Linux Partition HOWTO

Kristan Koehntopp <kris@koehntopp.de> diterjemahkan oleh Y. Heryadi <yheryadi@necnusa.co.id> V2.4, 3 November 1997, terj. 7 Des 1998

Linux Mini-HOWTO ini mengajarkan Anda bagaimana merancang dan membuat layout disk space untuk system Linux. Dokumen ini mengulas perangkat-keras disk, partisi, ukuran serta pertimbangan penempatan area swap, file system, tipe-tipe file sistem, dan topik-topik lain yang berhubungan dengannya. Tujuan penulisan dokumen ini adalah memberikan Anda latar belakang pengetahuan, tetapi bukan prosedurnya.

Contents

1 Pendahuluan

1.1 Dokumen apakah ini?

Dokumen ini merupakan sebuah Mini-HOWTO Linux. Sebuah Mini-HOWTO adalah tulisan yang menjelaskan hal-hal yang berkaitan dengan Instalasi Linux dan Maintenance dengan gaya tutorial. Tulisan ini disebut mini, karena isi tulisan maupun topik yang dibicarakan terlalu sempit bagi sebuah HOWTO yang sesungguhnya apalagi sebuah buku. Sebuah HOWTO bukan merupakan referensi, itulah tujuan dari dibuatnya halaman-halaman manual.

1.2 Apakah isi dokumen ini, dan dokumentasi-dokumentasi HOWTO yang berkaitan?

Mini-HOWTO ini mengajarkan Anda cara untuk merancang maupun membuat layout disk space untuk sistem Linux. Tulisan ini mengulas perangkat-keras disk, partisi, ukuran serta pertimbangan penempatan area swap, file system, tipe-tipe file sistem serta topik-topik yang berhubungan dengannya. Tujuan penulisan ini adalah untuk mengajarkan dasar pengetahuan, sehingga yang didiskusikan dalam tulisan ini adalah prinsip-prinsip utama dan bukan alatnya.

Secara ideal, dokumen ini harus dibaca sebelum Anda pertama kali melakukan instalasi, tetapi hal ini sulit dilakukan bagi banyak orang. Para pendatang baru biasanya memiliki problem lain selain masalah membuat optimisasi terhadap layout disk. Anda mungkin termasuk salah seorang yang baru selesai melakukan instalasi Linux dan sedang memikirkan cara untuk mengoptimalkan instalasi yang telah dilakukan atau cara menghindari terulangnya beberapa kesalahan perhitungan dimasa datang. Jangan terkejut jika sesudah selesai membaca tulisan ini Anda berkeinginan untuk menghapus dan membangun kembali instalasi yang Anda telah lakukan.

Ruang lingkup dari Mini-HOWTO ini dibatasi hanya pada perancangan dan pembuatan layout dari disk space. Sedangkan, penggunaan `fdisk`, LILO, `mke2fs`, ataupun program-program backup tidak termasuk kedalamnya. Hal-hal tsb sudah dibahas dalam beberapa HOWTO lainnya. Anda dipersilahkan untuk melihat index Linux HOWTO untuk mendapatkan informasi terakhir mengenai Linux HOWTO. Didalam indeks tsb Anda akan mendapatkan instruksi mengenai cara untuk mendapatkan dokumen-dokumen HOWTO tsb.

Untuk mengetahui cara membuat perkiraan mengenai kebutuhan ukuran dan kecepatan bagi bagian-bagian file sistem, silahkan membaca dokumen *Linux Multiple Disks Layout mini-HOWTO*, yang ditulis oleh Gjoen Stein (gjoen@nyx.net)

Untuk mendapatkan petunjuk dan pertimbangan mengenai disk yang memiliki silinder lebih dari 1024, silahkan membaca dokumen *Linux Large Disk mini-HOWTO*, yang ditulis oleh Andries Brouwer (aeb@cwi.nl)

Untuk mendapatkan petunjuk mengenai pembatasan dalam penggunaan disk space bagi setiap user (*quota*), silahkan membaca *Linux Quota mini-HOWTO* yang ditulis oleh Albert M.C. Tam (bertie@scn.org)

Saat ini, dokumentasi umum mengenai backup disk belum tersedia, namun terdapat beberapa dokumen yang memberikan rujukan ke berbagai solusi backup yang spesifik. Silahkan membaca dokumen *Linux ADSM backup Mini-HOWTO* yang ditulis oleh Thomas Koenig (thomas.Koenig@ciw.uni-karlsruhe.de) mengenai instruksi bagi pengintegrasian Linux dengan environment backup IBM ADSM. Juga silahkan membaca *Linux backup with MSDOS mini-HOWTO* yang ditulis oleh Christopher Neufeld (neufeld@physics.utoronto.ca) untuk informasi mengenai backup Linux yang dikendalikan dengan MSDOS.

