingga versi yang sebelumnya 13.04 menjadi versi yang terbaru yaitu 13.10.
2. **Erase Ubuntu 13.04 and reInstall.** Opsi ini berarti kita akan menghapus Ubuntu 13.04 dan menginstall Ubuntu 13.10, kedalam partisi hardisk yang sebelumnya ditempati oleh Ubuntu 13.04.
3. **Erase Everything and reInstall.** Opsi ini akan menghapus semua ruang (partisi) hardisk, dan mengisi / menginstall Ubuntu secara keseluruhan kedalam hardisk.
4. **Something Else.** Opsi ini berfungsi mempermudah kita melakukan manajemen partisi hardisk dan partisi yang akan digunakan sebagai media Instalasi Ubuntu 13.10.

Pada tahap ini kita akan memilih opsi ke-4 (Something Else), karena kita akan belajar bagaimana cara mempersiapkan ruang partisi yang akan digunakan untuk menyimpan file instalasi Ubuntu 13.10, centang “Continue”.

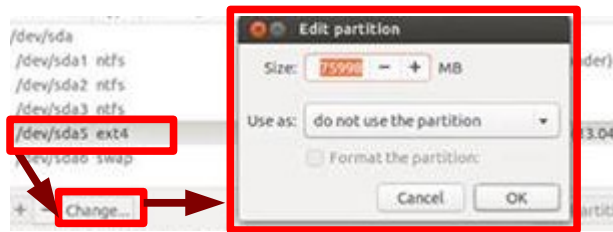


Selanjutnya akan ditampilkan window disamping, dimana terdapat beberapa informasi tentang hardisk, ukuran hardisk, dan informasi tentang partisi yang ada di hardisk secara keseluruhan. Pada Tahapan ini kita akan menjelaskan beberapa informasi penting terkait pengaturan partisi yang terdapat pada gambar dibawah :

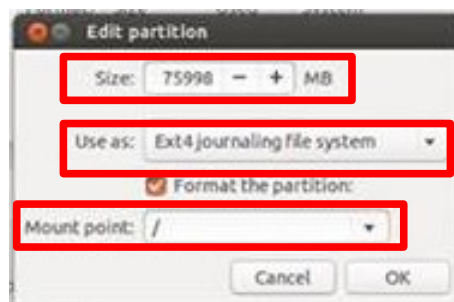


- Informasi Partisi, ukuran, dan atribut sistem**, pada menu ini berisi informasi tentang partisi yang ada didalam hardisk secara detail, mulai dari letak device, Type Filesystem, Mountpoint, ukuran partisi secara keseluruhan, ukuran partisi yang sudah digunakan, dan sistem operasi yang sudah terinstall dalam partisi-partisi tersebut.
- Navigasi pembuatan dan penghapusan partisi**, pada menu ini kita bisa melakukan beberapa navigasi aksi, seperti perubahan partisi (**change**), penghapusan partisi (**tombol (-)**), maupun penambahan partisi (**tombol (+)**),
- Hardiskdrive yang digunakan**, pada menu ini menginformasikan tentang hardiskdrive yang digunakan pada saat installasi saat ini, sesuai dengan informasi diatas ukuran partisi berjumlah 500 GB.
- Pembuatan Tabel Partisi baru**, pada menu ini berfungsi untuk membuat format pengaturan tabel partisi baru, sehingga semua pengaturan partisi dan data akan terhapus secara keseluruhan.

Pada tahapan ini kita akan melakukan perubahan pada partisi yang sebelumnya terinstall Ubuntu 13.04, dengan cara melakukan klik pada partisi yang ingin dipartisi selanjutnya menekan tombol “Change”, selanjutnya akan ditampilkan windows berikut :



Seperti yang ditunjukkan pada gambar diatas, kita perlu menentukan ukuran partisinya, selanjutnya penggunaan partisinya yaitu “Use as”, pada tahapan ini pilih tipe partisi “Ext4 Journalling file system”, selanjutnya lakukan centang pada opsi “Format the Partition”, selanjutnya arahkan Mount Point menjadi *root (atau yang disimbolkan dengan slash (/))*. seperti yang ditunjukkan pada gambar berikut :



Catatan :

Alokasi ukuran partisi untuk filesystem beragam sesuai dengan kebutuhan, minimal ukuran partisi yang dibutuhkan untuk memuat instalasi dasar yaitu 3 GB.

Perhatikan pengaturan partisi diatas, apabila sudah sesuai dengan pengaturan partisi diatas, selanjutnya tekan tombol “Ok”, sehingga pengaturan partisi akan berubah, sesuai dengan pengaturan partisi sebelumnya, yang dapat ditunjukkan pada gambar berikut :

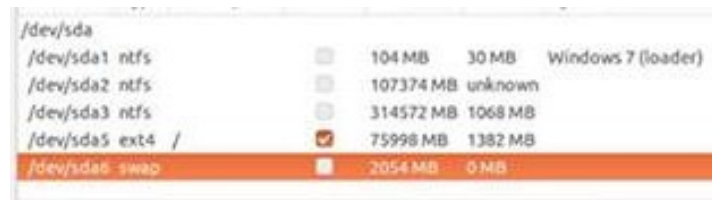
Device	Type	Mount point	Format?	Size	Used	System
/dev/sda						
/dev/sda1	ntfs		<input type="checkbox"/>	104 MB	30 MB	Windows 7 (loader)
/dev/sda2	ntfs		<input type="checkbox"/>	107374 MB	unknown	
/dev/sda3	ntfs		<input type="checkbox"/>	107374 MB	unknown	
/dev/sda4	ext4	/	<input checked="" type="checkbox"/>	75998 MB	1382 MB	
/dev/sda5	swap		<input type="checkbox"/>	2024 MB	0 MB	

Setelah membuat partisi untuk filesystem, selanjutnya kita perlu membuat partisi swap (virtual memory), yang bertujuan untuk meringankan beban dari memory fisik, aturan dasar pengalokasian ukuran memory virtual (swap memory) secara umum dapat mengikuti aturan berikut : **swap = 2 x ukuran memory fisik**

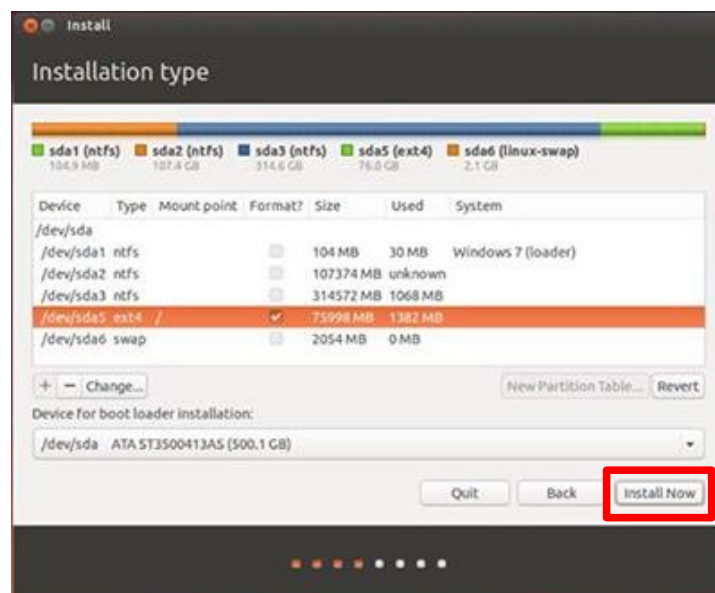
Yang berarti bahwa, apabila kita menggunakan memory fisik dengan ukuran 1 GB, maka alokasi partisi swap adalah **2 GB**, karena **1 GB x 2**. Selanjutnya lakukan pembuatan partisi untuk memory swap, seperti yang ditunjukkan pada gambar disamping :



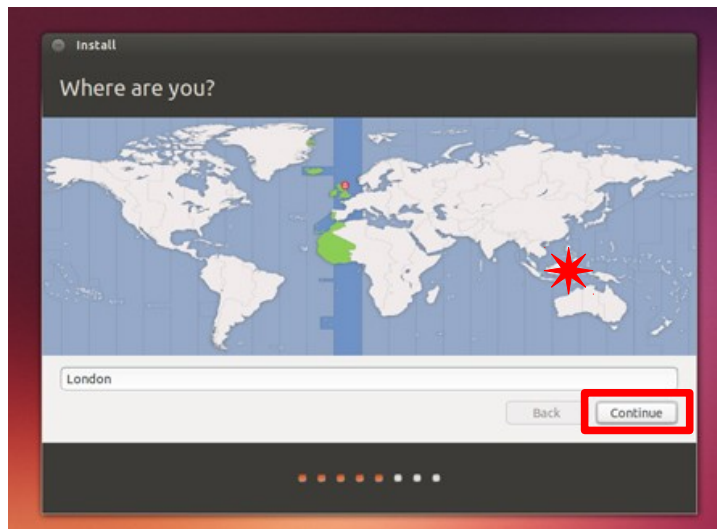
Selanjutnya apabila pembuatan partisi untuk swap memory, telah selesai. Pengaturan partisi akan berubah secara otomatis sesuai dengan perubahan yang dilakukan sebelumnya, seperti yang ditunjukkan pada gambar berikut :



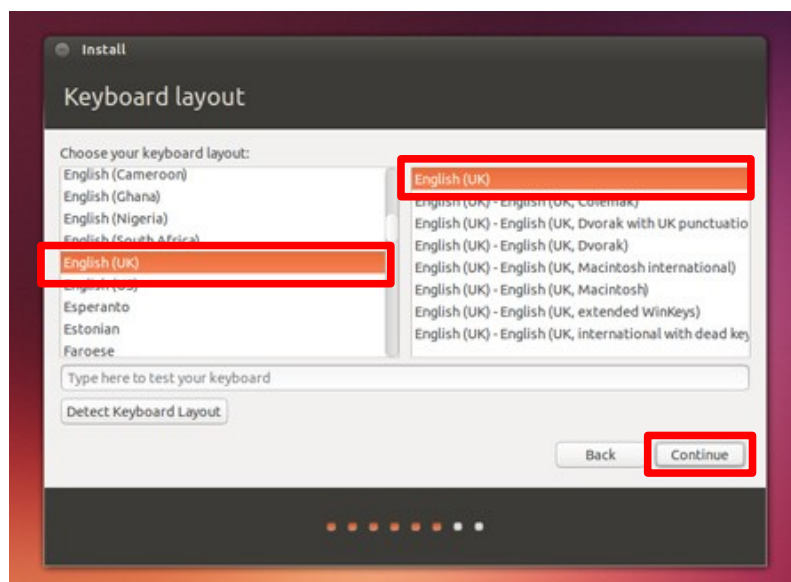
Apabila pengaturan partisi telah selesai, sesuai dengan kebutuhan selanjutnya tekan tombol “Install Now” seperti yang ditunjukkan pada gambar dibawah :



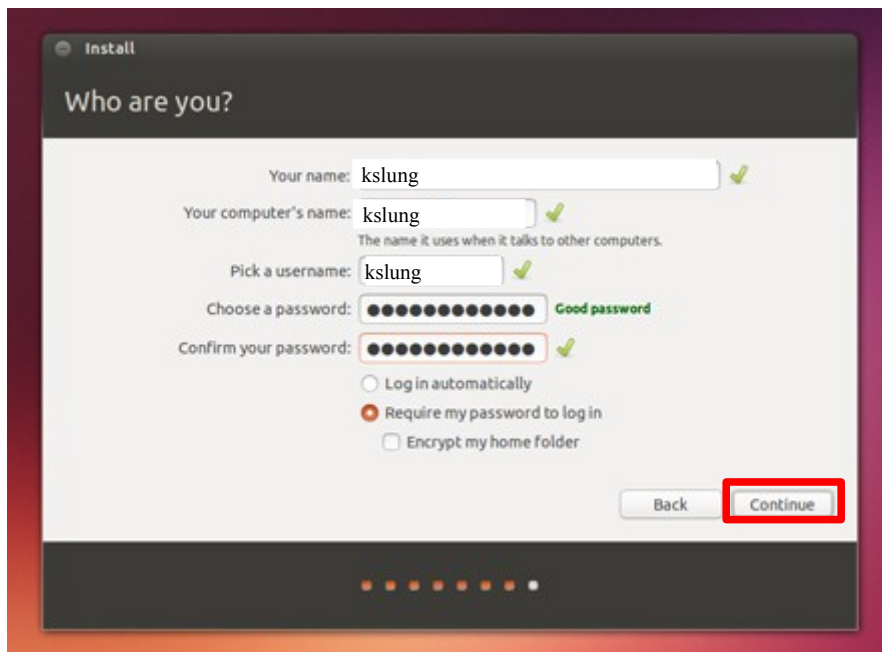
Pada tahapan selanjutnya, tentukan lokasi teritorial kita saat ini, sesuai dengan GMT yang ditentukan, dengan cara melakukan klik pada map, selanjutnya tekan tombol Continue untuk melanjutkan ke tahapan selanjutnya.



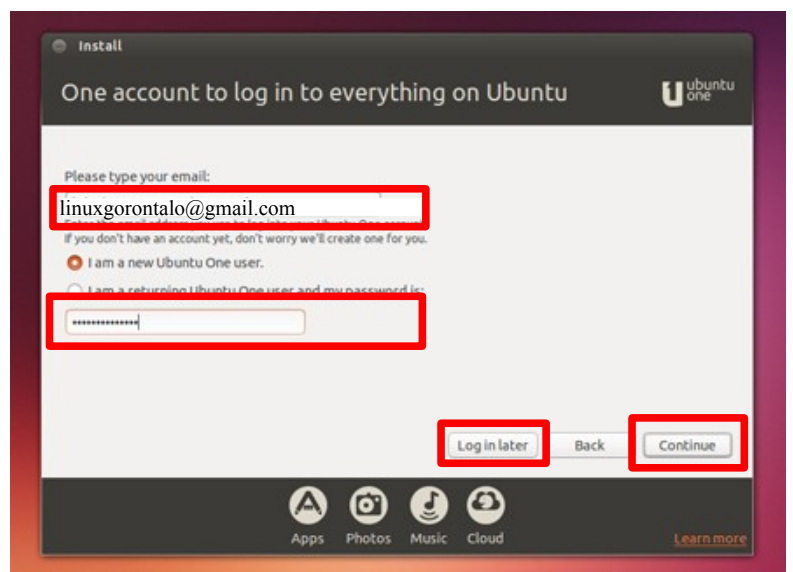
Selanjutnya kita akan dimintakan mengatur susunan (layout) papan keyboard yang anda gunakan, karena secara umum standar susunan papan keyboard yang digunakan di Indonesia adalah QWERTY, maka secara default pilih **“English UK”**. Selanjutnya tekan tombol Continue, untuk melanjutkan ke tahapan selanjutnya.



Selanjutnya kita akan dimintakan melakukan pengisian data pengguna sistem, username dan password untuk sistem Ubuntu 13.10 yang akan di install dan digunakan ketika proses installasi telah selesai. Apabila pengisian format pada tahapan ini selesai selanjutnya tekan tombol “Continue”.



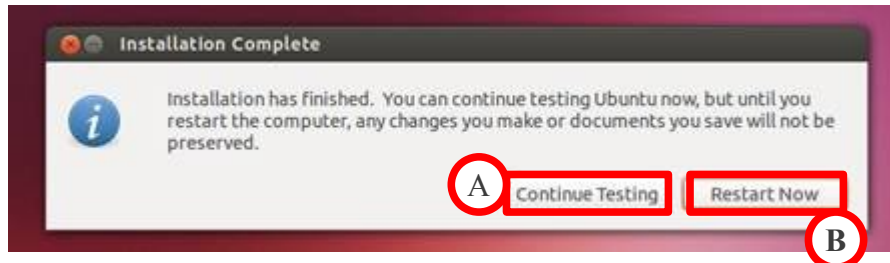
Tahapan selanjutnya yaitu tahapan, pembuatan akun untuk UbuntuOne (Cloud), UbuntuOne merupakan fasilitas Cloud yang disediakan oleh Ubuntu untuk memediasi akses dan layanan penyimpanan data, dan informasi pada infrastruktur Cloud yang disediakan oleh UbuntuOne. Lakukan penginputan data yang diperlukan seperti **alamat email**, dan password, untuk membuat akun UbuntuOne secara otomatis, dengan syarat komputer / laptop yang digunakan harus terkoneksi dengan layanan internet. Apabila semua data telah terisi silahkan pilih opsi **“Continue”**, apabila anda tidak ingin melakukan pembuatan akun / login ke UbuntuOne saat ini, anda bisa melakukan login / registrasi setelah instalasi selesai, dengan memilih opsi **“Log In Later”**.



Selanjutnya proses instalasi akan berjalan seperti yang ditunjukkan pada gambar berikut, silahkan menunggu beberapa menit, sampai proses instalasi selesai



Apabila proses instalasi telah selesai, akan ditampilkan window berikut, lakukan restart apabila ingin menggunakan Distro Linux Ubuntu 13.10 yang telah terinstall dengan memilih opsi B : **“Restart Now”**, apabila ingin melanjutkan pengetesan bisa memilih opsi A : **“Continue Testing”**



Setelah proses booting selesai, berarti proses instalasi telah berhasil dan sukses, Selanjutnya sistem akan masuk ke sistem Ubuntu 13.10 yang telah berhasil di install, selanjutnya lakukan login dengan menggunakan username dan password yang telah diatur pada proses instalasi sebelumnya. Selanjutnya kita akan bisa login kedalam sistem.

Bagian III

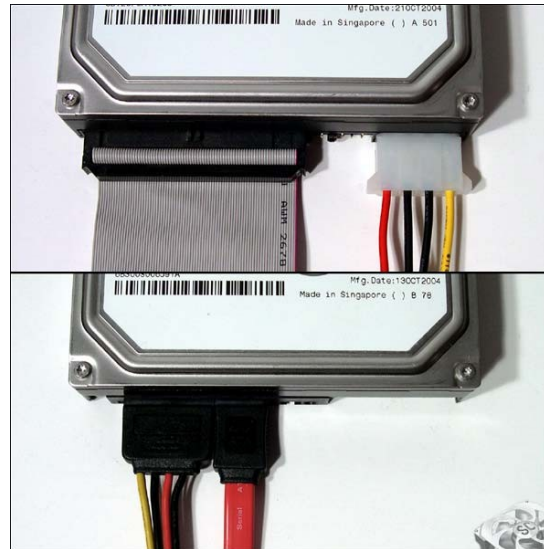
Konsep & Manajemen Pemartisian

Jika pada bagian II sebelumnya yaitu tahapan instalasi dalam proses instalasi kita diwajibkan untuk menyediakan ruang penyimpanan (partisi) hardisk untuk melakukan penyimpanan media instalasi kedalam sistem, pada bagian III ini, kita akan membahas secara lengkap tentang pemartisian.

3.1 Pengenalan media penyimpanan & Pemartisian

Sistem Linux mengenali media penyimpanan hardisk (storage) dalam istilah sebagai berikut :

- **hd (hardisk drive)**, sistem mengenali media penyimpanan hardisk dengan model **ATA** yang terkoneksi dengan sistem dengan istilah **hd**.
- **sd (storage device)**, sistem mengenali media penyimpanan hardisk dengan model **SATA** dengan istilah **sd**



Hardisk ATA (atas), SATA (bawah)

Jika kita memiliki lebih dari 1 (satu) hardisk dalam perangkat komputer kita, maka maka hardisk tersebut akan dikenali secara berurutan sesuai dengan susunan alfabet berikut :

- **hda, hdb, hdc, hdd,.....** jika anda menggunakan hardisk ATA, maka hardisk anda akan dikenali sesuai dengan abjad sesuai dengan format diatas
- **sda, sdb, sdc, sdd,.....** jika anda menggunakan hardisk ATA, maka hardisk anda akan dikenali sesuai dengan abjad sesuai dengan format diatas

Jika sebelumnya dijelaskan tentang konsep pengenalan hardisk / media penyimpanan selanjutnya kita akan belajar terkait partisi.

Partisi merupakan sebuah ruang penyimpanan yang dialokasikan dari media penyimpanan hardisk (storage) untuk penamaan partisi pada masing-masing hardisk adalah dengan menggunakan format angka untuk masing-masing partisinya.

sda				
sda1	sda2	sda3	sda4	sda5
sdb				
sdb1	sdb2	sdb3	sdb4	sdb5

Semua media penyimpanan seperti hardisk internal, hardisk eksternal, maupun flashdrive, pada sistem linux akan selalu dikenali dengan format diatas. Pengaturan secara keseluruhan pada media penyimpanan (Storage device) akan tersimpan pada direktori (**/dev**), penjelasan lengkapnya akan dijelaskan pada bagian VII.

3.2 Partisi Primary dan Logikal

Sebuah sistem operasi seperti linux atau lainnya mengenal dua model pemartisian yaitu :

- **Partisi Primary** : Jenis partisi ini merupakan jenis partisi utama diharddisk untuk system operasi umumnya. Partisi primary hanya bisa dibuat maksimal 4 partisi dalam 1 hardisk (storage).
- **Partisi Logical** : jenis partisi ini merupakan jenis partisi umum yang hanya dapat digunakan untuk menyimpan data.

Primary dan logical partisi. Pada sebuah hardisk hanya dibolehkan untuk membuat partisi primary sejumlah 4 buah, sedangkan sisanya adalah untuk partisi logical. Syarat sebuah partisi agar bisa dijadikan sebuah partisi filesystem sebuah sistem operasi, partisi tersebut haruslah menggunakan mode partisi primary (partisi utama), sedangkan untuk penyimpanan data lainnya diperbolehkan menggunakan logikal ataupun primary.

3.3 Filesystem Linux

Jika kita sebelumnya telah mengenal tentang konsep hardisk dan partisi, pada bagian ini kita akan belajar tentang filesystem. Filesystem merupakan sebuah sistem yang berfungsi untuk mengatur secara keseluruhan tentang manajemen penyimpanan dalam partisi yang akan digunakan, untuk itu sebelum kita melakukan penyimpanan data pada partisi hardisk kita wajib menentukan filesystem yang akan digunakan untuk partisi tersebut. Jika anda pengguna komputer aktif pasti sudah pernah mengenal tentang istilah NTFS, FAT32, FAT16, dll.

Istilah-istilah tersebut yang kita kenal dengan filesystem, akan tetapi filesystem diatas merupakan filesystem pada sistem operasi yang berbeda, akan tetapi semua tipe filesystem bisa terbaca pada sistem Linux. Linux juga memiliki format / aturan filesystem yang berbeda yaitu : extended filesystem yang kita bisa temui sampai dengan saat ini sudah mencapai pengembangan yang ke 4 (*ext2, ext3, ext4*). Pengaturan filesystem akan bisa kita dapatkan pada semua partisi pada media penyimpanan anda, seperti hardisk (in/eks) maupun flashdrive.

sda				
sda1 (ext4)	sda2 (ext4)	sda3 (ext4)	sda4 (ntfs)	sda5 (swap)
sdb				
sdb1 (ntfs)	sdb2 (fat32)	sdb3 (fat32)	sdb4 (fat32)	sdb5 (fat32)

Pengaturan hardisk / partisi pada sistem linux dapat kita lihat dengan mengetikkan perintah **fdisk**, lengkap dengan parameter (**-l**), dengan akses sebagai superuser.

```
root@ligo:/home/ligo# fdisk -l
```

```
Disk /dev/sda: 250.1 GB, 250059350016 bytes
255 heads, 63 sectors/track, 30401 cylinders, total 488397168 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x0e4f7ee5
```

```
Device Boot      Start         End      Blocks   Id  System
/dev/sda1  *           63    122881184    61440561    7  HPFS/NTFS/exFAT
/dev/sda2             122881246  488396799    182757777    f  W95 Ext'd (LBA)
/dev/sda5             122881248  324265750    100692251+    7  HPFS/NTFS/exFAT
/dev/sda6             482258944  488396799      3068928    82  Linux swap / Solaris
/dev/sda7             324265984  482254847      78994432    83  Linux
```

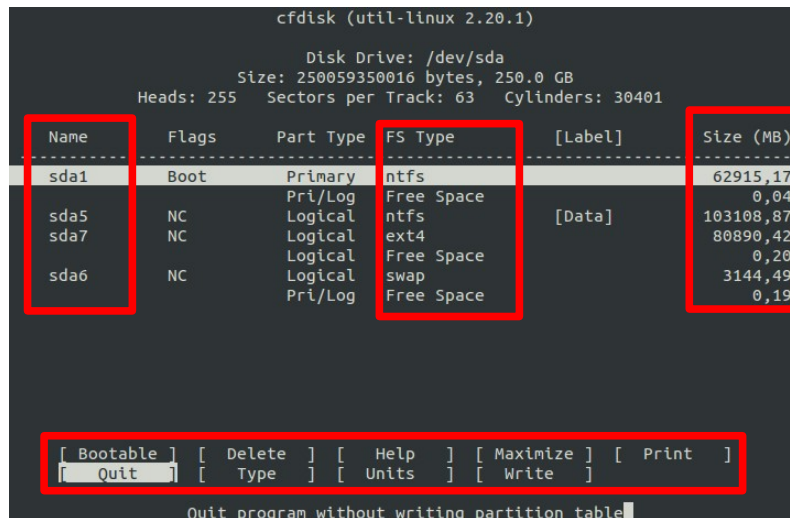
Seperti yang telah ditampilkan pada hasil perintah fdisk -l diatas, masing-masing partisi akan memiliki filesystem, Pengaturan partisi secara keseluruhan tersimpan pada direktori /dev atau device. Jika anda ingin melakukan pengaturan partisi anda bisa menggunakan aplikasi berbasis text seperti fdisk atau cfdisk.

- **Fdisk** : Penggunaan aplikasi pemartisian fdisk dapat anda jalankan menggunakan akses superuser selanjutnya dengan format hardisk (storage) yang akan anda partisi seperti yang ditunjukkan pada gambar dibawah :

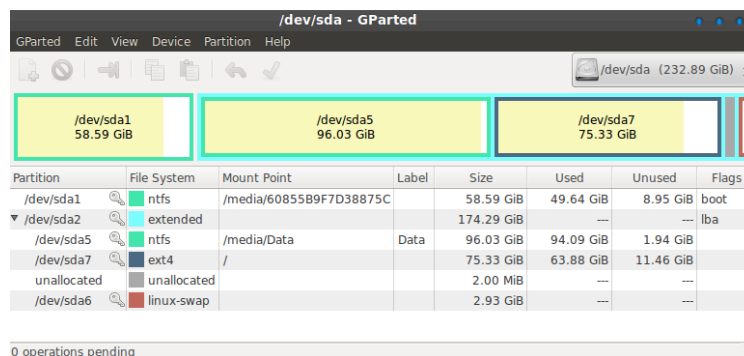
```
root@saronde:/home/saronde_os# fdisk /dev/sda
Command (m for help): m
Command action
  a  toggle a bootable flag
  b  edit bsd disklabel
  c  toggle the dos compatibility flag
  d  delete a partition
  l  list known partition types
  m  print this menu
  n  add a new partition
  o  create a new empty DOS partition table
  p  print the partition table
  q  quit without saving changes
  s  create a new empty Sun disklabel
  t  change a partition's system id
  u  change display/entry units
  v  verify the partition table
  w  write table to disk and exit
Command (m for help): p
Disk /dev/sda: 250.1 GB, 250059350016 bytes
255 heads, 63 sectors/track, 30401 cylinders, total 488397168 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x0e4f7ee5

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1  *           63    122881184    61440561    7  HPFS/NTFS/exFAT
/dev/sda2             122881246    488396799    182757777    f  W95 Ext'd (LBA)
/dev/sda5             122881248    324265750    100692251+    7  HPFS/NTFS/exFAT
/dev/sda6             482258944    488396799     3068928    82  Linux swap / Solaris
/dev/sda7             324265984    482254847     78994432    83  Linux
```

- **cfdisk** : merupakan aplikasi yang memiliki tujuan yang sama dengan fdisk, akan tetapi cfdisk lebih menawarkan kemudahan dalam pemartisian, dengan tersedianya opsi navigasi untuk menghapus, membuat, maupun melakukan format dan write.



- **Gparted** : merupakan aplikasi pemartisian manajer yang dibuat pada lingkungan Grafis, yang merupakan hasil pengembangan aplikasi pemartisian **parted**, yang masih berbasis teks. Dengan tampilan antarmuka yang berbasis grafis, memudahkan pengguna untuk melakukan manajemen pemartisian pada sistem Linux, sehingga lebih meminimalisir kemungkinan hilangnya data karena kesalahan dalam proses pemartisian.



3.4 Memory virtual (swap memory)

Konsep virtual memory merupakan konsep pemartisian pertama yang dikenalkan pada sistem Linux. Tujuannya adalah untuk mempercepat kinerja sistem, serta menurunkan efek terjadinya blank / hang / error.

Untuk itu dibuatlah konsep memory virtual untuk membantu kinerja dari memory fisik (RAM) pada sistem komputer dengan cara mengalokasikan partisi hardisk (media penyimpanan) untuk menjadi media penampung proses sementara, apabila memory fisik sudah tidak bisa

menampung proses, maka kesemua proses tersebut akan ditampung pada memory virtual, pada sistem Linux memory virtual dikenal dengan *swap memory*.

Secara sederhana swap memory dibuat dengan tujuan untuk meringankan beban dari memory fisik, aturan dasar pengalokasian ukuran memory virtual (swap memory) secara umum dapat mengikuti aturan berikut : **swap = 2 x memoryfisik.**

Jadi jika kita memiliki memory fisik dengan ukuran 2 GB, maka ukuran dari swap memory adalah **2 GB x 2 = 4 GB.**

3.5 Pembuatan Partisi hardisk & Konsep Mounting

Jika pada bagian sebelumnya sudah membahas tentang pemartisian dan konsep mounting, pada bagian ini kita akan belajar tahapan pembuatan partisi dan konsep mounting. Sebagai tahapan awal, akan menggunakan sebuah media penyimpanan (storage device), dengan label **/dev/sdb**. Untuk melakukan pemartisian kita akan menggunakan beberapa aplikasi / tools.

- **Fdisk**, merupakan sebuah tool yang berfungsi untuk melakukan pemartisian pada storage device, dengan cara yang bisa dibilang sulit untuk dimengerti oleh pengguna awam untuk itu kita akan belajar menggunakan **fdisk**.

```
root@saronde:~# fdisk /dev/sdb
Command (m for help): m
Command action
.....
n   add a new partition
o   create a new empty DOS partition table
p   print the partition table ~ untuk mencetak tabel partisi
q   quit without saving changes
s   create a new empty Sun disklabel
.....
Command (m for help): p
Disk /dev/sdb: 15.5 GB, 15545139200 bytes
64 heads, 32 sectors/track, 14825 cylinders, total 30361600 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0xc3072e18
```

Device	Boot	Start	End	Blocks	Id	System
--------	------	-------	-----	--------	----	--------

Catatan :

Seperti yang ditunjukkan pada gambar diatas, kita memiliki media penyimpanan dengan label /dev/sdb dengan ukuran 15.5 GB.

Selanjutnya kita akan membuat partisi dengan memilih opsi “n”.

```
Command (m for help): n
Partition type:
  p  primary (0 primary, 0 extended, 4 free)
  e  extended
Select (default p):
```

Seperti yang telah ditampilkan pada hasil output diatas, kita memiliki dua buah opsi yaitu p (primary) dan e (extended / logical). Seperti yang ditunjukkan pada output diatas, kita hanya diizinkan untuk membuat 4 partisi primary. Selanjutnya kita pilih opsi “p”

```
Select (default p): p
Partition number (1-4, default 1): 1
First sector (2048-30361599, default 2048): 2048
Last sector, +sectors or +size{K,M,G} (2048-30361599, default 30361599): 30361599
```

Seperti yang ditampilkan diatas, kita akan membuat sebuah partisi primary dengan number partisi 1, dengan sektor awal 2048 dan sektor akhirnya adalah 30361599. selanjutnya lakukan perubahan pada hardisk dengan menekan opsi “w” untuk melakukan penulisan perubahan ke hardisk.

```
Command (m for help): w
The partition table has been altered!
Calling ioctl() to re-read partition table.
```

Selanjutnya masuk kembali ke pengaturan partisi fdisk dengan perintah fdisk /dev/sdb, selanjutnya lakukan pengecekan kembali pada tabel partisi dengan mengetikkan opsi “p”

```
Command (m for help): p
Disk /dev/sdb: 15.5 GB, 15545139200 bytes
64 heads, 32 sectors/track, 14825 cylinders, total 30361600 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0xc3072e18
```

Device	Boot	Start	End	Blocks	Id	System
/dev/sdb1		2048	30361599	15179776	83	Linux

Seperti yang ditunjukkan pada output diatas satu partisi telah berhasil dibuat selanjutnya, kita akan membuat filesystem untuk partisi tersebut dengan perintah **mkfs (make filesystem)**. Filesystem yang akan kita gunakan adalah *extended 4 (ext4)*.

```
root@saronde:~# mkfs.ext4 /dev/sdb1
mke2fs 1.42 (29-Nov-2011)
Sistem berkas label=
OS type: Linux
Ukuran blok=4096 (log=2)
Ukuran pecahan=4096 (log=2)
.....
Cadangan superblok disimpan di blok:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208
Allocating group tables: selesai
Menulis tabel inode: selesai
Membuat jurnal (32768 blok): selesai
Menulis superblok dan informasi akuntansi sistem berkas:
```

Sedangkan untuk membuat filesystem FAT32, dapat dengan mengetikkan perintah berikut :

```
root@saronde:~# mkfs.vfat /dev/sdb1
mkfs.vfat 3.0.12 (29 Oct 2011)
```

Catatan :

mkfs menyediakan beberapa parameter fungsi sesuai dengan penggunaan filesystem, beberapa fungsinya adalah sebagai berikut :

```
root@saronde:~# mkfs ..(tab)
mkfs          mkfs.cramfs  mkfs.ext4    mkfs.msdos   mkfs.vfat
mkfs.bfs      mkfs.ext2    mkfs.ext4dev mkfs.ntfs
mkfs.btrfs    mkfs.ext3    mkfs.minix   mkfs.reiserfs
```

Selain menggunakan mkfs dengan format diatas, kita dapat dengan menggunakan parameter **-t (type)**, dilanjutkan dengan tipe partisi (ntfs, fat32, ext3, ext4, dll). Contoh penggunaan perintah mkfs dengan parameter adalah sebagai berikut :

```
root@saronde:~# mkfs -t vfat /dev/sdb1
root@saronde:~# mkfs -t ntfs /dev/sdb1
root@saronde:~# mkfs -t ext4 /dev/sdb1
root@saronde:~# mkfs -t brtfs /dev/sdb1
root@saronde:~# mkfs -t msdos /dev/sdb1
```

Setelah pembuatan filesystem partisi selesai, selanjutnya lakukan mounting terhadap partisi tersebut agar bisa digunakan untuk menyimpan data :

```
root@saronde:~# mkdir /media/flashdisc
root@saronde:~# mount /dev/sdb1 /media/flashdisc/
root@saronde:~# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda7        76G   64G   7,8G  90% /
.....
/dev/sda5        97G   95G   2,0G  98% /media/Data
/dev/sda1        59G   50G   9,0G  85% /media/60855B9F7D38875C
/dev/sdb1       15G   8,0K   15G   1% /media/flashdisc
```

Catatan :

1. **mkdir /media/flashdisc** : merupakan sebuah perintah yang berfungsi untuk membuat direktori mounting point yang nantinya akan digunakan untuk menampung media penyimpanan agar bisa digunakan dalam proses penyimpanan data
2. **mount /dev/sdb1 /media/mediaflashdisc** : merupakan perintah mounting dimana untuk melakukan mounting, untuk melakukan mounting kita membutuhkan dua komponen yaitu media penyimpanan (partisi) yang sudah memiliki filesystem, dan direktori mounting.

Keterangan :

Mounting bertujuan agar membuat media penyimpanan dapat digunakan untuk melakukan penyimpanan karena secara default linux mengenali partisi (storage device) dalam bentuk file yang tersimpan pada direktori device (/dev)

3. **df -h** : merupakan perintah yang berfungsi untuk melihat format partisi filesystem yang sudah dimounting, sedangkan parameter -h merupakan, parameter yang berfungsi untuk menampilkan informasi tentang partisi, filesystem dan ukuran yang mudah dibaca dan dimengerti.

Jika sebelumnya kita telah belajar melakukan mounting agar partisi dapat terbaca dan digunakan, apabila partisi sudah tidak ingin digunakan lagi partisi tersebut perlu kita lepaskan akses mounting (kait), atau dikenal dengan istilah **umount**.

```
saronde_os@saronde:~$ df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda7        76G   64G   7,8G  90% /
.....
/dev/sdb1        15G   1,5M   15G   1% /media/1FD2-2D78
saronde_os@saronde:~$ umount /media/1FD2-2D78
saronde_os@saronde:~$ df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda7        76G   64G   7,8G  90% /
.....
```

Proses *umounting* cukup sederhana, cukup dengan mengetikkan perintah umount dilanjutkan dengan direktori atau titik mounting seperti yang telah ditampilkan pada output diatas.