Untuk mendapatkan instruksi bagi penulisan dan penyerahan sebuah dokumen HOWTO, silahkan membaca Indeks Linux HOWTO yang ditulis oleh Tim Bynum (linux-howto@sunsite.unc.edu)

Melihat-lihat ke dalam direktori `/usr/src/Documentation` dapat bersifat instruktif. Bacalah file-file `ide.txt` dan `scsi.txt` untuk mendapatkan latar-belakang informasi mengenai sifat dari disk driver, dan juga silahkan lihat subdirektori dari / file sistem.

2 Apakah sebuah partisi itu?

Tidak lama sesudah hard disk PC ditemukan, orang kemudian banyak yang ingin meng-install beberapa operating sistem meskipun PC mereka hanya memiliki satu hard disk. Sehingga diperlukan suatu mekanisma untuk membagi sebuah fisik hard disk menjadi beberapa *logical disk*. Jadi apa yang disebut sebuah partisi adalah: suatu bagian yang bersambung (*contiguous*) dari blok-blok hard disk yang diperlakukan sebagai suatu hard disk oleh sistem operasi.

Partisi tidak boleh saling tumpang-tindih (*overlap*), alasannya sederhana: sebuah sistem operasi pasti tidak akan dapat bekerja apabila sebuah sistem operasi lain di-install pada komputer yang sama dan merubah informasi penting yang diakibatkan tumpang-tindihnya partisi. Begitu pula, dua partisi yang berdekatan tidak boleh memiliki selang antara. Meskipun susunan partisi tsb tidak membahayakan, namun dengan membuat selang antara dua partisi berarti Anda telah menyia-nyiakan disk space.

Sebuah disk tidak harus dipartisi secara lengkap. Anda dapat memutuskan untuk meninggalkan beberapa ruang pada disk Anda yang tidak digunakan oleh sistem operasi yang Anda install. Sesudah mengetahui instalasi mana yang akan Anda gunakan seterusnya, Anda dapat melakukan partisi ruang disk space yang tersisa tadi dan menempatkan file sistem ke dalamnya.

Partisi tidak dapat dipindahkan maupun dirubah ukurannya tanpa merusak file sistem yang terdapat di dalamnya. Sehingga melakukan partisi ulang biasanya meliputi pekerjaan mem*backup* dan me*restore* seluruh file sistem yang terpengaruh oleh proses ini. Sesungguhnya rusaknya seluruh isi disk saat mempartisi ulang merupakan suatu hal yang umum terjadi, sehingga sebelum Anda mulai menggunakan fdisk, Anda harus melakukan backup keseluruhan dari disk dalam komputer tersebut.

Beberapa partisi dengan tipe file sistem tertentu dapat dipecah menjadi dua bagian partisi tanpa menyebabkan hilangnya data (jika Anda beruntung). Misalnya, dengan program yang bernama *fips* , sebuah partisi MS-DOS dapat dipecah menjadi dua partisi untuk memberi tempat bagi instalasi Linux tanpa harus melakukan reinstall MS-DOS. Anda tentu tidak akan melakukan hal tsb sebelum seluruh data dari komputer Anda dibackup, ya kan?

2.1 Backup itu penting

Tape adalah teman Anda untuk mem-backup. Tape itu dapat digunakan secara cepat, reliable, dan mudah digunakan; sehingga Anda dapat melakukan backup yang kadang, lebih disukai untuk dilakukan secara otomatis, dan tanpa menimbulkan kesulitan.

Bersiaplah, penulis akan membicarakan tape tetapi tidak akan membicarakan ftape, sesuatu yang dikendalikan disk controller. Pertimbangkan untuk membeli SCSI: Linux secara native telah men-support SCSI. Dengan Linux, Anda tidak perlu untuk memasukkan ASPI driver, dan tidak akan kehilangan HMA yang berharga. Sesudah host adapter SCSI ter-install, Anda tinggal memasang tambahan disk, tape, atau CD-ROM kedalamnya. Anda tidak akan bersusah-payah lagi dengan I/O address, IRQ atau Master/Slave serta penyesuaian PIO-level.

Tambahan: host adapter SCSI yang tepat akan memberikan Anda performance I/O yang tinggi tanpa banyak membebani CPU. Meskipun dengan aktifitas disk yang tinggi Anda akan mendapatkan waktu respons yang baik. Jika Anda merencanakan untuk menggunakan sistem Linux sebagai USENET news feed utama, atau jika Anda akan memasuki bisnis ISP, jangan pernah berfikir untuk menjalankan sebuah sistem tanpa SCSI.