- **Cfdisk**, merupakan sebuah aplikasi yang berfungsi untuk melakukan pemartisian pada hardisk (storage device). **Cfdisk** menggunakan interface pengaturan yang mudah untuk digunakan jika dibandingkan dengan **fdisk**. Untuk melakukan pemartisian pada sebuah media penyimpanan kita hanya perlu mengetikkan perintah berikut :

```
root@saronde:~# cfdisk /dev/sdb1
```

```
cfdisk (util-linux 2.20.1)
Disk Drive: /dev/sdb1
Size: 15544090624 bytes, 15.5 GB
Heads: 64 Sectors per Track: 32 Cylinders: 14824
-----
Name      Flags      Part Type  FS Type    [Label]    Size (MB)
-----
Pri/Log   Free Space 15544,10

[ Help ] [ New ] [ Print ] [ Quit ] [ Units ]
[ Write ]

Create new partition from free space
```

Untuk melakukan pembuatan partisi kita bisa memilih opsi “new”. Selanjutnya lakukan pemilihan jenis / tipe partisi yang akan dibuat (primary / logical). Pada tahapan ini pilih opsi primary.

```
[ Primary ] [ Logical ] [ Cancel ]

Create a new primary partition
```

Selanjutnya tentukan ukuran partisi primary yang akan dibuat, alokasikan 10000 MB (10 GB), selanjutnya tekan [Enter]

```
Size (in MB): 10000
```

Selanjutnya tentukan mode / urutan partisi (Beginning / End), karena partisi yang akan kita buat adalah primary partisi, secara default pilih [Beginning].

```
[ Beginning ] [ End ] [ Cancel ]
```

Selanjutnya secara otomatis akan dibuatkan 1 buah partisi primary dengan, ukuran 10 GB. Seperti yang ditunjukkan pada gambar dibawah :

```
cfdisk (util-linux 2.20.1)

Disk Drive: /dev/sdb1
Size: 15544090624 bytes, 15.5 GB
Heads: 64 Sectors per Track: 32 Cylinders: 14824

Name      Flags      Part Type  FS Type    [Label]    Size (MB)
-----
sdb1p1    Primary   Linux     10000,27
          Prio/Log  Free Space 5543,83
```

Selanjutnya kita buat partisi selanjutnya dengan ukuran 5000 MB (5 GB).

```
cfdisk (util-linux 2.20.1)

Disk Drive: /dev/sdb1
Size: 15544090624 bytes, 15.5 GB
Heads: 64 Sectors per Track: 32 Cylinders: 14824

Name      Flags      Part Type  FS Type    [Label]    Size (MB)
-----
sdb1p1    Primary   Linux     10000,27
sdb1p2    Primary   Linux     5543,83
```

Selanjutnya lakukan penulisan dengan memilih opsi write, agar pengaturan partisi dapat diterapkan dalam sistem.

```
cfdisk (util-linux 2.20.1)

Disk Drive: /dev/sdb1
Size: 15544090624 bytes, 15.5 GB
Heads: 64 Sectors per Track: 32 Cylinders: 14824

Name      Flags      Part Type  FS Type    [Label]    Size (MB)
-----
sdb1p1    Primary   Linux     10000,27
sdb1p2    Primary   Linux     5543,83

Are you sure you want to write the partition table to disk? (yes or no): █

Warning!! This may destroy data on your disk!
```

Selanjutnya pilih opsi “yes” apabila sudah yakin dengan pengaturan partisi yang telah dilakukan. *Proses ini akan menghapus keseluruhan data dalam partisi hardisk*

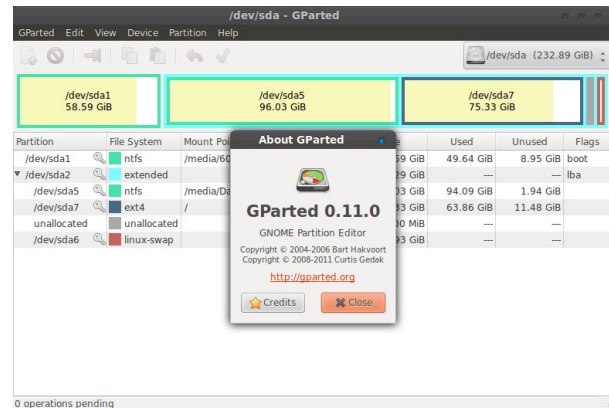
Setelah proses pembuatan partisi, filesystem, dan mounting point telah selesai dibuat selanjutnya lakukan pengecekan error terhadap partisi dan filesystem dengan menggunakan perintah **fsck (filesystem check)**.

```

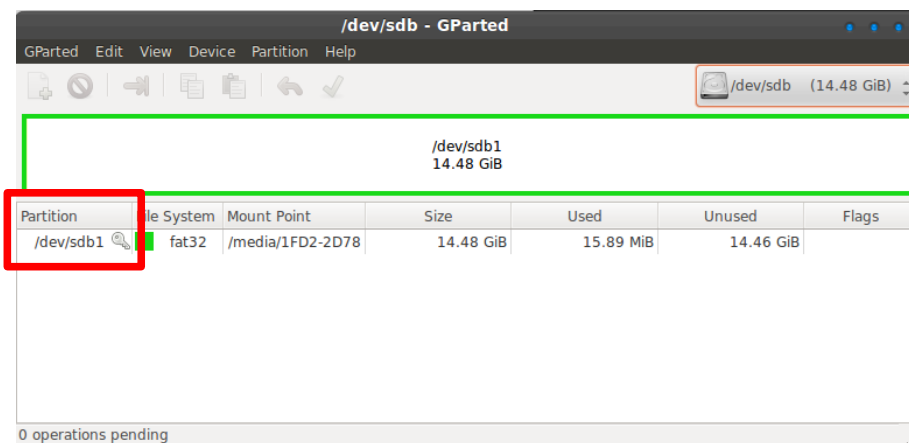
root@saronde:~# fsck /dev/sdb1
fsck from util-linux 2.20.1
dosfsck 3.0.12, 29 Oct 2011, FAT32, LFN
/dev/sdb1: 0 files, 1/1895618 clusters

```

- Gparted**, merupakan sebuah aplikasi manajemen partisi yang berjalan pada lingkungan grafik, **gparted** merupakan hasil pengembangan dari aplikasi **parted** yang masih berbasis text. Dengan Gparted kita akan lebih mudah melakukan pemartisian karena menggunakan mode grafis, sehingga dapat mengurangi resiko kesalahan dalam pengaturan, dan pembuatan partisi.

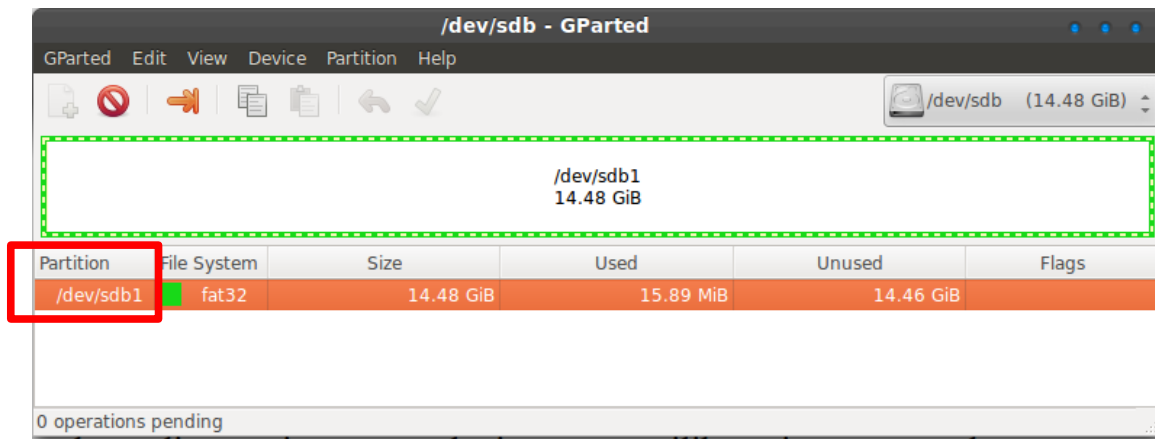


Selanjutnya siapkan media penyimpanan yang akan dibuatkan partisinya seperti yang ditunjukkan pada gambar berikut :

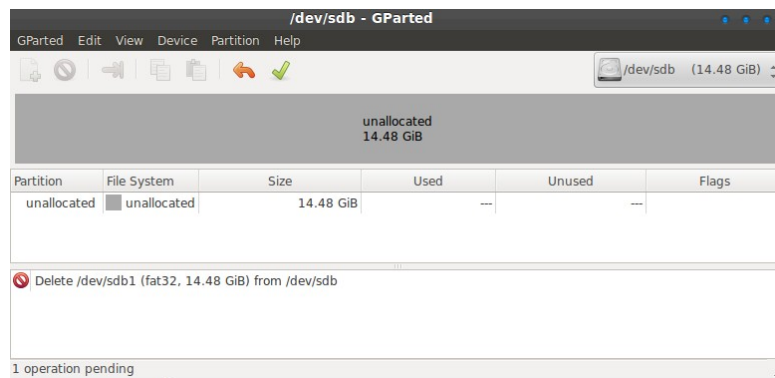


Lakukan umount terhadap partisi dengan cara memilih partisi yang ingin di umount, selanjutnya pilih menu “Partition”, selanjutnya pilih opsi “Umount”, atau bisa melakukan klik kanan pada media penyimpanan selanjutnya memilih opsi umount pada menu pop-up.

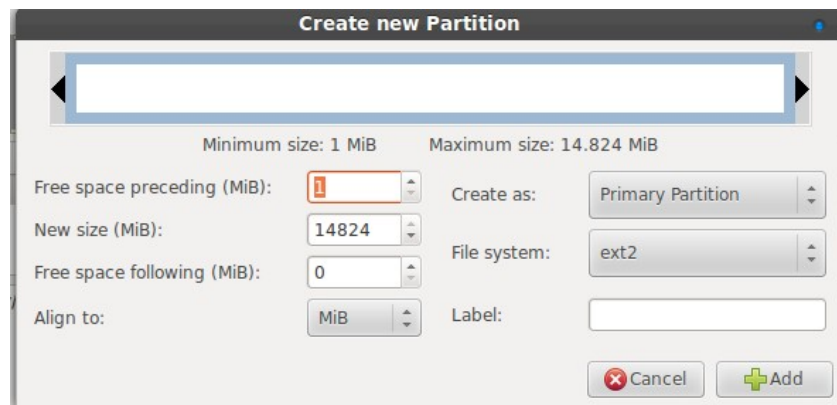
Proses umount akan berhasil dengan hilangnya tanda “kunci” partisi pada media penyimpanan seperti yang ditunjukkan pada gambar berikut :



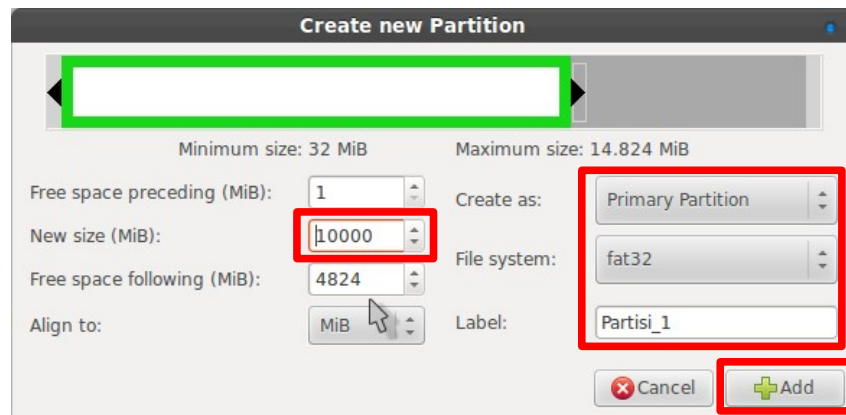
Selanjutnya terlebih dahulu hapus seluruh ruang partisi yang sebelumnya sudah dibuat, dengan memilih menu “Partition” selanjutnya pilih “Delete”, atau bisa dengan menekan tombol “Delete” melalui keyboard. Seperti yang ditunjukkan pada gambar disamping :



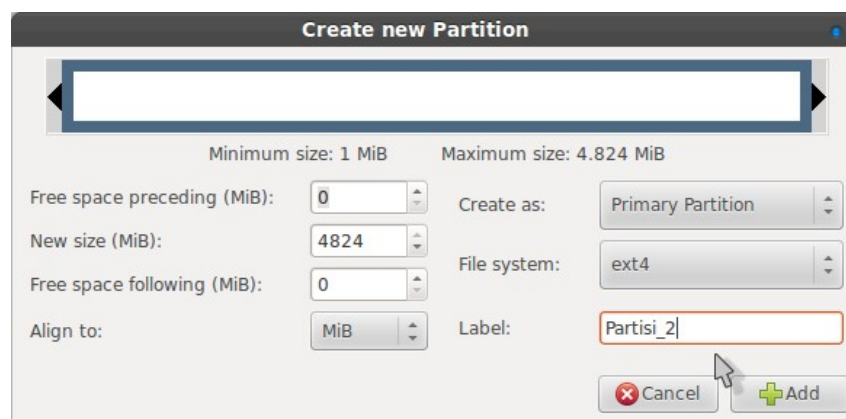
Selanjutnya lakukan pembuatan partisi dengan memilih menu “Partition”, selanjutnya memilih opsi “New”. Selanjutnya akan ditampilkan pop-up window seperti yang ditunjukkan pada gambar berikut :



Pada window diatas, kita bisa melakukan pengaturan filesystem, jenis partisi, ukuran partisi dan label partisi dengan cara yang mudah. Kali ini kita akan membuat sebuah partisi dengan ukuran 10 GB dengan filesystem FAT32 dan sisanya menggunakan filesystem Ext4.



Ruang Partisi 1 : Filesystem (Fat32), Label (Partisi_1), Tipe Partisi (Primary)



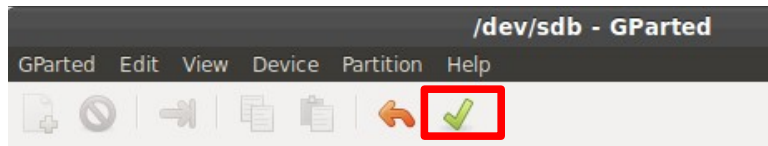
Ruang Partisi 2 : Filesystem (ext4), Label (Partisi_2), Tipe Partisi (Primary)

Hasil pengaturan partisi pada hardisk kita saat ini adalah sebagai berikut :

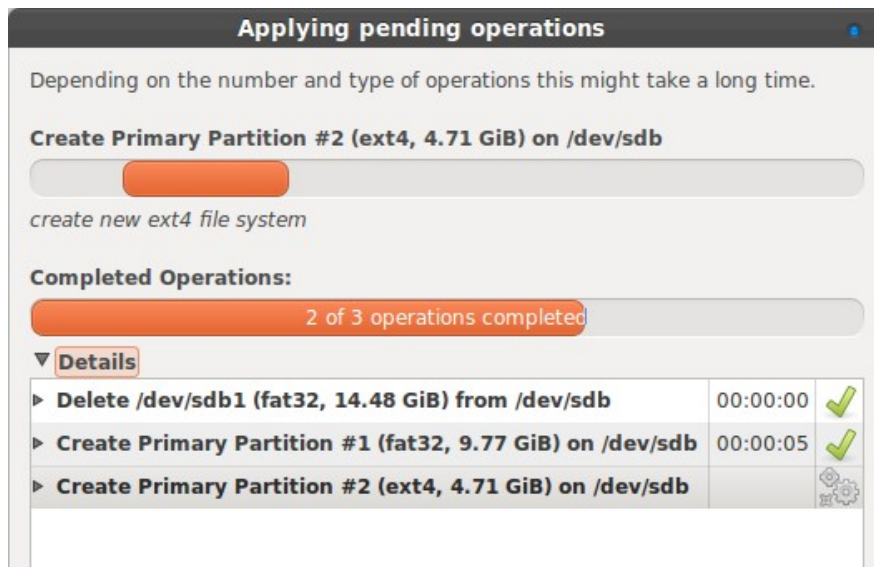
Partition	File System	Label	Size	Used	Unused	Flags
New Partition #1	fat32	Partisi_1	9.77 GiB	---	---	
New Partition #2	ext4	Partisi_2	4.71 GiB	---	---	

Selanjutnya terapkan pengaturan / perubahan yang telah dilakukan diatas pada hardisk kita dengan melakukan klik pada menu “Edit”, Selanjutnya pilih Opsi “**Apply All Operations**”, apabila anda ingin melakukan perubahan partisi / mengalami kesalahan dalam proses pemartisian anda bisa melakukan undo, dengan melakukan klik pad menu “Edit” selanjutnya pilih opsi “**Undo Last Operation**” untuk mengembalikan ke operasi terakhir, sedangkan opsi “**Clear all Operations**” untuk menghapus semua pengaturan sebelumnya.

Apabila sudah yakin dengan pengaturan yang telah dilakukan pilih opsi “**Apply all Operations**”, atau bisa dengan melakukan klik pada icon berikut :



Selanjutnya proses pembuatan partisi yang telah dilakukan akan berjalan seperti yang ditunjukkan pada gambar dibawah :



Partisi yang telah dibuat sebelumnya dapat kita lihat dengan menggunakan perintah **df**, seperti yang ditunjukkan pada gambar berikut :

```
saronde_os@saronde:~$ df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda7       76G   64G   7,8G  90% /
.....
/dev/sdb1       9,8G   8,0K   9,8G   1% /media/Partisi_1
/dev/sdb2       4,7G  199M   4,3G   5% /media/Partisi_2
```

Pembuatan partisi dengan menggunakan aplikasi gparted telah berhasil, seperti yang ditunjukkan pada hasil perintah diatas, dan proses mounting telah berhasil dilakukan.

- **Automounting**

Linux mengenal konsep *mounting*, dengan tujuan agar sebuah media penyimpanan data digunakan dan dimanfaatkan untuk melakukan proses penyimpanan, seiring dengan berkembangnya distribusi Linux, Linux sudah mengenalkan konsep otomatisasi dari konsep *mounting* yang diberi nama "*Automounting*".

Dikarenakan konsep *mounting* yang telah dijelaskan diatas sulit dipahami oleh pengguna komputer awam, hal tersebut tentu merepotkan bagi pengguna linux baru, untuk itu diperkenalkan konsep *automounting* atau bisa disebut *mounting otomatis*, jadi cara / proses kerjanya sistem kernel linux akan melakukan proses *mounting* secara otomatis ketika kernel mendeteksi perangkat penyimpanan yang terkoneksi dengan komputer, sehingga perangkat penyimpanan tersebut sudah bisa langsung digunakan. Konsep tersebut dikenal dengan istilah *plug & play* pada sistem operasi selain Linux.

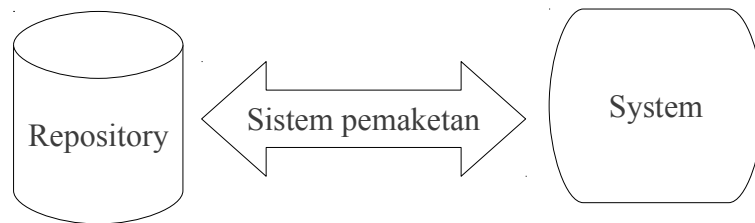
Bagian IV

Manajemen Pemaketan

Manajemen Pemaketan merupakan sebuah metode yang digunakan untuk melakukan proses instalasi aplikasi dalam sistem, manajemen paket merupakan hal yang terpenting dalam sistem linux, karena tanpa adanya manajemen paket proses install, remove, update, dan upgrade tidak bisa dilakukan secara efisien.

Tanpa adanya sistem pemaketan sebuah aplikasi yang hendak di install dalam sistem, pengguna haruslah mendownload source code melalui internet, selanjutnya menyediakan pustaka yang dibutuhkan selanjutnya melakukan kompilasi agar source code aplikasi tersebut dapat bekerja.

Hal tersebut dinilai tidak efisien untuk itu didesainlah sebuah sistem pemaketan yang menjadi perantara antar aplikasi dan sistem sehingga proses instalasi, update ataupun upgrade menjadi jauh lebih efisien dan cepat, sedangkan semua paket aplikasi disimpan dalam sistem repository. Berikut adalah alur yang dapat digambarkan dalam proses pengerjaan sistem pemaketan.



Repository merupakan sebuah server terpusat yang melakukan penyimpanan paket, library dan dependensi yang dibutuhkan, untuk mengambil paket pada repository tersebut kita perlu terhubung menggunakan sistem pemaketan.