2.2 Bilangan Device dan Nama Device

Sejak awal, Jumlah partisi dalam sistem berbasis Intel sudah dibatasi: Tabel partisi asli yang diinstall sebagai bagian dari boot sektor hanya menyediakan tempat bagi 4 (empat) entri partisi. Partisi-partisi tsb sekarang disebut partisi primer. Sesudah jelas bahwa orang membutuhkan lebih banyak partisi maka partisi logical dikembangkan. Jumlah partisi logical tidak dibatasi: Setiap partisi logical berisi sebuah pointer ke partisi logical berikutnya, sehingga secara potensial Anda dapat memiliki rantai entri partisi yang tidak terbatas.

Untuk alasan kompatibilitas, space yang dipakai oleh seluruh partisi logical harus diperhitungkan. Jika Anda menggunakan partisi logical, sebuah partisi primer ditandai sebagai partisi *extended*, blok awal dan blok akhirnya ditandai sebagai area yang digunakan oleh partisi logical Anda. Hal ini menyebabkan space yang digunakan oleh semua partisi logical harus bersambung tanpa antara (*contiguous*). Jumlah partisi extended hanya ada satu: program `fdisk` tidak akan membuat lebih dari satu partisi extended.

Linux hanya dapat menangani partisi dengan jumlah yang terbatas pada setiap drivenya. Sehingga dalam Linux Anda dapat memiliki 4 partisi primer (jika Anda menggunakan partisi logical maka 3 diantara partisi primer tsb dapat Anda gunakan), dan maksimum jumlah keseluruhan adalah 15 partisi bagi disk SCSI (atau berjumlah keseluruhan 63 bagi IDE disk).

Dalam Linux, partisi-partisi dinyatakan dengan file device. Sebuah file device adalah sebuah file dengan tipe c (untuk "character device", yaitu device yang tidak menggunakan buffer cache) atau b (untuk "blok device", device yang menggunakan buffer cache). Di dalam Linux, semua disk dinyatakan hanya sebagai blok device. Berlainan dengan beragam Unix yang lain, Linux tidak memberikan versi *raw character* bagi disk maupun partisi-partisinya.

Suatu hal yang penting bagi suatu file device adalah bilangan major dan minor dari device, yang ditampilkan sebagai pengganti ukuran file.

```
$ ls -l /dev/had
brw-rw----  1  root    disk      3,    0   Jul  18   1994   /dev/had
                                 ^     ^
                                       |     bilangan minor dari device
                                 |           bilangan major dari device
```

Ketika mengakses sebuah file device, bilangan major akan menentukan device driver mana yang dipanggil untuk melakukan operasi input/output. Pemanggilan device driver ini dilakukan dengan bilangan minor sebagai parameternya, dan interpretasi dari bilangan minor tsb tergantung kepada device driver tadi. Dokumentasi device driver biasanya menjelaskan bagaimana device driver menggunakan bilangan minor. Bagi disk IDE, dokumentasi ini terdapat pada `/usr/src/linux/Documentation/ide.txt`. Bagi disk SCSI, orang akan menduga bahwa dokumentasi tsb akan berada di dalam `/usr/src/linux/Documentation/scsi.txt`, tetapi dugaan itu salah. Orang harus memeriksa driver source untuk memastikannya (`/usr/src/linux/driver/scsi/sd.c:184-196`). Kita beruntung memiliki daftar Peter Anvin yang berisi bilangan serta nama device di dalam file `/usr/src/linux/Documentation/device.txt`; Silahkan lihat untuk blok device, major 3,22,33,34 untuk disk IDE serta major 8 untuk disk SCSI. Bilangan major dan minor masing-masing sebesar 1 byte dan itulah sebabnya mengapa jumlah partisi pada setiap disk dibatasi.

Sesuai konvensi, suatu file device memiliki nama tertentu dan program-program sistem mengetahui nama-nama tsb karena telah ter*compile* kedalamnya. Program-program itu menginginkan disk IDE diberi nama `/dev/hd*` sedangkan disk SCSI diberi nama `/dev/sd*`. Disk-disk dinamai dengan a, b, c, dst, sehingga

`/dev/hda` adalah disk IDE Anda yang pertama dan `/dev/sda` adalah disk SCSI Anda yang pertama. Kedua device tsb mewakili keseluruhan disk dimulai dari blok satu. Penulisan kedalam disk dengan alat yang salah akan merusak master boot loader dan tabel partisi dari disk tsb, menyebabkan semua data dalam disk tsb tidak terbaca atau membuat sistem Anda tidak dapat di-boot. Fahamilah dahulu apa-apa yang Anda akan lakukan, dan sekali lagi, lakukanlah backup sebelum Anda melakukannya.

Partisi primer dalam sebuah disk ditandai dengan angka 1, 2, 3, dan 4. Sehingga `/dev/hda1` adalah partisi primer pertama dari disk IDE pertama, dst. Partisi logical memiliki bilangan 5 dan seterusnya, sehingga `/dev/sdb5` adalah partisi logical pertama dari disk SCSI kedua.

Setiap entri partisi memiliki alamat blok awal dan alamat blok akhir dan juga tipenya. Tipe adalah sebuah kode bilangan (satu byte) yang menAndai suatu partisi dalam suatu sistem operasi. Demi kepentingan konsultan komputer, kode tipe partisi tidaklah khas (unik), sehingga dimungkinkan adanya dua sistem operasi yang menggunakan satu kode tipe yang sama.

Linux menggunakan kode tipe 0x82 bagi partisi swap, dan 0x83 bagi file sistem "native" (untuk hampir semua dari Anda itu berarti ext2). File sistem yang dulu pernah populer namun sekarang telah ketinggalan yaitu Linux/Minix, menggunakan kode tipe 0x81 bagi partisinya. OS/2 menAndai partisinya dengan tipe 0x07 seperti halnya NTFS dari Windows NT. MS-DOS memberikan beberpa kode tipe bagi berbagai jenis file sistem FAT: 0x01, 0x04, dan 0x06. DR-DOS menggunakan 0x81 untuk menAndai partisi FAT yang terproteksi, menyebabkan adanya konflik tipe dengan Linux/Minix saat itu, namun baik DR-DOS maupun Linux/Minix sudah tidak banyak lagi digunakan. Partisi extended yang digunakan sebagai kontainer bagi partisi logical memiliki kode tipe 0x05.

Partisi-partisi dibuat dan dihapus menggunakan program fdisk. Setiap program sistem operasi dilengkapi dengan sebuah fdisk, dan secara tradisi di setiap sistem operasi program dinamai `fdisk` (atau `FDISK.EXE`). Beberapa fdisk, termasuk yang terdapat di dalam DOS, memiliki keterbatasan jika menemui partisi sistem operasi lain. Keterbatasan-keterbatasan tsb meliputi ketidak-mampuan untuk melakukan apapun jika kode tipenya tidak dikenal, menangani silinder dengan nomor diatas 1024, dan membuat bahkan memahami partisi yang batas akhirnya diluar batas silinder. Misalnya, fdisk dari MS-DOS tidak dapat menghapus partisi NTFS, fdisk dari OS/2 diketahui dapat "mengoreksi" partisi yang dibuat dengan fdisk dari Linux yang bats akhirnya diluar batas silinder. Demikian pula, fdisk dari DOS dan OS/2 keduanya mempunyai masalah dengan disk yang memiliki lebih dari 1024 silinder (silahkan baca *large-disk* Mini-HOWTO mengenai hal yang terinci dari disk demikian.

3 Partisi-partisi apa saja yang akan saya perlukan?

3.1 Berapa jumlah partisi yang akan saya butuhkan?

Baiklah, sekarang, partisi-partisi apa saja yang penulis perlukan? Beberapa sistem operasi tidak dapat di-boot dari partisi logical dengan alasan yang sukar difahami. Oleh karenanya Anda mungkin perlu untuk menyediakan partisi primer sebagai partisi boot untuk MS-DOS, OS/2, Linux, atau sistem operasi lain yang Anda gunakan. Perlu diingat bahwa sebuah partisi primer diperlukan bagi suatu partisi extended, yang berfungsi sebagai kontainer bagi bagian disk yang digunakan sebagai partisi logical.

Proses booting dari sistem operasi merupakan suatu real mode yang melibatkan BIOS dan pembatasan 1024 silinder. Oleh karenanya, semua partisi boot perlu Anda tempatkan di dalam 1024 silinder pertama dari harddisk Anda, untuk menghindarkan adanya masalah. Sekali lagi, silahkan membaca Mini-HOWTO "Large-disk" untuk mendapatkan keterangan terperinci mengenai hal ini.

Untuk dapat meng-install Linux Anda membutuhkan minimal satu partisi. Jika kernel di-load dari partisi ini (misalnya dengan LILO), partisi ini harus dapat dibaca oleh BIOS. Jika Anda menggunakan cara lain untuk me-load kernel (misalnya dengan sebuah boot disk, atau dengan Linux Loader berbasis MS-DOS: `LOADLIN.EXE`), partisi boot dapat ditempatkan di partisi mana saja. Dalam kedua kasus diatas, partisi ini bertipe: 0x83 "Linux Native".

Sistem Anda memerlukan area swap. Anda akan memerlukan sebuah partisi yang didekasikan untuk swap kecuali kalau Anda akan melakukan swap kedalam file. Karena partisi ini hanya diakses oleh kernel Linux dan kernel Linux tidak terpengaruh oleh kekurangan PC BIOS, maka partisi swap dapat diletakan di mana saja. Penulis menyarankan untuk menggunakan sebuah partisi logical untuk keperluan ini (`/dev/?d?5` atau lebih tinggi dari ini). Partisi yang disediakan untuk swap bertipe 0x82 "Linux swap".

Itulah tadi kebutuhan minimal partisi. Namun akan berguna jika Anda membuat lebih banyak partisi, karena itu teruskanlah Anda membaca tulisan ini.