Dengan didesainnya sebuah sistem / manajemen pameketan dapat mempermudah pengguna untuk melakukan proses instalasi aplikasi, karena tidak perlu menyediakan dependensi secara manual seperti pada teknik instalasi tradisional (unix), dengan adanya manajemen dependensi akan diinstall secara otomatis apabila dibutuhkan oleh sebuah aplikasi.

4.1 Sistem Paket Distribusi

Seperti yang sudah dijelaskan pada bagian sebelumnya, Linux terbagi menjadi banyak distribusi, salah satu distribusi inti adalah Debian, & Redhat, dll. Pada bagian ini akan dijelaskan sistem pemaketan yang digunakan oleh dua distribusi diatas, selain dua distribusi diatas terdapat 1 distribusi yang masih menggunakan metode pemaketan tradisional.

Banyak distribusi linux yang dikembangkan saat ini yang sudah menggunakan manajemen pemaketan, kesemua pengembangan tersebut mengacu pada dua distribusi diatas (Debian & Redhat). Berikut adalah deskripsi detail tentang sistem pemaketan yang digunakan oleh kedua distribusi.

Sistem Pemaketan	Distribusi Linux
Debian (*.deb)	Debian, Ubuntu, dll
Redhat (*.rpm)	Redhat, Fedora, Centos, OpenSUSE

4.2 Sistem Server Repository

Jika sebelumnya sudah dibahas tentang sedikit definisi repository, bentuk / model server repository berbentuk bagaimana sebuah komputer yang menyediakan ratusan paket (aplikasi, pustaka dan dependensi), server repository tersebut terkoneksi melalui media internet / intranet, untuk menggunakan repository tersebut kita memerlukan alamat repository untuk dituju, berikut adalah contoh / format alamat repository untuk distribusi linux Ubuntu :

```
deb http://repo.ugm.ac.id/ubuntu/ lucid main restricted universe multiverse
```

Keterangan :

Format diatas dapat dijelaskan pada bagian yaitu :

Bagian 1 : pendefinisian paket yang ditandai dengan deb

Bagian 2 : pendefinisian alamat server repository dengan format domain

Bagian 3 : key-code dari distribusi yang digunakan

Bagian 4 : Kode & status dari paket, dapat dijelaskan pada format berikut :

- **Main** adalah paket yang didukung secara Resmi
- **Restricted** adalah paket yang didukung tapi tidak sepenuhnya dalam free license
- **Universe** adalah paket yang dikelola oleh komunitas (bukan software yang didukung dengan resmi)
- **Multiverse** adalah paket yang tidak Free .

Konfigurasi Repository

Untuk melakukan konfigurasi alamat repository, agar dapat mengarah ke server repository yang diinginkan dapat melakukan proses pengeditan pada file source.list yang terletak pada direktori /etc/apt/ dengan menggunakan text editor (nano, mcedit, atau vi).

```
root@saronde:~# nano /etc/apt/sources.list
```

```

GNU nano 2.2.6 File: /etc/apt/sources.list

deb http://repo.ugm.ac.id/ubuntu/ precise main restricted universe multiverse
deb http://repo.ugm.ac.id/ubuntu/ precise-updates main restricted universe mult$
deb http://repo.ugm.ac.id/ubuntu/ precise-security main restricted universe mu$

deb http://www.remastersys.com/ubuntu precise main

# deb http://repo.ugm.ac.id/ubuntu/ precise main restricted universe multiverse
# deb http://repo.ugm.ac.id/ubuntu/ precise-updates main restricted universe mu$
# deb http://repo.ugm.ac.id/ubuntu/ precise-security main restricted universe m$

```

4.3 Alat Manajemen Pemaketan

Alat untuk melakukan proses pemaketan aplikasi pada sistem linux terbagi menjadi dua tools yang dapat dikategorikan menjadi Tools High Level dan Low Level, masing-masing sistem pemaketan dari distribusi memiliki tools-tools tersebut, pendefinisian tools pemaketan dapat ditunjukkan pada gambar berikut :

Distribusi	Low Level Tool	High Level Tool
Debian	dpkg	apt-get
Redhat	rpm	yum

Masing-masing tools memiliki fungsi yang sama akan tetapi memiliki kinerja / metode kerja yang berbeda, berikut adalah penggunaan dari masing-masing tools untuk fungsi yang berbeda – beda.

Pencarian paket dari Repository

Jika sebelumnya kita sudah terkoneksi dengan repository, maka sebelum melakukan proses instalasi aplikasi, kita membutuhkan nama aplikasi yang spesifik untuk itu dibutuhkan proses pencaharian otomatis menggunakan perintah berikut :

Debian	apt-cache search nama_aplikasi
Redhat	yum search nama_aplikasi

Contoh :

Debian

```
root@saronde:~# apt-cache search nano
nano - small, friendly text editor inspired by Pico
alpine-pico - Simple text editor from Alpine, a text-based email client
```

Redhat

```
root@saronde:~# yum search nano
nano - small, friendly text editor inspired by Pico
alpine-pico - Simple text editor from Alpine, a text-based email client
```

Instalasi Paket dari repository

Setelah proses pencarian usai selanjutnya kita akan, melakukan instalasi paket / aplikasi yang akan diinstall kedalam sistem untuk itu bisa dengan menggunakan perintah berikut :

Debian	apt-get install nama_aplikasi
Redhat	yum install nama_aplikasi

Contoh :

Debian

```
root@saronde:~# apt-get install nano
Reading package lists... Done
Building dependency tree
Reading state information... Done
.....
```

Fedora

```
root@saronde:~# yum install nano
Reading package lists...
```

Instalasi paket dari file paket

Jika sebelumnya proses instalasi yang dijelaskan melalui repository, pada bagian ini akan dijelaskan proses instalasi low leve aplikasi melalui file package yang sudah didownload, file tersebut secara spesifik dikenali dari penggunaan ekstensi, untuk debian (*.deb) dan redhat (*.rpm). Sebelum melakukan proses eksekusi ini, pastikan semua dependensi / library dari paket yang akan diinstall telah terpenuhi, jika tidak anda bisa melakukan proses instalasi dalam kondisi sedang terhubung ke repository server (internet / intranet). Penggunaan perintahnya untuk melakukan proses instalasi pada paket adalah sebagai berikut :

Debian	dpkg --install nama_aplikasi
Redhat	rpm -i nama_aplikasi

Contoh :

Debian

```
root@saronde:# dpkg --install virtualbox-4.3_4.3.4.deb
```

Redhat

```
root@saronde:# rpm -i virtualbox-4.3_4.3.4.deb
```

Catatan :

Pastikan semua dependensi sudah terinstall, jika tidak proses eksekusi akan dihentikan atau akan dihentikan dengan meninggalkan pesan error dependensi.

Penghapusan Paket

Untuk melakukan proses penghapusan paket aplikasi, sangat mudah dibandingkan proses instalasi karena, proses penghapusan tidak membutuhkan akses ke repository. Berikut adalah perintah yang dapat digunakan :

Debian	apt-get remove nama_aplikasi
Redhat	yum erase nama_aplikasi

Contoh :

Debian

```
root@saronde:# apt-get remove nano
```

Redhat

```
root@saronde:# yum erase nano
```

Update Paket dari Repository

Proses update / pembaharuan daftar repository pada sistem sangat diperlukan, karena dengan adanya proses ini, sistem akan dapat membuat sebuah daftar paket, dependensi dan library yang tersedia pada repository, sehingga kita dapat melakukan pencaharian dan proses instalasi.

Proses update haruslah dilakukan secara berkala, untuk melakukan pengecekan perubahan paket pada repository, sehingga daftar (list) aplikasi, library dan dependensi dapat diperbaharui secara berkala untuk melakukan update anda perlu terhubung ke server repository melalui internet / intranet.

Debian	apt-get update
Redhat	yum update

Daftar paket aplikasi, library dan dependensi

Merupakan proses yang sangat penting, dalam manajemen pemaketan, karena proses ini akan bisa menampilkan informasi dari seluruh paket yang telah terinstall dalam sistem.

Debian	dpkg --list
Redhat	rpm -qa

Menampilkan informasi tentang aplikasi

Jika sebelumnya kita dapat menggunakan sebuah perintah untuk melihat status paket yang terinstall dalam sistem, pada bagian ini kita akan melihat status keterangan tentang paket secara spesifik (lengkap).

Debian	dpkg - - status nama_aplikasi
Redhat	rpm -q nama_aplikasi

Contoh :

Debian

```
root@saronde:~# dpkg --status nano
Package: nano
Status: install ok installed
Priority: important
Section: editors
Installed-Size: 584
.....
```

Bagian V

BASH

(Bourne Again Shell)

Jika kita berbicara tentang antarmuka baris perintah (line command), pastilah akan banyak membahas tentang shell. Shell merupakan sebuah program yang menerima inputan perintah melalui keyboard, untuk selanjutnya perintah tersebut akan dieksekusi oleh sistem. Keseluruhan Distribusi sistem operasi berbasis linux memiliki sebuah shell program yang dikembangkan dibawah lisensi GNU Project (pengembangan publik), pengembangan tersebut adalah *bash*.

Bash merupakan sebuah akronim dari Bourne Again Shell, bash dikembangkan dari shell sistem unix yang dikembangkan pertama kali oleh Steven Bourne. Untuk berinteraksi dengan program bash, apabila kita menggunakan linux desktop kita dapat menggunakan aplikasi emulator yang disebut Terminal Emulator, terminal emulator yang akan mengkomunikasikan pengguna linux desktop dengan sistem secara keseluruhan. Hal diatas berbanding terbalik apabila kita menggunakan linux server, kita tidak membutuhkan Terminal Emulator lagi, karena interface linux server secara default sudah menggunakan bash.

Pengenalan Shell

Jika ingin belajar sesuatu pastilah kita harus secara langsung berinteraksi tentang hal-hal tersebut, untuk itu silahkan buka terminal emulator anda, masing-masing distribusi linux memiliki emulator yang berbeda-beda.

```
saronde_os@saronde:~$
```

Jika sejenak kita melihat prompt console diatas, struktur prompt dapat kita pecah menjadi beberapa bagian inti, yaitu :

```
saronde_os@saronde:~$
```

Keterangan

- **Bagian 1** : Menunjukkan pengguna yang sedang menggunakan user diatas. Yang berarti pengguna yang sedang aktif / digunakan sekarang adalah saronde_os.
- **Bagian 2** : Menunjukkan hostname / group yang sedang digunakan pada prompt tersebut. Yang berarti bahwa hostname / group yang sedang aktif / digunakan saat ini adalah saronde.
- **Bagian 3** : Lokasi path direktori yang ditempati prompt. Yang berarti bahwa lokasi path yang menggunakan simbol dash ~ adalah menandakan, lokasi / path prompt saat ini berada di home direktori dari user yang sedang digunakan.

- **Bagian 4** : Hak akses dari pengguna yang aktif. Yang berarti bahwa hak akses dalam pengguna dalam sistem linux terbagi menjadi dua : userbiasa (\$) dan superuser (#). Simbolisasi untuk user biasa adalah string (\$), sedangkan superuser adalah hash (#).

Perintah History

Sudah cukup untuk berkenalan dengan struktur prompt bash, bash memiliki sistem perekaman perintah, kesemua perintah yang pernah disimpan akan di simpan secara otomatis dalam sistem histroy, untuk melihat struktur command yang sebelumnya sudah pernah diketikkan kita bisa menekan tombol pada keyboard (arrow-up (panah atas) atau arrow-down (panah bawah)). Sedangkan perintah untuk melihat struktur / daftar perintah kita bisa menggunakan perintah *history*.

```
saronde_os@saronde:~$ history
1771 df -h
1772 man parted
1773 parted -l
1774 sudo parted -l
1775 clear
1776 man parted
.....
```

Perintah bash sederhana

Jika sebelumnya kita sudah mencoba beberapa perintah bash seperti history dan beberapa navigasi lainnya, pada bagian ini kita akan mencoba beberapa perintah sederhana, berikut adalah beberapa perintah bash :

- **date** – merupakan perintah yang berfungsi untuk menampilkan informasi tentang tanggal secara lengkap, yang meliputi hari, bulan tanggal, jam dan tahun.

```
saronde_os@saronde:~$ date
Kam Jan 30 17:49:13 CIT 2014
```

- **free** – merupakan perintah yang berfungsi untuk menampilkan informasi tentang penggunaan memory RAM dalam sistem, informasi yang ditampilkan berupa, total kapasitas memory fisik dan swap, penggunaan, kapasitas bebas, jumlah buffer dan cache memory.

```
saronde_os@saronde:~$ free
```

	total	used	free	shared	buffers	cached
Mem:	1989948	1646384	343564	0	267408	871152
-/+ buffers/cache:		507824	1482124			
Swap:	4063896	0	4063896			

- **cal** – merupakan perintah yang berfungsi untuk menampilkan kalender dengan format penanggalan yang lengkap.

```
saronde_os@saronde:~$ cal
```

Januari 2014

Mi	Se	Se	Ra	Ka	Ju	Sa
		1	2	3	4	
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	

- **df** – berfungsi untuk menampilkan informasi secara detail tentang media penyimpanan, informasi yang ditampilkan meliputi partisi, filesystem, ukuran block, kapasitas media penyimpanan (partisi), persentasi penggunaan media penyimpanan (partisi), dan informasi tentang mounting point media penyimpanan yang dilakukan.

```
saronde_os@saronde:~$ df
```

Filesystem	1K-blocks	Used	Available	Use%	Mounted on
/dev/sda7	78867896	66844572	8073604	90%	/
/dev/sda1	61440560	52055364	9385196	85%	/media/60855B9F7D38875C
/dev/sda5	100692244	98662556	2029688	98%	/media/Data

5.1. Navigasi Shell

Pada bagian ini kita akan membahas tentang proses navigasi shell, kenapa memerlukan navigasi...?. Ya karena sebelumnya kita sudah belajar tentang lokasi path, untuk itu kita membutuhkan layanan navigasi. Bagaimana cara melakukan navigasi ? Dalam hal ini bagaimana cara berpindah dari satu path (lokasi) direktori ke lokasi direktori yang lain.

Linux memiliki struktur filesystem berbentuk hirarki, dan konsep pohon terbalik dimana terdapat satu direktori inti, yang kita kenal dengan direktori *root*, dan dibawah direktori *root* tersebut terdapat banyak direktori turunannya. Navigasi hanya mengenal dua tipe / karakter file yang terdiri dari direktori dan file. proses navigasi direktori, membutuhkan beberapa perintah yaitu `pwd`, `cd` dan `ls`.

- **pwd (print working direktori)**, merupakan perintah yang berfungsi untuk menampilkan letak (path) direktori secara struktural bergantung pada posisi (path) dari sebuah prompt.

```
saronde_os@saronde:~$ pwd
/home/saronde_os
```

- **cd (change directory)**, perintah yang berfungsi untuk melakukan perpindahan (masuk / keluar) antar direktori.

```
saronde_os@saronde:~$ pwd                (baris pertama)
/home/saronde_os
saronde_os@saronde:~$ cd Desktop/         (baris kedua)
saronde_os@saronde:~/Desktop$ pwd
/home/saronde_os/Desktop
saronde_os@saronde:~/Desktop$ cd ..      (baris ketiga)
saronde_os@saronde:~$ pwd
/home/saronde_os
```

Keterangan :

Seperti yang telah dijelaskan diatas, perintah pada baris pertama mencetak informasi tentang letak, selanjutnya pada baris perintah yang kedua, ketikkan perintah `cd` (change directory) untuk masuk ke direktori desktop, selanjutnya kita mengetikkan perintah `cd ..`, untuk keluar dari direktori desktop.

Untuk masuk ke direktori kita bisa mengetikkan perintah dengan format :

```
saronde_os@saronde:~$ cd [spasi] direktori_tujuan
```

Atau jika kita memiliki kondisi path direktori yang berbeda, dimana didalam direktori tersebut terdapat direktori tujuan.

Contoh :

Terdapat direktori dengan struktur sebagai berikut :

direktoriA -> direktoriB -> direktoriC

Jika posisi kita berada pada direktoriA, maka untuk masuk ke direktoriC, kita perlu mengetikkan perintah berikut :

```
saronde_os@saronde:~$ cd direktoriA/direktoriB/direktoriC
saronde_os@saronde:~/direktoriA/direktoriB/direktoriC$ pwd
/home/saronde_os/direktoriA/direktoriB/direktoriC
```

Sedangkan untuk keluar dari direktori kita bisa mengetikkan perintah dengan format berikut :

```
saronde_os@saronde:~$ cd ..
```

Apabila kita berkeinginan untuk keluar dari 1 direktori, jika lebih kita bisa menghubungkan tanda dua titik, dengan tanda slash, dilanjutkan dengan dua titik, dst.

```
saronde_os@saronde:~$ cd ../..
```

- **ls (list directory content)**, merupakan perintah yang berfungsi untuk menampilkan informasi tentang daftar / isi dari sebuah direktori secara detail dan lengkap.

```
saronde_os@saronde:~/Desktop$ ls
Linux      Pembelajaran      Project (urgent)
```

Perintah *ls*, juga bisa digunakan untuk melihat isi direktori yang memiliki lokasi yang berbeda dari path prompt.

```
saronde_os@saronde:~/Desktop$ ls /usr/share
aclocal          iptables
aclocal-1.11     java
.....
```

Masing-masing perintah memiliki parameter atau fungsi ekstra yang bisa kita gunakan untuk menampilkan informasi secara detail dan lengkap. Contoh penggunaan parameter dalam fungsi *ls* adalah sebagai berikut :

```
saronde_os@saronde:~/Desktop$ ls -l
total 3
-rw-r--r-- 1 saronde_os saronde_os 119 Jan 23 11:08 Linux
drwxr-xr-x 3 saronde_os saronde_os 4096 Jan  8 17:47 Pembelajaran
drwxr-xr-x 5 saronde_os saronde_os 4096 Des 24 10:24 Project (urgent)
```

Keterangan :

Detail informasi tentang keterangan parameter untuk masing-masing fungsi, dapat anda akses melalui menu help (file dokumentasi) dari perintah bash, dengan mengetikkan perintah :

```
saronde_os@saronde:~/Desktop$ ls - - help
Usage: ls [OPTION]... [FILE]...
List information about the FILES (the current directory by default).
Sort entries alphabetically if none of -cftuvSUX nor --sort is specified.
Mandatory arguments to long options are mandatory for short options too.
  -a, --all          do not ignore entries starting with .
  -l                use a long listing format
.....
```

```
saronde_os@saronde:~/Desktop$ man help
```

```
LS(1)                                User Commands                                LS(1)
NAME
    ls - list directory contents
SYNOPSIS
    ls [OPTION]... [FILE]...
DESCRIPTION
    List information about the FILES (the current directory by default).
    Sort entries alphabetically if none of -cftuvSUX nor --sort is speci-
    fied.
    Mandatory arguments to long options are mandatory for short options
    too.
    -a, --all
    -l      use a long listing format
```

- **Parameter -l**, merupakan sebuah fungsi yang berfungsi untuk menampilkan informasi dari file / direktori dalam sebuah direktori secara detail, meliputi jumlah konten direktori, akses permission, hak kepemilikan user & group terhadap file, kapasitas, dan tanggal/bulan/jam pembuatan.