3.2 Berapa besar seharusnya ukuran area swap itu?

Jika Anda telah memutuskan untuk membuat partisi yang terdedikasi untuk swap, yang umumnya merupakan gagasan bagus, ikuti pedoman berikut:

- Di dalam Linux, ukuran RAM dan swap digabungkan (Hal ini tidak berlaku untuk semua jenis Unices). Misalnya Anda mempunyai RAM 8MB dan area swap 12 MB maka secara total Anda memiliki 20 MB virtual memory.
- Di dalam menentukan ukuran area swap, Anda memerlukan minimal 16 MB dari total virtual memory. Misalnya untuk RAM 4 MB Anda memerlukan area swap minimal 12 MB, sedangkan untuk RAM 8 MB Anda memerlukan area swap minimal 8 MB.
- Di dalam Linux, ukuran sebuah partisi swap tidak dapat melebihi 128 MB. Meskipun ukuran partisinya dapat melebihi ukuran diatas tetapi kelebihan space dari 128 MB tidak pernah digunakan. Jika menginginkan area swap berukuran lebih dari 128 MB, Anda dapat membuat lebih dari satu partisi swap.
- Pada saat menetapkan ukuran area swap, perlu diingat bahwa ukuran area swap yang terlalu besar juga mungkin tidak ada gunanya sama sekali. Setiap proses memiliki "working set" yang berisi page dari memory yang akan diakses oleh prosesor dalam waktu dekat. Linux mencoba untuk melakukan estimasi terhadap pengaksesan memory tsb (diasumsikan bahwa page yang baru diakses akan digunakan kembali dalam waktu dekat) dan menyimpan page tsb kedalam RAM selama memungkinkan. Apabila program memiliki "locality of reference" yang baik maka asumsi ini akan berlaku demikian juga algoritma pendugaan akan bekerja baik. Melakukan penyimpanan sebuah working set di dalam main memory hanya akan dapat dilakukan apabila tersedia cukup main memory. Jika proses yang berjalan dalam sebuah komputer terlalu banyak jumlahnya, kernel akan dipaksa untuk menaruh page kedalam disk yang akan diambil lagi kemudian (memaksa sebuah page untuk dikeluarkan dari sebuah working set dan kemudian memanggil sebuah page yang akan diakses). Biasanya proses ini akan mengakibatkan peningkatan dari aktifitas paging dan penurunan performance yang drastis. Sebuah komputer dalam keadaan ini disebut sedang melakukan "thrasing". Didalam sebuah komputer yang sedang thrasing sebuah proses running dari disk dan bukannya dari RAM. Penurunan performance dapat diperkirakan yaitu sekitar rasio antara kecepatan akses memori dengan keepatan akses disk. Sebuah pedoman lama dari masa PDP dan Vax, ukuran working set dari suatu program adalah 25% dari ukuran virtualnya.

Sehingga mungkin percuma saja untuk menyediakan swap lebih dari tiga kali ukuran RAM. Tetapi perlu diingat pula bahwa itu hanya pedoman umum saja. Sangat mungkin terjadi suatu skenario di mana program-program mempunyai working set dengan ukuran sangat besar ataupun sangat kecil. Misalnya, sebuah program simulasi dengan data set yang besar yang diakses secara random akan tidak terlihat memiliki locality of reference dalam data segmennya, sehingga working setnya akan sangat besar. Kebalikannya, sebuah program xv dengan banyak file JPEG yang dibuka secara simultan, semuanya kecuali satu yang ter-ikonifikasi, akan memiliki data segmen yang besar. Akan tetapi, transformasi image dilakukan semuanya pada satu image sehingga sebagian besar memory yang digunakan oleh xv tidak pernah disentuh. Hal sama akan berlaku pada sebuah editor yang memiliki banyak window editor namun hanya satu window yang dimodifikasi pada suatu waktu. Program-program tsb - jika didisain secara benar - memiliki locality of reference yang sangat tinggi dan bagian besar dari data segmennya dapat di-swap keluar tanpa menyebabkan pengurangan performance yang berarti. Sesorang dapat menduga bahwa 25% angka-angka yang berasal dari masa command line tidak berlaku pada lagi dimasa program editing dengan modern GUI dan mamou mengedit beberapa dokumen sekaligus, namun penulis belum membaca satu paperpun yang mencoba memverifikasi angka-angka tersebut.

Dengan demikian, untuk suatu konfigurasi dengan RAM 16MB, swap tidak diperlukan bagi konfigurasi minimal dan area swap berukuran lebih dari 48 MB mungkin tidak berguna. Jumlah memory yang pasti diperlukan tergantung dari gabungan aplikasi dalam komputer tsb (apa yang Anda harapkan?).

3.3 Di manakah saya harus meletakkan area swap?

- Mekanik bersifat lambat, sedangkan elektronik bersifat cepat. Hard disk modern memiliki beberapa head. Perpindahan antar head dalam track yang sama terjadi dengan cepat karena bersifat elektronik. Perpindahan antar track terjadi dengan lambat karena melibatkan pergerakan benda. Dengan demikian jika Anda memiliki sebuah disk dengan banyak head dan sebuah lagi disk dengan sedikit head, sedangkan parameter lain dari keduanya sama, disk dengan banyak head akan memiliki kecepatan lebih tinggi. Dengan membagi swap dan meletakkan setiap bagiannya di dalam kedua disk akan menyebabkan kecepatannya meningkat.

- Disk model lama memiliki jumlah sektor yang sama pada setiap tracknya. Dengan disk ini, kecepatan akan lebih tinggi apabila swap Anda taruh ditengah-tengah disk, dengan asumsi bahwa disk head akan bergerak dari suatu track secara random ke swap area.

- Disk model baru memiliki ZBR (zone bit recording) yaitu sektornya berjumlah lebih banyak pada track yang lebih luar. Dengan bilangan rpm yang tetap akan menyebabkan performance pada track luar lebih besar dibanding pada track lebih dalam. Karena itu, letakkanlah swap pada track yang cepat.

- Sudah barang tentu disk head tidak akan bergerak secara random. Apabila Anda memiliki area swap ditengah-tengah disk diantara home partisi yang selalu sibuk dengan partisi arsip yang hampir tidak digunakan, akan lebih baik jika Anda letakkan swap ditengah-tengah partisi home ini untuk memperpendek gerakan head. Akan lebih baik lagi jika swap ditempatkan dalam disk yang lain yang tidak digunakan.

Ringkasan: Letakkan swap di dalam disk yang cepat yaitu yang tidak sibuk melakukan pekerjaan. Jika Anda memiliki beberapa disk: pecah swap dan letakkan secara tersebar pada berbagai tempat pada disk atau bahkan pada kontroler yang berbeda.

Lebih baik lagi: belilah RAM lebih banyak.

3.4 Beberapa fakta mengenai file sistem dan fragmentasi

Disk space diatur oleh sistem operasi dalam satuan blok atau fragmen dari blok. Karena dalam ext2, fragmen dan blok harus berukuran sama maka kita dapat membatasi pembicaraan hanya pada blok saja.

File dapat memiliki berbagai ukuran dan tidak berakhir tepat pada batas blok. Dengan demikian setiap bagian dari blok yang tidak digunakan oleh file menjadi terbuang. Dengan asumsi file berukuran random maka sekitar setengah blok terbuang dari setiap file yang terdapat pada disk. Tanenbaum dalam bukunya berjudul "Operating System" menyebut hal ini sebagai fragmentasi internal.

Anda dapat membuat pendugaan mengenai jumlah file yang terdapat pada disk Anda dari jumkah inodes pada disk yang dialokasikan. Dalam disk penulis:

```
#df     -I
Filesystem
/dev/hda3
/dev/hda5
```

Di dalam / terdapat 12000 file dan sekitar 44000 file pada `/var`. Dengan ukuran 1 KB setiap blok, terdapat sekitar 6+22 = 28 MB dari disk space yang terbuang pada ujung file. Apabila penulis memilih ukuran 4 KB per blok, penulis akan kehilangan 4 kali lipat dari jumlah tadi.

Transfer data akan lebih cepat bila datanya berupa bagian yang tidak terputus-putus. Itulah sebabnya ext2 berusaha untuk melakukan praalokasi space dalam satuan yang terdiri dari 8 blok yang tidak terputus untuk file yang ukurannya berkembang. Block yang telah di-praalokasi namun tidak digunakan akan dikembalikan pada saat file ditutup, sehingga tidak ada space yang terbuang.

Penggantian blok yang terputus-putus letaknya berakibat kurang baik terhadap performance, karena file seringkali diakses secara sekuensial. Hal tsb akan memaksa sistem operasi untuk menggerakan headnya sewaktu melakukan akses kedalam disk. Kondisi blok yang terpisah-pisah ini disebut fragmentasi eksternal atau disingkat fragmentasi yang merupakan hal umum terjadi pada file sistem DOS.

Ext2 memiliki beberapa strategi untuk menghindarkan external fragmentasi. Umumnya fragmentasi bukan merupakan masalah besar dalam ext2, demikian pula pada partisi yang tingkat penggunaannya sangat tinggi seperti pada spool USENET news. Meskipun untuk menghilangkan fragmentasi sudah terdapat alatnya, dan tidak ketinggalan meskipun dengan versi terbaru dari ext2, namun tidak seorangpun pernah menggunakannya. Gunakanlah alat itu, namun Anda harus menanggung sendiri resikonya.