Latihan :

Jika terdapat struktur direktori dengan struktur hirarki, sebagai berikut :

```
linux
├─ debian
│   └─ ubuntu
│       └─ versidesktop
├─ redhat
│   └─ fedora
│       └─ versidesktop
└─ slackware
    └─ gentoo
        └─ versiserver
6 directories, 3 files
```

- Masuklah ke direktori fedora, slackware dan fedora, selanjutnya keluar dari direktori tersebut sampai dengan masuk pada direktori linux
- Tampilkan isi dari masing-masing direktori yang ditampilkan diatas.

Penyelesaian :

```
root@saronde:/linux# cd redhat/fedora/  
root@saronde:/linux/redhat/fedora# pwd  
/linux/redhat/fedora  
root@saronde:/linux/redhat/fedora# ls -l  
-rw-r--r-- 1 saronde_os saronde_os 119 Jan 23 11:08 versidesktop
```

```
root@saronde:/linux# cd slackware/  
root@saronde:/linux/slackware# pwd  
/linux/slackware  
root@saronde:/linux/slackware# ls -l  
drwxr-xr-x 3 saronde_os saronde_os 4096 Jan 8 17:47 gentoo
```

```
root@saronde:/linux# cd redhat  
root@saronde:/linux/redhat# cd fedora/  
root@saronde:/linux/redhat/fedora# pwd  
/linux/redhat/fedora/  
root@saronde:/linux/redhat/fedora# ls -l  
-rw-r--r-- 1 saronde_os saronde_os 119 Jan 23 11:08 versidesktop
```

Catatan :

Penggunaan perintah `cd`, apabila diketikkan secara single command (tanpa direktori tujuan), akan secara otomatis mengarah ke home direktori. Selain fungsi `cd`, untuk masuk ke home direktori kita bisa mengetikkan perintah `cd ~`, yang secara otomatis akan mengarahkan path ke direktori home secara default.

```
root@saronde:/linux/redhat/fedora# pwd  
/linux/redhat/fedora  
root@saronde:/linux/redhat/fedora# cd  
root@saronde:~# pwd  
/root
```

5.2. Manipulasi File dan direktori

Pada tahapan ini kita akan belajar proses manipulasi file dan direktori menggunakan program bash shell, perintah-perintah manipulasi bash terbagi menjadi beberapa perintah yang dianggap urgen adalah sebagai berikut :

- **cp (copy)**, perintah yang berfungsi untuk melakukan proses copy-paste file / direktori
 - **mv (move & rename)**, perintah yang berfungsi untuk melakukan perpindahan (cut) file & direktori dan proses perubahan nama dari file atau direktori
 - **mkdir (make directory)**, perintah yang berfungsi untuk melakukan proses pembuatan direktori (direktori)
 - **touch**, perintah yang berfungsi untuk membuat file.
 - **rmdir (remove directory)**, perintah yang berfungsi untuk melakukan proses penghapusan direktori (direktori) yang kosong.
 - **rm (remove)**, perintah yang berfungsi untuk melakukan proses penghapusan file / direktori
 - **ln**, perintah yang berfungsi untuk membuat link (pintas) / shortcut dari sebuah file direktori.
 - **cat**, perintah yang berfungsi untuk membaca sebuah file dan fungsi lainnya.
 - **cp (copy)**, perintah yang berfungsi untuk melakukan proses copy-paste file / direktori.
- Contoh penggunaan adalah sebagai berikut :

```
root@saronde:/# cp [direktori/file] [tujuan direktori]
```

Berikut adalah contoh hirarki direktori :

```
linux
├─ debian
│   └─ ubuntu
│       └─ versidesktop
├─ redhat
│   └─ fedora
│       └─ versidesktop
└─ slackware
    └─ gentoo
        └─ versiserver

6 directories, 3 files
```

Latihan

1. Duplikat file versidesktop yang terletak pada direktori ubuntu ke direktori gento

```
root@saronde:/linux# cp debian/ubuntu/versidesktop slackware/gentoo/  
root@saronde:/linux# ls slackware/gentoo/  
versidesktop versiserver
```

2. Duplikat file versiserver yang terletak pada direktori gentoo ke direktori fedora

```
root@saronde:/linux# cp slackware/gentoo/versiserver redhat/fedora/  
root@saronde:/linux# ls redhat/fedora/  
versidesktop versiserver
```

3. Duplikat direktori gento dan masukkan kedirektori fedora

```
root@saronde:/linux# cp -R slackware/gentoo redhat/fedora/  
root@saronde:/linux# ls redhat/fedora/  
gentoo versidesktop versiserver
```

4. Duplikatlah direktori fedora, menjadi direktorifedora2 dan direktorifedora3

```
root@saronde:/linux# cp -R redhat/fedora redhat/direktorifedora1  
root@saronde:/linux# cp -R redhat/fedora redhat/direktorifedora2  
root@saronde:/linux# ls redhat  
direktorifedora1 direktorifedora2 fedora
```

5. Duplikatlah direktori gentoo, menjadi direktori slacksoft1 dan slacksoft2

```
root@saronde:/linux# cp -R slackware/gentoo slackware/slacksoft1  
root@saronde:/linux# cp -R slackware/gentoo slackware/slacksoft2  
root@saronde:/linux# ls slackware/  
gentoo slacksoft1 slacksoft2
```

Keterangan :

Perintah *cp* (copy) merupakan perintah yang memiliki parameter, parameter yang digunakan diatas adalah *-R* (*recursive*) yang berfungsi untuk melakukan duplikasi sebuah direktori beserta isi dan strukturnya secara keseluruhan. Perintah *cp* jika digunakan tanpa parameter, hanya akan dapat melakukan duplikasi single data, misalnya 1 buah file, dan 1 buah direktori yang kosong (*jika direktori tersebut terisi file / direktori, isi dari direktori tersebut tidak akan terduplikasi*)

Informasi lengkap tentang penggunaan parameter pada fungsi duplikasi “*cp*”, dapat anda lihat pada halaman manual yang telah disediakan :

```
root@saronde:/linux# cp --help
Usage: cp [OPTION]... [-T] SOURCE DEST
  or:  cp [OPTION]... SOURCE... DIRECTORY
  or:  cp [OPTION]... -t DIRECTORY SOURCE...
Copy SOURCE to DEST, or multiple SOURCE(s) to DIRECTORY.
Mandatory arguments to long options are mandatory for short options too.
-a, --archive                same as -dR --preserve=all
  --attributes-only          don't copy the file data, just the attributes
  --backup[=CONTROL]        make a backup of each existing destination file
-f, --force                  if an existing destination file cannot be
                             opened, remove it and try again (redundant if
                             the -n option is used)
-R, -r, --recursive          copy directories recursively
.....
```

- **mv (move & rename)**, perintah yang berfungsi untuk melakukan perpindahan (cut) file & direktori dan proses perubahan nama dari file atau direktori, *mv* merupakan perintah yang bisa dibilang unik, ya karena memiliki dua fungsi dan kegunaannya, terkadang fungsi ini bisa membuat pengguna kesulitan untuk membedakan antara satu fungsi dengan yang lainnya.

mv (perubahan nama)

```
root@saronde:/linux# mv namafile namafilebaru
```

```
root@saronde:/linux# mv namadirektori namadirektoribaru
```

mv (perpindahan)

```
root@saronde:/linux# mv file/direktori direktori_tujuan
```

Penggunaan perintah mv untuk melakukan proses perubahan nama dan perpindahan file / direktori dirasa cukup mudah, agar bisa lebih dipahami, lakukan praktik penggunaan fungsi mv.

Berikut adalah struktur hirarki direktori, yang akan digunakan untuk proses latihan.

```
linux
├── debian
│   └── ubuntu
├── redhat
│   └── fedora
└── slackware
    └── suse
3 directories, 3 files
```

1. Lakukan perubahan nama pada direktori **redhat** menjadi **RHEL** dan direktori **fedora** menjadi **fedora12**

```
root@saronde:/linux# mv redhat RHEL
root@saronde:/linux# ls
debian RHEL slackware
root@saronde:/linux# mv RHEL/fedora RHEL/fedora12
root@saronde:/linux# ls RHEL/
fedora12
```

2. Lakukan perubahan nama pada file **suse** menjadi **openSUSE**

```
root@saronde:/linux# mv slackware/suse slackware/openSUSE
root@saronde:/linux# ls slackware/
openSUSE
```

3. Pindahlah file **openSUSE** ke direktori **RHEL**

```
root@saronde:/linux# mv slackware/openSUSE RHEL/
root@saronde:/linux# ls RHEL/
fedora12  openSUSE
root@saronde:/linux# ls -l slackware/
total 0
root@saronde:/linux# tree
  linux
  ├── debian
  |   └── ubuntu
  ├── RHEL
  |   ├── fedora12
  |   └── openSUSE
  └── slackware
3 directories, 3 files
```

4. Pindahlah direktori **debian** kedalam direktori **RHEL**

```
root@saronde:/linux# mv debian RHEL/
root@saronde:/linux# ls
RHEL  slackware
root@saronde:/linux# ls RHEL/
debian  fedora12  openSUSE
root@saronde:/linux# tree
  linux
  ├── RHEL
  |   ├── debian
  |   |   └── ubuntu
  |   ├── fedora12
  |   └── openSUSE
  └── slackware
3 directories, 3 files
```

- **mkdir (make directory)**, perintah yang berfungsi untuk membuat direktori (direktori).

```
root@saronde:/linux# mkdir [namadirektori]
```

Contoh penggunaan perintah mkdir adalah sebagai berikut

```
root@saronde:/linux# mkdir latihan
root@saronde:/linux# ls
latihan
```

Jika kita ingin membuat sebuah direktori baru (latihan1) didalam direktori latihan adalah sebagai berikut :

```
root@saronde:/linux# mkdir latihan/latihan1
root@saronde:/linux# ls latihan/
latihan1
```

Jika kita ingin membuat sebuah direktori baru (latihan1.1) didalam direktori latihan dan latihan1 adalah sebagai berikut :

```
root@saronde:/linux# mkdir latihan/latihan1/latihan1.1
root@saronde:/linux# tree latihan/
latihan/
├── latihan1
│   └── latihan1.1
2 directories, 0 files
```

- **touch**, perintah yang berfungsi untuk membuat file. Penggunaan perintah touch memiliki sedikit kesamaan dengan perintah sebelumnya

```
root@saronde:/linux/latihan# touch [nama file]
```

Contoh penggunaan adalah sebagai berikut :

```
root@saronde:/linux/latihan# touch filepertama
root@saronde:/linux/latihan# ls -l
total 4
-rw-r--r-- 1 root root  0 Jan 31 10:21 filepertama
drwxr-xr-x 3 root root 4096 Jan 31 10:18 latihan1
```

Penggunaan perintah touch, untuk membuat file pada sub-direktori adalah sebagai berikut :

```
root@saronde:/linux# touch latihan/latihan1/latihan1.1/sublatihan
root@saronde:/linux# tree latihan
latihan
├─ filepertama
├─ latihan1
│   └─ latihan1.1
│       └─ sublatihan
2 directories, 2 files
```

- **rmdir (remove directory)**, perintah yang berfungsi untuk melakukan proses penghapusan direktori (direktori) yang kosong.

```
root@saronde:/linux# rmdir namadirektori
```

Berikut adalah penggunaan perintah `rmdir` untuk menghapus direktori, akan tetapi direktori yang akan dihapus, tidak bisa terhapus dikarenakan direktori tersebut tidak kosong / masih memiliki isi.

```
root@saronde:/linux# rmdir latihan
rmdir: failed to remove `latihan': Directory not empty
```

Berikut adalah penggunaan perintah `rmdir` yang benar, penggunaan perintah `rmdir`, harus menghapus sub-sub direktori terlebih dahulu, jika direktori tersebut sudah kosong, maka direktori tersebut sudah bisa dihapus.

```
root@saronde:/linux# rmdir latihan2/latihan2.1/
root@saronde:/linux# rmdir latihan2/latihan2.2/
root@saronde:/linux# rmdir latihan2
```

- **rm (remove)**, perintah yang berfungsi untuk melakukan proses penghapusan file / direktori. `rm` memiliki kelebihan dibanding perintah `rmdir`, karena dengan `rm` kita bisa melakukan proses penghapusan direktori secara rekursif, dengan menggunakan fungsi parameter (*rf*) untuk informasi lengkap tentang parameter `rm` dapat anda akses melalui manual page dengan perintah (`rm --help` dan `man rm`). Penggunaan fungsi `rm` adalah sebagai berikut :

```
root@saronde:/linux# rm [parameter] [nama file / direktori]
```

Berikut adalah beberapa latihan penggunaan fungsi rm, terdapat struktur direktori sebagai berikut :

```
root@saronde:/linux# tree
.
├─ file1
├─ file2
├─ file3
├─ latihan1
│   └─ filelatihan1
3 directories, 6 files
```

1. Hapuslah semua file dengan nama file1, file2 dan file3

Teknik penghapusan 1

Merupakan teknik tradisional yang untuk menghapus file1, 2, dan 3 secara berurutan.

```
root@saronde:/linux# rm file1
root@saronde:/linux# rm file2
root@saronde:/linux# rm file3
root@saronde:/linux# tree
.
├─ latihan1
│   └─ filelatihan1
├─ latihan2
│   └─ filelatihan2
└─ latihan3
    └─ filelatihan3
3 directories, 3 files
```

Teknik penghapusan 2

Merupakan teknik penghapusan file yang cukup efisien karena menghapus semua file dalam 1 baris perintah, secara berurutan yang dipisahkan dengan spasi.

```
root@saronde:/linux# rm file1 file2 file3
```

Teknik penghapusan 3

Merupakan teknik yang paling efisien untuk menghapus direktori secara keseluruhan tanpa perlu mengetikkan keseluruhan nama file, kita hanya perlu menuliskan nama file diikuti dengan tanda (*) yang berarti bahwa semua file yang menggunakan nama file didepannya akan dihapus, dalam hal ini terdapat 3 buah file yang menggunakan nama dengan awalan file yaitu file1, file2, dan file3.

```
root@saronde:/linux# rm file*
```

2. Hapuslah direktori latihan1 beserta filelatihan1, dengan menggunakan parameter.

```
root@saronde:/linux# rm -rf latihan1/
root@saronde:/linux# tree
linux
├─ latihan2
│   └─ filelatihan2
└─ latihan3
    └─ filelatihan3
2 directories, 2 files
```

- **ln**, perintah yang berfungsi untuk membuat link (pintas) / shortcut dari sebuah file direktori. Penggunaan perintah *ln*, dapat dengan mengikuti format berikut :

```
root@saronde:/linux# ln [file/direktori] [link]
```

Misalnya kita membuat sebuah file dengan nama latihan2.txt, dimana didalam file tersebut berisi sebuah data / informasi berikut : *“buku panduan Linux fundamental”*.

```
root@saronde:/linux# touch latihan1.txt
root@saronde:/linux# nano latihan1.txt
```

```
GNU nano 2.2.6
```

```
File: latihan1.txt
```

```
buku panduan linux fundamental
```

```
root@saronde:/linux# ln latihan1.txt shortcut.txt
root@saronde:/linux# ls
latihan1.txt latihan2 latihan3 shortcut.txt
root@saronde:/linux# cat shortcut.txt
buku panduan linux fundamental
root@saronde:/linux# cat latihan1.txt
buku panduan linux fundamental
```

Keterangan :

Seperti yang telah ditunjukkan diatas, kita sudah membuat sebuah file dengan nama *latihan1.txt* dengan isian data “*buku panduan linux fundamental*”, selanjutnya dibuatkan link, dengan perintah *ln*. Selain membuat shortcut file, *ln* juga bisa membuat shortcut dari direktori.

- **cat**, perintah yang berfungsi untuk membaca sebuah file dan fungsi lainnya. Perintah *cat* sudah pernah digunakan pada penjelasan sebelumnya, selain membaca file *cat* juga dapat difungsikan untuk membuat sebuah file, dan menggabungkan antar file-file yang berbeda.

Membaca file dengan *cat*

```
root@saronde:/linux# cat namafile.txt
buku panduan linux fundamental
```

Membuat file dengan *cat*

```
root@saronde:/linux# cat > namafile.txt
buku panduan linux fundamental
root@saronde:/linux# cat namafile.txt
buku panduan linux fundamental
```

```
root@saronde:/linux# cat >> namafile.txt
ini baris tambahan saja
root@saronde:/linux# cat namafile.txt
buku panduan linux fundamental
ini baris tambahan saja
```

Keterangan :

Fungsi simbolisasi *>* untuk melakukan replace inputan data dalam file tersebut, sedangkan simbolisasi *>>* berfungsi untuk menambahkan isi dari file tersebut.

Menggabungkan file

```
root@saronde:/linux# cat > file1.txt
ini isi file1
root@saronde:/linux# cat > file2.txt
ini isi file2
root@saronde:/linux# cat > file3.txt
ini isi file3
root@saronde:/linux# cat file1 file2 file3 > filesaya.txt
root@saronde:/linux# cat filesaya.txt
ini isi file1
ini isi file2
ini isi file3
```

5.3. Perintah Dasar Shell

Pada bagian ini akan dijelaskan perintah-perintah dasar yang wajib dikenali oleh seorang pengguna linux.

Sebelum menjelaskan tentang perintah shell, terlebih dahulu kita bisamempelajari sebuah perintah shell, kita bisa menggunakan perintah-perintah yang berfungsi untuk menampilkan informasi secara detail tentang perintah, parameter dan penggunaannya.

man

menampilkan informasi detail tentang perintah, dalam bentuk halaman page lengkap.

Penggunaan :

```
root@saronde:/linux# man ls
```

LS(1) User Commands LS(1)

NAME

ls - list directory contents

SYNOPSIS

ls [OPTION]... [FILE]...

DESCRIPTION

List information about the FILES (the current directory by default).
Sort entries alphabetically if none of -cftuvSUX nor --sort is specified.

.....

whatis

menampilkan informasi tentang fungsi / kegunaan dari sebuah perintah shell secara singkat.

Penggunaan :

```
root@saronde:/linux# whatis ls
ls (1)          - list directory contents
root@saronde:/linux# whatis nano
nano (1)       - Nano's ANOther editor, an enhanced free Pico clone
```

locate

Melakukan pencarian sebuah file dengan kata kunci yang diinginkan, kepada seluruh media penyimpanan.

Penggunaan :

```
root@saronde:/linux# locate [kata kunci]
```

apropos

mencari petunjuk penggunaan dari sebuah perintah shell berdasarkan deskripsinya.

Penggunaan :

```
root@saronde:/linux# apropos nano
nano (1)          - Nano's ANOther editor, an enhanced free Pico clone
nanorc (5)       - GNU nano's rcfile
pico (1)         - Nano's ANOther editor, an enhanced free Pico clone
rnano (1)       - Restricted mode for Nano's ANOther editor, an enhance...
```

Jika penjelasan diatas sudah bisa dipahami dan dimengerti, SELAMAT....!!! Anda sudah bisa belajar *shell* secara mandiri, karena setiap perintah bash yang dikembangkan akan selalu memiliki dokumentasi penjelasan dan penggunaannya

Pada bagian selanjutnya akan dijelaskan beberapa perintah bash shell yang perlu diketahui dan dipelajari.

which – menampilkan letak sebuah command line

berfungsi untuk menampilkan informasi tentang letak / lokasi penyimpanan sebuah perintah shell dalam sistem linux.