File sistem MS-DOS terkenal dengan kejelekan dalam penanganan disk space. Selain dari buffer cache yang digunakan MS-DOS berukuran besar, efek dari fragmentasi file terhadap performance sangat nampak. Pemakai DOS sudah menjadi terbiasa untuk menghilangkan fragmentasi disk pada setiap minggunya, bahkan bagi sebagian pemakai sudah berkembang semacam kepercayaan mengenai defragmentasi ini. Tidak satupun kebiasaan diatas perlu dibawa kedalam Linux dan ext2. Linux native file sistem, dalam penggunaan normal tidak memerlukan defragmentasi, hal ini berlaku juga pada setiap keadaan di mana free space minimal 5% dari kapasitas disk.

File sistem MS-DOS terkenal menyebabkan hilangnya sejumlah besar disk space disebabkan fragmentasi internal. Untuk partisi berukuran lebih dari 256 MB, ukuran blok dari DOS berkembang sangat besar sehingga tidak berguna lagi (hal ini telah diperbaiki sampai ketingkat tertentu dengan FAT32).

Ext2 tidak mengharuskan Anda untuk memilih ukuran blok yang besar untuk file sistem yang besar, kecuali untuk file sistem yang sangat besar dalam kisaran 0.5 TB atau lebih (1 TB atau 1 terabyte sama dengan 1024 GB), di mana ukuran blok yang kecil kurang efisien. Berbeda dengan pada DOS, sebuah disk berukuran besar tidak perlu dibagi menjadi beberapa partisi untuk menjaga agar ukuran bloknya tetap. Jika memungkinkan, pilihlah *default* dari ukuran blok yaitu 1 KB. Anda dapat bereksperimen menggunakan ukuran blok sebesar 2 KB pada beberapa partisi, tetapi jangan terkejut unruk menemui bug yang tidak diketahui. Kebanyakan orang menggunakan ukuran *default*.

3.5 Masa hidup file dan siklus backup sebagai kriteria pembuatan partisi

Keputusan pembuatan partisi dengan ext2 harus didasarkan pada pertimbangan pembuatan backup dan usaha menghindari adanya fragmentasi eksternal berbagai file dengan masa hidup file yang berbeda-beda.

File memiliki masa hidup. Sesudah dibuat, sebuah file akan berada dalam sistem sampai file tsb dihapus. Masa hidup dalam sistem sangat bervariasi dan sebagian tergantung kepada pathname file tsb. Misalnya, file-file di dalam direktori-direktori `/bin`, `/sbin`, `/usr/sbin`, `/usr/bin`, dan direktori-direktori sejenis umumnya memiliki masa hidup lama: beberapa bulan atau lebih. File-file dalam direktori `/home` memiliki masa hidup sedang: sekitar beberapa minggu. File-file di dalam direktori `/var` umumnya berumur pendek: hampir tidak ada satu file pun di dalam direktori `/var/spool/news` yang berumur lebih dari beberapa hari, bahkan file-file di dalam direktori `/var/spool/lpd` memiliki masa hidup dalam ukuran menit atau bahkan kurang dari itu.

Untuk keperluan backup, akan sangat berguna jika ukuran backup harian lebih kecil dari kapasitas media backup yang akan digunakannya. Backup harian dapat berupa backup penuh atau hanya berupa incremental backup.

Anda dapat menjaga agar ukuran partisi cukup kecil sehingga cukup untuk dimasukkan kedalam satu media backup (pilihan backup penuh harian). Dalam setiap kasus, ukuran sebuah partisi harus cukup kecil sehingga delta harian (file-file yang telah mengalami perubahan) cukup dimasukan kedalam satu media backup (pilihan incremental backup dan direncanakan untuk mengganti media backup setiap minggu atau bulanan bagi full dump - tanpa kehadiran operator).

Strategi backup Anda tergantung atas keputusan-keputusan diatas.

Pada saat merencanakan dan membeli disk space, perlu diingat untuk menyisakan dana untuk keperluan backup! Data yang tidak dibackup menjadi tidak berguna! Hal yang berlaku umum adalah biaya untuk mereproduksi data lebih tinggi daripada biaya untuk membackup!

Untuk keperluan performance, sangat berguna untuk menyimpan file-file yang yang berbeda mas hidupnya pada partisi yang berbeda. Dengan cara ini maka file-file yang bermasa hidup pendek dalam partisi news dapat mengalami fragmentasi yang tinggi. Hal tsb tidak mempengaruhi performance dari partisi `/` maupun `/home`.