Penggunaan :

```
root@saronde:/linux# which ls
/bin/ls
root@saronde:/linux# which touch
/usr/bin/touch
```

type – menampilkan tipe perintah

berfungsi untuk menampilkan informasi tentang perintah dari sebuah shell.

Penggunaan :

```
root@saronde:/linux# type cat
cat is hashed (/bin/cat)
root@saronde:/linux# type nano
nano is hashed (/usr/bin/nano)
root@saronde:/linux# type ls
ls is aliased to `ls --color=auto'
```

tree – menampilkan struktur hirarki file/direktori

berfungsi untuk menampilkan susunan file / direktori dalam bentuk pohon (tree).

Penggunaan :

```
root@saronde:/linux# tree
linux
├─ latihan1
│   └─ filelatihan1
├─ latihan3
│   └─ filelatihan3
```

2 directories, 2 files

alias – membuat perintah shell

jika sebelumnya kita hanya belajar menggunakan perintah shell, dengan perintah alias kita bisa membuat sebuah perintah shell yang merupakan sebuah deskriptif fungsi dari fungsi bash lain.

Penggunaan :

```
root@saronde:/linux# alias catatan='nano'
root@saronde:/linux# catatan file1
```

ini isi dari file1. yang dibuat menggunakan editor catatan ('nano')

```
root@saronde:/linux# type catatan
catatan is aliased to `nano'
```

Sedangkan untuk menghapus fungsi alias dari perintah yang sudah dibuat sebelumnya, bisa menggunakan perintah berikut :

```
root@saronde:/linux# unalias catatan
root@saronde:/linux# type catatan
bash: type: catatan: not found
```

grep

berfungsi untuk melakukan pencarian kata dalam sebuah file

Penggunaan :

```
root@saronde:/linux# cat fileurutan.txt
redhat
linux
debian
slackware
ubuntu
xandros
openSuse
root@saronde:/linux# grep ubuntu fileurutan.txt
ubuntu
```

Keterangan :

Perintah grep dapat digunakan secara pipeline (|), untuk melakukan filtering terhadap file yang akan ditampilkan. File yang ingin ditampilkan adalah file *passwd* yang terletak di direktori */etc*. Didalam file tersebut berisikan banyak data tentang user dalam sistem, akan tetapi kita bisa melakukan filtering data yang ditampilkan menggunakan perintah grep.

uniq

berfungsi untuk menampilkan karakter dalam sebuah file secara unik.

Penggunaan :

```
root@saronde:/linux# cat fileuniq.txt
1
3
3
3
5
root@saronde:/linux# uniq fileuniq.txt
1
3
5
```

sort

berfungsi untuk menampilkan karakter dalam file secara berurutan

Penggunaan :

```
root@saronde:/linux# cat fileurutan.txt
redhat
debian
slackware
xandros
openSuse
root@saronde:/linux# sort fileurutan.txt
debian
openSuse
redhat
slackware
xandros
```

head & tail (kepala & ekor)

pada bagian ini akan dijelaskan dua buah perintah yang berfungsi untuk menampilkan informasi dalam sebuah file, perbedaan antar head & tail adalah jika head menampilkan informasi mulai dari baris pertama, sedangkan tail akan menampilkan informasi dari bawah.

Penggunaan :

```
root@saronde:/linux# head [parameter] [namafile]
```

```
root@saronde:/linux# head /etc/apt/sources.list
```

```
deb http://repo.ugm.ac.id/ubuntu/ precise main restricted universe multiverse
deb http://repo.ugm.ac.id/ubuntu/ precise-updates main restricted universe multiverse
deb http://repo.ugm.ac.id/ubuntu/ precise-security main restricted universe multiverse
.....
```

```
root@saronde:/linux# tail /etc/apt/sources.list
```

```
# deb http://archive.canonical.com/ubuntu precise partner
# deb-src http://archive.canonical.com/ubuntu precise partner
.....
```

Secara default perintah head/tail akan menampilkan data dari atas atau dari bawah, akan menampilkan data dengan jumlah 10 baris dari awal dan akhir. Untuk menampilkan jumlah baris sesuai dengan keinginan, bisa dengan menggunakan parameter *-n (jumlah baris)*.

```
root@saronde:/linux# cat fileurutan.txt
```

```
redhat
linux
debian
slackware
ubuntu
xandros
openSuse
```

Diatas terdapat file dengan nama fileurutan.txt, dengan jumlah data 7 baris, selanjutnya kita akan menampilkan 3 baris dari atas dengan (*head*) dan 3 baris dari bawah (*tail*).

```
root@saronde:/linux# head -n 3 fileurutan.txt
```

```
redhat
linux
debian
```

```
root@saronde:/linux# tail -n 3 fileurutan.txt
```

```
ubuntu
xandros
openSuse
```

top

Berfungsi untuk menampilkan informasi tentang proses (tasking) dalam sistem.

root@saronde:/linux# top

Tasks: 173 total, 2 running, 170 sleeping, 0 stopped, 1 zombie
Cpu(s): 5.0%us, 2.3%sy, 0.0%ni, 92.4%id, 0.4%wa, 0.0%hi, 0.0%si, 0.0%st
Mem: 1989948k total, 1826400k used, 163548k free, 258196k buffers
Swap: 4063896k total, 288k used, 4063608k free, 997200k cached

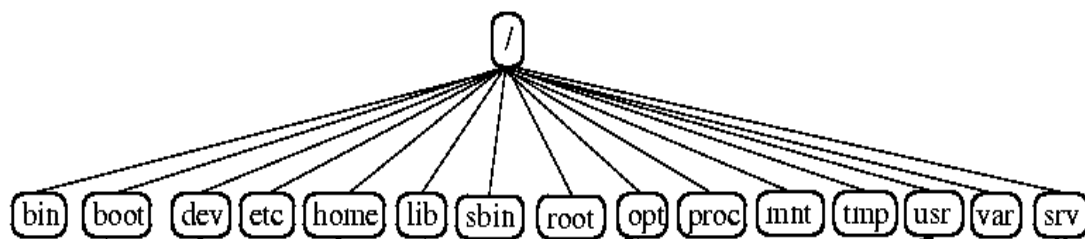
PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
9446	saronde_	20	0	307m	56m	21m	S	23	2.9	6:28.57	totem
1379	root	20	0	79312	22m	8828	S	10	1.2	19:07.83	Xorg
4826	saronde_	20	0	473m	104m	24m	R	10	5.4	20:07.21	cinnamon

Bagian VI

Struktur Filesystem

Semua distribusi sistem operasi berbasis linux memiliki model filesystem berbentuk hirarki tree (pohon), dimana berawal dari satu direktori inti yaitu root. Bentuk struktur direktori akan membentuk sebuah pohon terbalik, dimana diseluruh direktori tersebut berisi direktori / file yang memiliki kegunaan dan fungsi yang berbeda-beda, beberapa kegunaan direktori filesystem sudah dijelaskan pada bagian sebelumnya.

Struktur direktori pada masing-masing distribusi linux relatif berbeda, bergantung pada sistem kernel yang digunakan. Struktur Direktori Linux mengikuti standar “Filesystem Hierarchy Structure (FHS)” yang dipelihara oleh “Free Standars Group”.



Pada bagian ini, akan dijelaskan tentang fungsi dan kegunaan dari masing-masing direktori diatas.

6.1. /bin

Merupakan sebuah direktori filesystem yang berisi file executable, binarycommand dalam sistem linux yang dapat dieksekusi secara langsung oleh pengguna. Seperti perintah-perintah bash : ls, mkdir, dll

6.2. /boot

Merupakan direktori yang berisi file yang dibutuhkan oleh sistem boot loader termasuk boot_loader_sys, kernel, initrd, memtest, config device dan system.mapping, yang dibutuhkan dan digunakan pada saat sistem restart / booting (reboot).

6.3. /dev

Merupakan direktori yang berisi file konfigurasi hardware yang ada di komputer, dan modul-modul driver yang terdeteksi dalam sistem. Contohnya : File hardisk => sda, sda1, sda2, sr0 dll.

6.4. /etc

Direktori yang berisi file-file yang digunakan untuk mengkonfigurasi layanan (service) sistem linux. File-file ini biasanya merupakan file-file yang berbentuk script atau file-file dengan tipe konfigurasi text. Contoh : ip address, dhcp, dns, webserver, network, routing, dll

6.5. /home

Berisi data dan informasi tentang users yang terdaftar dalam sistem linux. Seperti file pengguna, pengaturan pengguna, profil dan pengaturan sistem dalam lingkup pengguna.

6.6. /lib

Berisi file-file pendukung / pustaka untuk yang dibutuhkan oleh program, file binari dan file executable, yang berada di dalam direktori /bin

6.7. /sbin

berisi file executable / program atau perintah shell yang digunakan oleh superuser. Contoh : seperti *ifconfig* (perintah yang menampilkan informasi tentang kartu jaringan / network device yang terpasang pada mesin) , yang dapat mengeksekusi / menggunakan perintah tersebut hanyalah seorang superuser (*root*).

6.8. /root

merupakan home direktori dari superuser dalam sistem Linux

6.9. /opt

Sebuah direktori yang digunakan oleh perangkat lunak pihak ketiga dan bukan merupakan fasilitas / utilitas dalam sistem GNU/Linux. Contohnya : X/LAMP, JVM, dll

6.10. /proc

Direktori yang berisikan file buffer (hasil proses) yang digunakan untuk menginformasikan proses yang sedang berjalan pada sistem Linux. Contoh : seperti hasil running / buffer dari perintah shell => meminfo, cpuinfo, version, uptime, dll .

6.11. /mnt

Direktori mounting yang berfungsi untuk menampung tautan antara storage device dengan sistem.

6.12. /tmp

Merupakan sebuah direktori sementara yang menyimpan file sementara dan akan dihapus ketika sistem di-reboot (dijalankan kembali).

6.13. /usr

Berisi file paket aplikasi, program, dokumentasi, konfigurasi, binary, library dan source aplikasi linux, yang dapat dijalankan oleh semua pengguna (user) yang merupakan utilitas dari aplikasi dalam sistem.

6.14. /var

Direktori yang berisi variabel sistem seperti logs, file spool untuk printer, webserver, mail, proses yang dilakukan, file logs dan lain-lain.

6.15. /srv

Merupakan direktori yang berisi file service dan layanan dari sistem utilitas program pada sistem GNU/Linux.

Bagian VII

Manajemen File

Manajemen file merupakan sebuah kelebihan dalam sistem berbasis Linux, Linux dapat mengatur sebuah file dalam sistem linux dengan attribut yang sangat lengkap yang meliputi hak akses kepemilikan user dan group dari sebuah file tersebut, maupun hak akses dari sebuah file tersebut yang meliputi tiga metode akses (baca, tulis dan eksekusi).

Beberapa hal diatas merupakan sebuah tradisi yang diambil dari sistem berbasis Unix Like, yang sangat berbeda dengan sistem MS-DOS, yang tidak hanya menawarkan sistem yang aman, melainkan menawarkan sistem yang multitasking, dan multiuser.

1. **Multitasking** merupakan kemampuan untuk memproses data dengan cepat (secara bersamaan).
2. **Multiuser**, fitur yang memungkinkan lebih dari satu pengguna untuk menggunakan sistem komputer tersebut secara bersamaan.

Sistem Linux membagi file di linux kedalam beberapa 2 kategori meliputi : file dan direktori. Pada bagian ini akan dibahas beberapa perintah bash yang berkaitan dengan manajemen file diantaranya adalah sebagai berikut :

- `chmod` — melakukan perubahan hak akses permission pada sebuah file/direktori
- `chown` — Mengganti kepemilikan (user) dari sebuah file / direktori
- `chgrp` — Mengganti kepemilikan (group) dari sebuah file / direktori
- `stat` — menampilkan informasi secara detail tentang sebuah file

7.1 Mode Kepemilikan (Pengguna, Group pengguna dan Orang lain)

Dalam sistem Linux, seorang pengguna yang membuat sebuah file / direktori pada shell dari user tersebut secara otomatis file / direktori tersebut akan menjadi milik dari user tersebut, bukti kepemilikan tersebut dapat kita lihat menggunakan perintah `stat` atau `ls` dengan parameter `-l`.

Berikut adalah hasil eksekusi perintah `stat` dan `ls`, untuk menampilkan informasi tentang sebuah file *shadow*, yang terletak pada direktori */etc*.

```
saronde_os@saronde:~$ stat /etc/shadow
  File: `/etc/shadow'
  Size: 1677      Blocks: 8      IO Block: 4096   regular file
Device: 807h/2055d Inode: 1846612  Links: 1
Access: (0640/-rw-r-----)  Uid: (  0/   root)   Gid: ( 42/  shadow)
Access: 2014-02-01 17:20:01.565324557 +0800
Modify: 2014-01-23 06:22:50.760755207 +0800
Change: 2014-01-23 06:22:50.792755356 +0800
 Birth: -
```

Keterangan :

1. File: `/etc/shadow'

Baris pertama menunjukkan nama dan lokasi file.

2. Size: 1677 Blocks: 8 IO Block: 4096 regular file

Baris kedua menunjukkan ukuran sebuah file, I/O block, dan size blog

3. Device: 807h/2055d Inode: 1846612 Links: 1

Baris ketiga menunjukkan device code, Inode, dan links dari sebuah file.

4. Access: (0640/-rw-r-----) Uid: (0/ root) Gid: (42/ shadow)

Baris keempat menunjukkan hak akses, hak kepemilikan user dan group dari sebuah file dengan format oktal dan alfabet (rwx) (*permission*), Uid (untuk user id), dan Gid (untuk group).

5. Access: 2014-02-01 17:20:01.565324557 +0800

6. Modify: 2014-01-23 06:22:50.760755207 +0800

7. Change: 2014-01-23 06:22:50.792755356 +0800

Baris lima, enam dan tujuh, merupakan baris yang menunjukkan tanggal dan waktu akses, modifikasi dan perubahan yang dilakukan terhadap file.

Pada bagian pembahasan bagian ini kita akan belajar untuk melakukan perubahan hak kepemilikan dari sebuah file, untuk melakukan perubahan kepemilikan bisa dengan menggunakan perintah *chown* dan *chgroup*.

Format perintah *chown* dan *chgrp* adalah sebagai berikut :

Format :

```
root@saronde:~# chown namauser namafile/direktori
```

Penggunaan :

```
root@saronde:/home/saronde_os# ls -l gorindo.sql
-rw-r--r-- 1 saronde_os saronde_os 7794 Okt 27 20:08 gorindo.sql
root@saronde:/home/saronde_os# chown root gorindo.sql
root@saronde:/home/saronde_os# ls -l gorindo.sql
-rw-r--r-- 1 root saronde_os 7794 Okt 27 20:08 gorindo.sql
```

Format :

```
root@saronde:~# chgrp namagroup namafile/direktori
```

Penggunaan :

```
root@saronde:/home/saronde_os# ls -l gorindo.sql
-rw-r--r-- 1 root saronde_os 7794 Okt 27 20:08 gorindo.sql
root@saronde:/home/saronde_os# chgrp root gorindo.sql
root@saronde:/home/saronde_os# ls -l gorindo.sql
-rw-r--r-- 1 root root 7794 Okt 27 20:08 gorindo.sql
```

Perintah *chown* memiliki kelebihan dibandingkan *chgrp*, karena *chown* bisa melakukan perubahan kepemilikan dan group dari sebuah file, tanpa menggunakan perintah *chgrp*. Adalah sebagai berikut :

- Perintah untuk melakukan perubahan pemilikan user dan group
root@saronde:~# **chown namauser:namagroup namafile/direktori**
- Perintah untuk melakukan perubahan pemilikan user
root@saronde:~# **chown namauser: namafile/direktori**
- Perintah untuk melakukan perubahan pemilikan group
root@saronde:~# **chown :namagroup namafile/direktori**

Daftar pengguna dan group yang terdaftar dalam sistem sebuah sistem linux dapat dilihat pada file *passwd* dan *group* yang terletak pada direktori */etc/*.

```
root@saronde:/# cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
saronde_os:x:1000:1000:saronde OS,,,:/home/saronde_os:/bin/bash
.....
```

Keterangan :

Pada file passwd berisi daftar informasi dari sebuah user, mulai dari nama user, enkripsi password, id user dan id group, home direktori, dan bash home.

```
root@saronde:/# cat /etc/group
root:x:0:
saronde_os:x:1000:
.....
```

Keterangan :

Pada file group berisi daftar informasi dari sebuah group, mulai dari nama group, enkripsi akses, id user dan id group.

7.2 Mode Akses (Baca, Tulis dan Eksekusi)

Mode akses sebuah file dalam sistem Linux terbagi menjadi tiga bagian baca, tulis dan hapus, seorang user yang memiliki sebuah file / data dalam sebuah media penyimpanan dapat mengatur keamanan dari sebuah data baik berupa file / direktori, agar tidak diakses (baca/tulis/eksekusi) oleh pengguna lainnya yang tidak memiliki hak akses atas sebuah file.

Proses pengatur mode akses terhadap sebuah file, dapat dengan menggunakan perintah *chmod*, diikuti dengan nilai oktal dari sebuah akses permission sebuah file. Jika kita mengamati hasil output dari file melalui perintah *ls -l* berikut :

```
saronde_os@saronde:~$ ls -l
total 2
drwxr-xr-x 2 saronde_os saronde_os 4096 Feb  2 13:46 direktori1
-rw-r--r-- 1 saronde_os saronde_os   0 Feb  2 13:46 file1
```

Terdapat dua output dari perintah diatas yaitu output yang memiliki result nama file *direktori1* dan *file1*, karakteristik sebuah file adalah terletak pada simbolisasi yang terletak diawal baris untuk masing-masing output.

Pada awal digit pada karakter yang ditandai diatas, terdapat simbol “d” dan “-”, hal tersebut mendeskripsikan bahwa data tersebut adalah sebuah direktori (d), dan file (-), sedangkan karakter yang selanjutnya (9 karakter) mengkombinasikan simbolisasi rwx, yang berarti bahwa r (read), w (write) dan x (excute), sedangkan ketiga kali pengulangan karakter tersebut memberikan tanda bahwa 3 karakter yang pertama adalah hak akses untuk pengguna (owner/user), yang kedua (group) dan yang terakhir adalah untuk orang lain (other).

Sehingga dapat dideskripsikan dalam bentuk format tabel berikut :

d	rwx	rwx	rwx
Type file	User / owner permission	Group permission	Other permission

Simbolisasi untuk tipe file dapat dideskripsikan secara detail pada tabel berikut :

Atribut	Tipe File
-	Mendeskripsikan file
d	Mendeskripsikan sebuah direktori / folder
l	Sebuah file yang berbentuk link (tautan/shorcut)
c	Sebuah file ekstra yang berisi konfigurasi perangkat sistem seperti stream byte, terminal, dan modem.
d	Sebuah file unik yang berisi konfigurasi hardware seperti hardisk (sda, hda, sda1, dll)

Simbolisasi untuk permission sebuah file dapat dideksripsikan sebuah berikut :

Atribut	File	Folder
r	Memungkinkan file untuk dibuka dan dibaca.	Memungkin sebuah direktori untuk dilihat dan dibaca attributnya.
w	Memungkinkan file dapat ditulis, dihapus, dan dimodifikasi nama filenya.	Memungkin untuk membuat file dalam direktori, menghapus direktori, perubahan nama, dan melakukan perubahan attribut dari sebuah direktori.
x	Memungkin file untuk dieksekusi, contoh file yang bisa dieksekusi adalah file source code seperti (*.java, *.cpp, *.py, dll)	Memungkinkan direktori untuk dimasuki.

Seperti yang sudah dijelaskan sebelumnya untuk melakukan perubahan permission / hak akses sebuah file dapat diterapkan dengan menggunakan kombinasi angka oktal. *Oktal merupakan sebuah bilangan yang terdiri dari 8 karakter yang mencakup (0-7)*. Model sebuah hak akses permission dalam tabel oktal adalah sebagai berikut :

Oktal	Binary	File Permission
0	000	- - -
1	001	- - x
2	010	- w -
3	011	- w x
4	100	r - -
5	101	r - x
6	110	r w -
7	111	r w x

Berdasarkan pada pendefinisian permission diatas, untuk itu akan diberikan beberapa latihan untuk menyelesaikan permasalahan-permasalahan permission berikut :

1. Terdapat sebuah file dengan nama file1, yang hanya dapat dibaca, dan dieksekusi oleh user, group dan other.
2. Lakukan perubahan permission sebuah file dengan nama file2 agar dapat hanya dapat ditulis dan dieksekusi oleh user dan group, dan tidak dapat diakses oleh orang lain.
3. Lakukan perubahan permission pada sebuah file yang hanya dapat dibaca oleh user, group dan other.

Penyelesaian :

1. Berikut adalah penyelesaian permasalahan permission bagian 1

```
saronde_os@saronde:~$ chmod 555 file1
saronde_os@saronde:~$ ls -l file1
-r-xr-xr-x 1 saronde_os saronde_os 0 Feb  2 13:46 file1
```

2. Berikut adalah penyelesaian permasalahan permission bagian 2

```
saronde_os@saronde:~$ chmod 330 file2
saronde_os@saronde:~$ ls -l file2
--wx-wx--- 1 saronde_os saronde_os 0 Feb  2 15:31 file2
```

3. Berikut adalah penyelesaian permasalahan permission bagian 3

```
saronde_os@saronde:~$ chmod 444 file3
saronde_os@saronde:~$ ls -l file3
-r--r--r-- 1 saronde_os saronde_os 0 Feb  2 15:31 file3
```

Bagian VIII

Manajemen User

Sistem Linux membagi sistem manajemen user kedalam tiga bagian inti yang terdiri dari superuser, user biasa, dan user tamu. Masing-masing pengguna (user) dalam sistem Linux akan dikelompokkan kedalam sebuah group, sehingga semua user yang ada akan memiliki group yang sama maupun berbeda.

Untuk melihat informasi tentang manajemen user dalam sistem linux, kita bisa menggunakan beberapa perintah *bash*, yang akan dijelaskan pada bagian berikut :

- **id**, mencetak informasi lengkap tentang user dan group dan spesifikasi sebuah user sesuai dengan user yang login dalam sistem.

```
saronde_os@saronde:~$ id
uid=1000(saronde_os) gid=1000(saronde_os) groups=1000(saronde_os)
```

Sebelum membahas secara meluas tentang perintah-perintah bash yang berhubungan dengan manajemen user, terlebih dahulu kita harus mengetahui sebuah file penting yang berhubungan dengan user dan pengguna dalam sistem linux.

Direktori yang menyimpan informasi tentang user tersimpan pada direktori */etc*, file konfigurasi tersebut adalah *passwd* dan *group*.

```
root@saronde:/# cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
saronde_os:x:1000:1000:saronde OS,,,:/home/saronde_os:/bin/bash
.....
```

```
root@saronde:/# cat /etc/group
root:x:0:
saronde_os:x:1000:
.....
```

Catatan :

Penjelasan tentang *passwd* dan *group* sudah dijelaskan pada bagian sebelumnya.

- **useradd vs adduser**

Sistem linux memperkenalkan konsep multiuser, yang berarti bahwa sebuah sistem linux dapat digunakan oleh beberapa pengguna yang berbeda-beda, untuk itu kita bisa membuat user sesuai dengan jumlah yang diinginkan. Untuk untuk sebuah user kita bisa menggunakan dua buah perintah *useradd* dan *adduser* ? *Apakah perbedaan antara dua tersebut ?* Kedua perintah tersebut memiliki fungsi dan tujuan yang sama.

useradd

```
root@saronde:~# useradd pengguna1
root@saronde:~# tail /etc/passwd
.....
ntop:x:122:134::/var/lib/ntop:/bin/false
pengguna1:x:1003:1003::/home/pengguna1:/bin/sh
```

adduser

```
root@saronde:~# adduser pengguna2
Adding user `pengguna2' ...
Adding new group `pengguna2' (1005) ...
Adding new user `pengguna2' (1001) with group `pengguna2' ...
Creating home directory `/home/pengguna2' ...
```

Catatan :

Kedua perintah diatas, memiliki kegunaan yang dominan sama, akan tetapi mekanisme pembuatannya berbeda, *useradd* akan membuat sebuah pengguna dalam sistem linux secara instan, berbeda dibandingkan dengan *adduser*, yang mengharuskan untuk kepada pembuat untuk memasukkan informasi-informasi terkait user yang akan dibuat.

Setelah user selesai dibuat selanjutnya kita sudah bisa login ke user yang dibuat dengan menggunakan perintah *su* dan *login*, proses pengguna fungsi *su* dan *login* akan dijelaskan pada bagian selanjutnya.

- **userdel / deluser**

Jika sebelumnya, sudah dijelaskan tatacara pembuatan user dengan dua metode (useradd & adduser) pada bagian ini akan dijelaskan tahapan penghapusan user (pengguna) dengan menggunakan dua metode yaitu userdel dan deluser, kedua perintah diatas memiliki fungsi dan tujuan yang sama, akan tetapi memiliki metode / cara penghapusan yang berbeda.

userdel

```
root@saronde:~# userdel pengguna1
root@saronde:~# tail -n 3 /etc/passwd
proftpd:x:121:65534::/var/run/proftpd:/bin/false
ftp:x:1002:1002:,,,:bagi:/bin/sh
ntop:x:122:134::/var/lib/ntop:/bin/false
```

deluser

```
root@saronde:~# deluser pengguna2
Removing user `pengguna2' ...
Done.
root@saronde:~# tail -n 3 /etc/passwd
proftpd:x:121:65534::/var/run/proftpd:/bin/false
ftp:x:1002:1002:,,,:bagi:/bin/sh
ntop:x:122:134::/var/lib/ntop:/bin/false
```

- **groupadd / addgroup**

Pada bagian ini akan dijelaskan tahapan pembuatan sebuah group dalam sistem Linux, sebelum dijelaskan tentang pengguna perintah *bash*, pada bagian ini akan dijelaskan tentang direktori penting yang berhubungan dengan group, yang terletak pada direktori */etc/group*. Untuk melakukan pembuatan group, bisa dengan menggunakan dua buah perintah.

groupadd

```
root@saronde:~# groupadd group1
root@saronde:~# tail -n 2 /etc/group
ntop:x:134:
group1:x:1005:
```

addgroup

```
root@saronde:~# addgroup group2
Adding group `group2' (GID 1003) ...
Done.
root@saronde:~# tail -n 2 /etc/group
group1:x:1005:
group2:x:1003:
```

Catatan :

Masing-masing group yang telah dibuat memiliki nama group, dan *id* yang berbeda yang ditandai dengan kombinasi bilangan desimal *1005* dan *1003* seperti yang ditunjukkan diatas.

- **groupdel / delgroup**

Pada perintah bash sebelumnya, sudah dijelaskan tentang metode pembuatan group, pada bagian ini akan dijelaskan perintah bash untuk menghapus group, yang sudah dibuat sebelumnya.

groupdel

```
root@saronde:~# groupdel group1
root@saronde:~# tail -n 2 /etc/group
ntop:x:134:
group2:x:1003:
```

delgroup

```
root@saronde:~# delgroup group2
Removing group `group2' ...
Done.
root@saronde:~# tail -n 2 /etc/group
ftp:x:1002:
ntop:x:134:
```

- **usermod**

Merupakan perintah yang berfungsi untuk memodifikasi / melakukan perubahan pengaturan user akun dalam sistem. usermode dapat digunakan untuk beberapa fungsi dan kegunaan, penggunaan fungsi usermod secara detail, dapat diakses melalui *manual page command*, dengan perintah *# man usermod*.

usermod -G

Pada bagian ini akan dijelaskan bagaimana cara untuk memindahkan user (pengguna) untuk tergabung dalam 1 buah group. Misalnya terdapat 4 *user* yang akan digabungkan kedalam 1 buah *group* yang bernama *grouppengguna*.

Tahap 1 : pembuatan 4 buah user dalam sistem.

```
root@saronde:~# useradd pengguna1
root@saronde:~# useradd pengguna2
root@saronde:~# useradd pengguna3
root@saronde:~# useradd pengguna4
root@saronde:~# tail -n 4 /etc/passwd
pengguna1:x:1003:1003::/home/pengguna1:/bin/sh
pengguna2:x:1004:1005::/home/pengguna2:/bin/sh
pengguna3:x:1005:1006::/home/pengguna3:/bin/sh
pengguna4:x:1006:1007::/home/pengguna4:/bin/sh
```

Tahap 2 : pembuatan 1 buah group

```
root@saronde:~# groupadd grouppengguna
root@saronde:~# tail -n 1 /etc/group
grouppengguna:x:1008:
```

Tahap 3 : penyalinan user kedalam group

```
root@saronde:~# usermod -G grouppengguna pengguna1
root@saronde:~# usermod -G grouppengguna pengguna2
root@saronde:~# usermod -G grouppengguna pengguna3
root@saronde:~# usermod -G grouppengguna pengguna4
```

Tahap 4 : pembuktian user yang sudah dibuat telah masuk kedalam group yang diinginkan

```
root@saronde:~# tail -n 1 /etc/group
```

```
grouppengguna:x:1008:pengguna1,pengguna2,pengguna3,pengguna4
```

usermod -L

merupakan parameter yang berfungsi untuk mengunci / menonaktifkan sebuah user, agar tidak bisa digunakan oleh penggunanya.

```
root@saronde:~# usermod -L pengguna1
root@saronde:~# login pengguna1
Password:
Login incorrect
```

usermod -U

merupakan parameter yang berfungsi untuk membuka kunci / mengaktifkan kembali sebuah user agar bisa kembali diakses, jika sebelumnya user tersebut pernah dikunci menggunakan parameter (-L).

```
root@saronde:~# usermod -U pengguna1
root@saronde:~# login pengguna1
Password:
Welcome to Ubuntu 12.04 LTS (GNU/Linux 3.2.0-23-generic-pae i686)

* Documentation:  https://help.ubuntu.com/

The programs included with the Ubuntu system are free software;
the exact distribution terms for .....

$ whoami
pengguna1
```

usermod - -password

Berfungsi untuk melakukan pengaturan password pada sebuah user dalam sistem Linux.

Penggunaan :

```
root@saronde:~# usermode - -password [PASSWORD] [namapengguna]
```

Implementasi :

```
root@saronde:~# usermod --password 12345 pengguna1
```

Beberapa parameter yang digunakan diatas, hanyalah sebagian parameter dari *usermod*, akan tetapi *usermod* masih memiliki banyak parameter lainnya yang bisa diakses melalui menu berikut :

```
root@saronde:~# usermod --help
```

```
Usage: usermod [options] LOGIN
```

Options:

-c, --comment COMMENT	new value of the GECOS field
-d, --home HOME_DIR	new home directory for the user account
-e, --expiredate EXPIRE_DATE	set account expiration date to EXPIRE_DATE
-f, --inactive INACTIVE	set password inactive after expiration to INACTIVE
-g, --gid GROUP	force use GROUP as new primary group
-G, --groups GROUPS	new list of supplementary GROUPS
-a, --append	append the user to the supplemental GROUPS mentioned by the -G option without removing him/her from other groups
-h, --help	display this help message and exit
-l, --login NEW_LOGIN	new value of the login name
-L, --lock	lock the user account
-m, --move-home	move contents of the home directory to the new location (use only with -d)
-o, --non-unique	allow using duplicate (non-unique) UID
-p, --password PASSWORD	use encrypted password for the new password
-s, --shell SHELL	new login shell for the user account
-u, --uid UID	new UID for the user account
-U, --unlock	unlock the user account
-Z, --selinux-user	new SELinux user mapping for the user account

- **passwd**

Merupakan sebuah perintah yang berfungsi untuk melakukan perubahan password (sandi) dari sebuah user (pengguna).

Penggunaan :

```
root@saronde:~# passwd [option/parameter] namauser
```

Implementasi :

```
root@saronde:~# passwd pengguna1
Enter new UNIX password: [password]
Retype new UNIX password: [password]
passwd: password updated successfully
root@saronde:~# login pengguna1
Password: [password]
pengguna1@gruppengguna:~#
```

Selain melakukan perubahan password, `passwd` dapat digunakan untuk menampilkan informasi tentang status sebuah pengguna dengan menggunakan parameter `-S`.

```
pengguna1@gruppengguna:~$ passwd -S
pengguna1 P 02/03/2014 0 99999 7 -1
```

Informasi yang ditampilkan diatas menampilkan informasi terkait status tanggal dan waktu penggunaan sebuah password yang digunakan oleh suatu pengguna,

`passwd` mempunyai banyak parameter dan fungsi, adapun penggunaan parameter fungsi dari perintah `passwd` adalah sebagai berikut :

```
root@saronde:~# passwd --help
Usage: passwd [options] [LOGIN]
Options:
  -a, --all                report password status on all accounts
  -d, --delete             delete the password for the named account
  -e, --expire            force expire the password for the named account
  -h, --help              display this help message and exit
  -k, --keep-tokens       change password only if expired
  -i, --inactive INACTIVE set password inactive after expiration
                          to INACTIVE
  -l, --lock              lock the password of the named account
  -n, --mindays MIN_DAYS  set minimum number of days before password
                          change to MIN_DAYS
  -q, --quiet             quiet mode
  -r, --repository REPOSITORY change password in REPOSITORY repository
  -S, --status            report password status on the named account
  -u, --unlock            unlock the password of the named account
  -w, --warndays WARN_DAYS set expiration warning days to WARN_DAYS
  -x, --maxdays MAX_DAYS set maximum number of days before password
                          change to MAX_DAYS
```

- **sudo (superuser do)**

sudo merupakan sebuah perintah yang berfungsi untuk mengizinkan eksekusi sebuah perintah dengan akses pengguna tertinggi dalam sistem (superuser/root). Sehingga sebuah user biasa dapat mengeksekusi sebuah perintah *bash*, dengan hak akses superuser.

Adapun pemanfaatan perintah *sudo* adalah sebagai berikut :

```
saronde_os@saronde:~$ cat /etc/shadow
cat: /etc/shadow: Permission denied
saronde_os@saronde:~$ whoami
saronde_os
```

Perintah diatas merupakan sebuah aksi uuntuk membaca sebuah file shadow yang hanya dapat diakses oleh superuser (root), tidak bisa diakses oleh user lainnya, oleh karena itu ditampilkan pesan “**Permission denied**”.

Untuk itu jika hendak mengakses (membaca) file */etc/shadow* dengan posisi user biasa (saronde_os), kita bisa menggunakan perintah *sudo*.

```
saronde_os@saronde:~$ sudo cat /etc/shadow
[sudo] password for saronde_os:
root:!:15878:0:99999:7:::
saronde_os:
$6$b1B9d.IN$8GC2Qq168ZUC/78AFxM6Cg06/qWDz7/1QlxbV4CjPRzbN4bma09GxymMLn.mX8jyF.75rNcN3kqL4cW
i6a/041:15878:0:99999:7:::
pengguna1:$6$H9wtaKUe$/qyKE8Ng650bZAtiEPFFgVwnBit3FP2Ix/FCzYJ3FM1g7PZDds5d6jT.f80qFQ973xtBQ
TJdC8xgL0lHS.E0S/:16104:0:99999:7:::
```

Pendefinisian dan penjelasan penggunaan tentang perintah *sudo*, bisa dengan menggunakan opsi *-h*. Sebagai berikut :

```
saronde_os@saronde:~$ sudo -h
```

Bagian XI

Manajemen Jaringan

Linux merupakan sebuah sistem operasi yang awalnya dikembangkan dengan sistem keamanan yang sangat tinggi, dominan semua server dalam instansi pemerintah maupun swasta menggunakan linux sebagai sistem keamanan data, website dan data center secara keseluruhan.

Banyak distribusi linux handal yang dapat dikembangkan untuk kepentingan keamanan sebuah server data dalam jaringan komputer, sistem operasi berbasis linux tersebut adalah redhat, debian, centos, ubuntu, ipcop, ipfire, dll.

Pada bagian ini akan dijelaskan beberapa perintah bash yang banyak digunakan dalam administratif jaringan, adalah sebagai berikut :

- **ping** – mengirimkan paket ICMP ke host/network lainnya.
- **traceroute** – mencetak rute paket yang berjalan dalam sebuah jaringan.
- **netstat** – mencetak informasi koneksi jaringan, tabel routing, statistik interface, informasi koneksi masquerade, dan single/multicast.
- **ftp** – aplikasi untuk menjalankan dan menggunakan layanan ftp server
- **nmap** – aplikasi untuk menampilkan informasi tentang sebuah host dan port dalam jaringan
- **ssh** – aplikasi client untuk secure shell login
- **scp** – aplikasi untuk melakukan transfer file dengan media jaringan
- **wget** – aplikasi download manager berbasis terminal
- **hping** – pengembangan aplikasi dari ping, dengan banyak fitur tambahan
- **ifconfig** – menampilkan informasi tentang interface jaringan lokal
- **iwconfig** – menampilkan informasi tentang interface jaringan wireless

- **ping**

ping merupakan sebuah tool jaringan yang paling ampuh yang paling banyak digunakan disemua platform sistem operasi baik linux maupun non-linux, cara kerja ping yaitu dengan mengirimkan paket ICMP ke host/network lainnya, secara otomatis host tujuan akan merespon paket ICMP yang dikirimkan, sehingga berdasarkan pada hal tersebut kita bisa mengetahui bahwa komputer yang digunakan sudah terkoneksi dengan host yang lainnya / yang dituju.