4 Sebuah Contoh

4.1 Sebuah model yang disarankan bagi user yang ambisius

Sebuah model yang umum adalah membuat partisi `/`, `/home`, dan `/var` seperti telah diuraikan diatas. Model tsb mudah untuk di-install dan dipelihara, juga telah melakukan pemisahan cukup baik untuk menghindari efek dari perbedaan waktu hidup. Hal tsb juga sesuai dengan model backup: tidak ada satupun yang tertarik untuk membackup USENET news spool dan hanya beberapa file di dalam `/var` saja yang berharga untuk dibackup (ingatlah direktori `/var/spool/mail`). Di lain fihak, partisi `/` jarang mengalami perubahan dan dapat dibackup sesuai kebutuhan (sesudah perubahan konfigurasi) dan cukup kecil untuk dimasukan kedalam hampir semua media backup modern sebagai full backup (siapkan ukuran 250 sampai 500 MB tergantung ukuran software yang ter-install). Partisi `/home` berisi data yang penting bagi user dan harus dibackup setiap hari. Beberapa instalasi memiliki partisi `/home` yang sangat besar dan harus melakukan incremental backup.

Beberapa sistem menempatkan `/tmp` kedalam partisi terpisah, sistem lainnya membuat symlink kedalam `/var/tmp` untuk mendapatkan efek yang sama (catatan hal ini akan mempengaruhi mode single user, di mana `/var` tidak akan ada dan sistem tidak akan memiliki `/tmp` sampai Anda membuatnya atau melakukan mount secara manual dari `/var`) atau menempatkannya kedalam RAM disk (sebagai contoh: Solaris melakukan hal ini). Hal itu menyebabkan `/tmp` tidak berada di dalam `/`, suatu gagasan bagus.

Model ini sangat baik digunakan untuk upgrade maupun reinstall: menyimpan file konfigurasi (atau keseluruhan `/etc`) kedalam direktori `/home`, hapus `/`, lakukan reinstall dan ambil kembali konfigurasi lama dari `/home`.

5 Bagaimana melakukannya pada komputer penulis

Diatas rak penulis terdapat sebuah komputer 386/40 dengan bus ISA yang penulis tinggalkan sejak dua tahun lalu karena penulis kurang puas dengannya. Penulis merencanakan untuk menjadikan komputer itu sebagai sebuah server tanpa X bagi jaringan komputer penulis di rumah.

Inilah yang penulis lakukan: penulis tambahkan RAM 16 MB kedalam komputer 386 tsb, pasang sebuah disk EIDE murah sebesar ukuran yang mampu penulis beli (800 MB), dan sebuah ethernet card. Penulis tambahkan sebuah Hercules tua karena penulis masih memiliki monitor jenis itu. Penulis install Linux kedalamnya dan penulis memiliki server bagi local server NFS, SMB, HTTP, LPD/LPR dan NNTP, juga mail router dan server POP3. Dengan tambahan ISDN card, komputer tsb bahkan menjadi TCP/IP router dan firewall.

Hampir semua disk space dalam komputer itu digunakan untuk direktori-direktori `/var`, `/var/spool/mail`, `/var/spool/news`, dan `/var/httpd/html`. Penulis meletakkan `/var` di dalam partisi yang terpisah dan membuatnya berukuran besar. Komputer ini hampir tidak memiliki user, sehingga penulis tidak membuat partisi home, dan melakukan mount direktori `/home` dari workstation lain dengan NFS.

Linux tanpa X dengan beberapa utility yang terinstal secara local cukup baik dengan hanya sebuah partisi 250 MB sebagai `/`. Komputer memiliki RAM 16 MB tetapi akan menjalankan beberapa server. Swap sebesar 16 MB haruslah cukup, sedangkan 32 MB cukup banyak. Karena disk space pada komputer tsb tidak kurang maka komputer akan mendapatkan 32 MB sebagai swap. Untuk alasan sentimentil terhadap MS-DOS, sebuah partisi sebesar 20 MB dibuat di dalam disk. Penulis memutuskan untuk meng-import `/home` dari komputer lain, sehingga sisa 500+ MB akan digunakan sebagai `/var`. Ukuran itu sudah lebih dari cukup untuk USENET news feed rumahan.

Kita mendapatkan seperti berikut:

```
Device              Mounted on                  Size
/dev/hda1           /dos_c                      25 MB
/dev/hda2           - (Swap space)              32 MB
```

```
/dev/hda3            /                              250 MB
/dev/hda4            - (Extended container)         500 MB
/dev/hda5            /var                           500 MB

homeserver: /home    /home                          1.6 GB
```

Penulis melakukan backup komputer diatas dengan tape di dalam homeserver melalui network. Karena semua dalam komputer ini diinstall melalui CD-ROM, yang penulis perlu simpan adalah beberapa file konfigurasi dari `/etc`. Sistem yang penulis buat khusus dapat menginstall secara lokal file-file `*.tgz` dari `/root/Source/Installed`, `/var/spool/mail`, dan `/var/httpd/html`. Penulis meng-copy file-file tsb kedalam direktori khusus yaitu /home/backmeup yang berada di dalam komputer homeserver di mana setiap malam backup homeserver akan mengambilnya secara rutin.