Adapun penggunaan perintah ping dapat dengan mengetikkan format berikut :

```
saronde_os@saronde:~$ ping [ipv4]  
saronde_os@saronde:~$ ping [domain]
```

Untuk implementasi penggunaan perintah ping adalah sebagai berikut :

```
saronde_os@saronde:~$ ping 10.12.114.30  
PING 10.12.114.30 (10.12.114.30) 56(84) bytes of data.  
64 bytes from 10.12.114.30: icmp_req=1 ttl=255 time=0.421 ms  
64 bytes from 10.12.114.30: icmp_req=2 ttl=255 time=0.411 ms  
64 bytes from 10.12.114.30: icmp_req=3 ttl=255 time=0.424 ms  
--- 10.12.114.30 ping statistics ---  
3 packets transmitted, 3 received, 0% packet loss, time 5998ms  
rtt min/avg/max/mdev = 0.368/0.410/0.424/0.017 ms
```

```
saronde_os@saronde:~$ ping google.com  
PING google.com (118.98.106.99) 56(84) bytes of data.  
64 bytes from 99.subnet118-98-106.astinet.telkom.net.id (118.98.106.99): icmp_req=1 ttl=59  
time=59.2 ms  
64 bytes from 99.subnet118-98-106.astinet.telkom.net.id (118.98.106.99): icmp_req=2 ttl=59  
time=62.7 ms  
64 bytes from 99.subnet118-98-106.astinet.telkom.net.id (118.98.106.99): icmp_req=3 ttl=59  
time=76.6 ms  
--- google.com ping statistics ---  
3 packets transmitted, 3 received, 0% packet loss, time 4005ms  
rtt min/avg/max/mdev = 59.264/70.555/86.127/9.742 ms
```

Hasil output yang ditampilkan berarti proses ping atau hubungan koneksi antara satu host dengan host yang lain berjalan dengan lancar ditandai dengan output yang ditampilkan diatas, selain itu diakhir proses ping ditampilkan resume dari proses ping, dimana paket yang gagal berjumlah 0%.

proses ping akan menghasilkan output yang berbeda bergantung pada kondisi jaringan / host yang dituju, hasil output dari lainnya adalah sebagai berikut :

Output berikut akan ditampilkan, jika sistem jaringan secara tiba-tiba nonaktif ketika proses koneksi yang sebelumnya sudah dilakukan.

```
saronde_os@saronde:~$ ping 10.12.115.30
PING 10.12.115.30 (10.12.115.30) 56(84) bytes of data.
ping: sendmsg: Network is unreachable
ping: sendmsg: Network is unreachable
ping: sendmsg: Network is unreachable
ping: sendmsg: Network is unreachable
--- 10.12.115.30 ping statistics ---
47 packets transmitted, 0 received, 100% packet loss, time 46359ms
```

Output berikut akan ditampilkan ketika interface jaringan tidak terkoneksi dengan jaringan.

```
saronde_os@saronde:~$ ping 10.12.115.30
connect: Network is unreachable
```

Output berikut akan ditampilkan jika proses koneksi ke host yang dituju gagal atau host yang dituju sedang down.

```
saronde_os@saronde:~$ ping 10.12.114.30
PING 10.12.114.30 (10.12.114.30) 56(84) bytes of data.
From 10.12.114.28 icmp_seq=3 Destination Host Unreachable
From 10.12.114.28 icmp_seq=5 Destination Host Unreachable
From 10.12.114.28 icmp_seq=6 Destination Host Unreachable
From 10.12.114.28 icmp_seq=9 Destination Host Unreachable
From 10.12.114.28 icmp_seq=10 Destination Host Unreachable
From 10.12.114.28 icmp_seq=11 Destination Host Unreachable
From 10.12.114.28 icmp_seq=12 Destination Host Unreachable
From 10.12.114.28 icmp_seq=14 Destination Host Unreachable
```

```
From 10.12.114.28 icmp_seq=15 Destination Host Unreachable
--- 10.12.114.30 ping statistics ---
15 packets transmitted, 0 received, +9 errors, 100% packet loss, time 14079ms
pipe 3
```

- **traceroute**

Merupakan sebuah perintah yang berfungsi untuk mencetak rute paket (hop) dalam jaringan yang akan dilalui untuk mencapai sebuah host yang dituju.

Format penggunaan perintah traceroute adalah sebagai berikut :

```
saronde_os@saronde:~$ traceroute [ip_address/domain]
```

Berikut adalah implementasi penggunaan perintah traceroute :

```
saronde_os@saronde:~$ traceroute kaskus.co.id
traceroute to kaskus.co.id (103.6.117.3), 30 hops max, 60 byte packets
 1. 10.12.114.30 (10.12.114.30) 0.529 ms 0.600 ms 0.585 ms
 2. 192.178.1.33 (192.178.1.33) 4.403 ms 4.396 ms 4.381 ms
 3. 10.10.1.1 (10.10.1.1) 0.780 ms 0.767 ms 0.751 ms
 4. 192.168.249.17 (192.168.249.17) 21.917 ms 21.892 ms 21.855 ms
 5. 61.94.115.193 (61.94.115.193) 155.203 ms 158.758 ms 158.951 ms
 6. 118.98.15.53 (118.98.15.53) 154.924 ms 153.730 ms 123.329 ms
 7. 181.subnet118-98-53.astinet.telkom.net.id (118.98.53.181) 132.377 ms 132.171 ms
   132.165 ms
 8. 61.5.116.173 (61.5.116.173) 138.635 ms 61.94.117.225 (61.94.117.225) 128.669 ms
   128.624 ms
 9. telkomnet.openixp.net (218.100.27.29) 122.464 ms 122.477 ms 122.462 ms
10. giga-0-0.openixp.net (218.100.27.129) 126.010 ms 126.002 ms 125.752 ms
11. kaskus.openixp.net (218.100.36.85) 123.052 ms 123.202 ms 123.111 ms
12. ksk.lbl02-3.id.kaskus.com (103.6.117.3) 127.606 ms 127.543 ms 127.534 ms
```

Seperti yang ditampilkan pada output diatas, proses rote paket data dari host awal (1), ketujuan (12) melewati 12 rute.

- **Netstat**

Mencetak informasi koneksi jaringan, tabel routing, statistik interface, informasi koneksi masquerade, dan single/multicast.

Penggunaan netstat dapat dengan mengikuti format perintah berikut :

```
saronde_os@saronde:~$ netstat [parameter]
```

Implementasi netstat dapat dijelaskan sebagai berikut :

- **netstat [tanpa_operator]**

yang menampilkan informasi tentang proses koneksi antara operator dan socket yang berjalan dalam sistem.

```
saronde_os@saronde:~$ netstat
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
Active UNIX domain sockets (w/o servers)
Proto RefCnt Flags           Type           State           I-Node  Path
unix  2      [ ]           DGRAM                    121647  /dev/log
unix  3      [ ]           STREAM           CONNECTED       96870   @/tmp/.X11-unix/X0
unix  3      [ ]           STREAM           CONNECTED       97732
unix  3      [ ]           STREAM           CONNECTED       96868   @/tmp/.X11-unix/X0
unix  3      [ ]           STREAM           CONNECTED       97731
unix  3      [ ]           STREAM           CONNECTED       96865   @/tmp/.X11-unix/X0
```

- **netstat -ie**

berfungsi untuk menampilkan informasi tentang interface jaringan (ethernet / wireless) secara lengkap secara extended

```
saronde_os@saronde:~$ netstat -ie
Kernel Interface table
eth0      Link encap:Ethernet  HWaddr 00:1f:16:c6:81:3b
          UP BROADCAST MULTICAST  MTU:1500  Metric:1
```

```
RX packets:0 errors:0 dropped:0 overruns:0 frame:0
TX packets:0 errors:0 dropped:0 overruns:0 carrier:1
collisions:0 txqueuelen:1000
RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)
Interrupt:44
.....
```

- **netstat -r**

menampilkan informasi tentang tabel rute jaringan dalam sebuah host (komputer).

```
saronde_os@saronde:~$ netstat -re
```

```
Kernel IP routing table
```

Destination	Gateway	Genmask	Flags	Metric	Ref	Use	Iface
10.42.0.0	*	255.255.255.0	U	2	0	0	wlan0
link-local	*	255.255.0.0	U	1000	0	0	wlan0

Penggunaan parameter untuk perintah netstat dapat diakses secara lengkap menggunakan perintah *netstat -h*.

- **nmap**

Sebuah tools / aplikasi yang berfungsi untuk menampilkan informasi tentang sebuah host dan port dalam jaringan. Penggunaan *nmap* dapat mengikuti format berikut :

```
saronde_os@saronde:~$ nmap [parameter] [host]
```

Adapun implementasi *nmap* adalah sebagai berikut :

```
saronde_os@saronde:~$ nmap saronde.org
```

```
Starting Nmap 5.21 ( http://nmap.org ) at 2014-02-03 21:31 CIT
```

```
Nmap scan report for saronde (127.0.1.1)
```

```
Host is up (0.0010s latency).
```

```
Not shown: 991 closed ports
```

```
PORT      STATE SERVICE
```

```
21/tcp    open  ftp
```

```
22/tcp    open  ssh
```

```
80/tcp    open  http
```

```
Nmap done: 1 IP address (1 host up) scanned in 0.16 seconds
```

```
saronde_os@saronde:~$ sudo nmap 10.42.0.1
Starting Nmap 5.21 ( http://nmap.org ) at 2014-02-03 21:31 CIT
Nmap scan report for 10.42.0.1
Host is up (0.000027s latency).
Not shown: 990 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
53/tcp    open  domain
80/tcp    open  http
Nmap done: 1 IP address (1 host up) scanned in 13.17 seconds
```

Seperti yang sudah dijelaskan sebelumnya, output dari perintah nmap akan menampilkan informasi tentang status sebuah port jaringan yang sedang terbuka dan layanan yang digunakan dalam port tersebut. Adapun penggunaan fungsi nmap dengan parameter secara detail dapat diakses dengan perintah berikut : *nmap -h*.

- **ssh**

ssh atau bisa disingkat secure shell merupakan sebuah sistem shell login berbasis remote login, yang memungkinkan seorang pengguna (sysadmin) untuk mengakses sebuah host (komputer) melalui dengan media jaringan. Cara kerja ssh lebih aman dibandingkan remote login lainnya seperti telnet, ssh menggunakan sistem kunci enkripsi fingerprint (handshake) untuk bisa saling terkoneksi antara host dan server remote.

Sebelumnya pastikan port ssh:22 sedangkan terbuka, menggunakan perintah nmap. Jika service *ssh:22* sudah berjalan diserver, selanjutnya penggunaan fungsi ssh adalah sebagai berikut :

```
saronde_os@saronde:~$ ssh namauser@ipaddress
saronde_os@saronde:~$ ssh namauser@domain
```

Implementasi penggunaan perintah *ssh* adalah sebagai berikut :

```
root@saronde:~# ssh saronde_os@10.42.0.1
The authenticity of host '10.42.0.1 (10.42.0.1)' can't be established.
ECDSA key fingerprint is 8c:50:be:79:d5:ae:d8:4b:d3:0e:5d:ae:cc:19:f0:a6.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.42.0.1' (ECDSA) to the list of known hosts.
saronde_os@10.42.0.1's password:
Last login: Wed Jan 22 08:01:42 2014
saronde_os@saronde:~$ whoami
saronde_os
```

```
root@saronde:~# ssh saronde_os@saronde .org
The authenticity of host 'saronde (127.0.1.1)' can't be established.
ECDSA key fingerprint is 8c:50:be:79:d5:ae:d8:4b:d3:0e:5d:ae:cc:19:f0:a6.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'saronde' (ECDSA) to the list of known hosts.
saronde_os@saronde's password:
Welcome to Ubuntu 12.04 LTS (GNU/Linux 3.2.0-23-generic-pae i686)
saronde_os@saronde:~$ whoami
saronde_os
```

Adapun penggunaan fungsi dan parameter tentang tool *ssh*, dapat dilihat menggunakan perintah : *ssh -h*.

- **ifconfig**

Merupakan sebuah perintah yang berfungsi untuk menampilkan informasi tentang sebuah interface jaringan, selain menampilkan untuk menampilkan informasi, *ifconfig* dapat juga digunakan untuk mengaktifkan / menonaktifkan sebuah interface dan melakukan pengaturan *ipaddress* secara instan.

Adapun penggunaan perintah *ifconfig* adalah sebagai berikut :

```
root@saronde:~# ifconfig
eth0      Link encap:Ethernet  HWaddr 00:1f:16:c6:81:3b
          UP BROADCAST MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:1
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
          Interrupt:44
```

```
lo          Link encap:Local Loopback
           inet addr:127.0.0.1  Mask:255.0.0.0
           inet6 addr: ::1/128 Scope:Host
           UP LOOPBACK RUNNING  MTU:16436  Metric:1
           RX packets:9522 errors:0 dropped:0 overruns:0 frame:0
           TX packets:9522 errors:0 dropped:0 overruns:0 carrier:0
           collisions:0 txqueuelen:0
           RX bytes:551730 (551.7 KB)  TX bytes:551730 (551.7 KB)

wlan0      Link encap:Ethernet  HWaddr 0c:60:76:17:1a:0c
           inet addr:10.42.0.1  Bcast:10.42.0.255  Mask:255.255.255.0
           inet6 addr: fe80::e60:76ff:fe17:1a0c/64 Scope:Link
           UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
           RX packets:0 errors:0 dropped:0 overruns:0 frame:0
           TX packets:253 errors:0 dropped:0 overruns:0 carrier:0
           collisions:0 txqueuelen:1000
           RX bytes:0 (0.0 B)  TX bytes:54863 (54.8 KB)
```

Untuk menampilkan informasi tentang interface dapat mengikuti format perintah berikut :

```
root@saronde:~# ifconfig eth0
eth0      Link encap:Ethernet  HWaddr 00:1f:16:c6:81:3b
           UP BROADCAST MULTICAST  MTU:1500  Metric:1
           RX packets:0 errors:0 dropped:0 overruns:0 frame:0
           TX packets:0 errors:0 dropped:0 overruns:0 carrier:1
           collisions:0 txqueuelen:1000
           RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
           Interrupt:44
```

Untuk mengaktifkan (up) dan menonaktifkan (down) sebuah interface jaringan, dapat dengan menggunakan perintah berikut informasi tentang interface dapat mengikuti format perintah berikut :

```
root@saronde:~# ifconfig eth0 up
root@saronde:~# ifconfig eth0 down
```

Untuk melakukan konfigurasi ip address untuk sebuah ipaddress secara instan, dapat mengikuti format perintah berikut :

```
root@saronde:~# ifconfig wlan0 192.168.1.1/24
root@saronde:~# ifconfig wlan0
wlan0      Link encap:Ethernet  HWaddr 0c:60:76:17:1a:0c
           inet addr:192.168.1.1  Bcast:192.168.1.255  Mask:255.255.255.0
           inet6 addr: fe80::e60:76ff:fe17:1a0c/64 Scope:Link
           UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
           RX packets:0 errors:0 dropped:0 overruns:0 frame:0
           TX packets:265 errors:0 dropped:0 overruns:0 carrier:0
           collisions:0 txqueuelen:1000
           RX bytes:0 (0.0 B)  TX bytes:59143 (59.1 KB)
```

- **iwconfig**

Berfungsi menampilkan informasi tentang interface jaringan wireless, dan melakukan konfigurasi pada perangkat wireless. Penggunaan perintah *iwconfig* dapat ditunjukkan pada format berikut :

```
root@saronde:~# iwconfig
wlan0      IEEE 802.11bgn  ESSID:"saronde"
          Mode:Ad-Hoc  Frequency:2.412 GHz  Cell: DA:01:24:12:53:F2
          Tx-Power=16 dBm
          Retry  long limit:7   RTS thr:off   Fragment thr:off
          Encryption key:off
          Power Management:off
```

Selain menampilkan informasi tentang wireless, *iwconfig* dapat digunakan untuk berbagai fungsi, diantaranya melakukan konfigurasi ESSID, frekuensi wireless, sistem keamanan, power management, dll. Penggunaan detail tentang *iwconfig* dapat dilihat menggunakan perintah berikut : *iwconfig -h* dan *man iwconfig*.

- **scp**

Aplikasi yang berfungsi untuk melakukan transfer file melalui media jaringan, scp menggunakan layanan / jalur dari port ssh untuk mengirimkan paket bit untuk melintasi jaringan secara berurutan. Penggunaan perintah scp membutuhkan dua buah host (komputer). Dimana satu host akan bertindak sebagai pengirim dan host lain bertugas sebagai penerima. Penggunaan perintah scp dapat mengikuti format berikut :

```
root@saronde:~# scp [data] namauser@ipaddress:/lokasidirektori
root@saronde:~# scp [data] namauser@domain:/lokasidirektori
```

Implementasi penggunaan perintah scp adalah sebagai berikut :

Pengirim

```
tkj:~# cat > fileTKJ
ini filedari serverpengirim (tkj:192.168.1.2)
tkj:~# scp fileTKJ saronde_os@192.168.1.1:/home/saronde_os
saronde_os@192.168.1.1's password:
fileTKJ                                100%  21    0.0KB/s  00:00
```

Penerima

```
saronde_os@saronde:~$ ls -l fileTKJ
-rw-r--r-- 1 saronde_os saronde_os 21 Feb  3 22:14 fileTKJ
saronde_os@saronde:~$ cat fileTKJ
ini filedari serverpengirim (tkj:192.168.1.2)
```

- **wget**

Merupakan sebuah aplikasi download manager berbasis terminal, yang dapat digunakan untuk mendownload sebuah file / halaman HTML dari server. Penggunaan wget adalah sebagai berikut :

```
saronde_os@saronde:~$ wget 192.168.1.2
--2014-02-03 22:18:14-- http://192.168.1.2/
Connecting to 192.168.1.2:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 67 [text/html]
Saving to: `index.html'
100%[=====] 67          --.-K/s   in 0s
2014-02-03 22:18:14 (3,66 MB/s) - `index.html' saved [67/67]
```

```
saronde_os@saronde:~$ wget saronde.org/Joomla_3.0.1-Stable-Full_Package.zip
--2014-02-03 22:19:56-- http://saronde.org/Joomla_3.0.1-Stable-Full_Package.zip
Connecting to saronde.org:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 7844212 (7,5M) [application/zip]
Saving to: `Joomla_3.0.1-Stable-Full_Package.zip'

100%[=====] 7.844.212   17,3M/s   in 0,4s

2014-02-03 22:19:56 (17,3 MB/s) - `Joomla_3.0.1-Stable-Full_Package.zip' saved
[7844212/7844212]
```

Adapun opsi parameter yang dapat digunakan untuk perintah *wget*, dapat dengan menggunakan perintah *wget -h*.

Penutup

Alhamdulillah, buku panduan ini dapat disusun sesuai dengan target yang ditentukan, akan tetapi kami (tim penulis) menyadari dalam buku panduan ini masih terdapat banyak kekurangan, kesalahan dan kekeliruan baik dalam penjelasan, penggunaan tata bahasa, dan kesalahan-kesalahan yang lainnya, karena terlepas dari semua itu kami tim penulis hanya seorang manusia biasa yang terus belajar dan berkarya untuk membuat sesuatu yang bermanfaat untuk banyak orang.

Untuk itu kami mohon dukungan / kontribusi dengan memberikan kritik dan saran terkait penyusunan buku panduan ini agar menjadi lebih baik lagi dan dapat terlihat “*sempurna*”.

Akhir Kata, dan terus belajar, berkarya bagi generasi muda untuk INDONESIA Tercinta...!!!

Wassalamualaikum Warahmatullahi Wabarakatu

Referensi Pustaka

- *Shoots W.E Jr.- The Linux Command Line A Complete Introduction - No_Strach_Press (2012)*
- *Kelompok Studi Linux UNG, Modul Panduan penggunaan Linux “Kita bisa dengan OpenSource” (2011)*
- *Kelompok Studi Linux UNG, Modul Panduan penggunaan Linux “Ayo Migrasi ke Linux” (2012)*
- *Catatan group Forum diskusi Kelompok Studi Linux UNG (sosial media : <http://facebook.com/groups/kslung>) (2009 - 2014)*