

Manajemen Proses dan Memori di Linux

M Zainal Arifin
agendogget@yahoo.com

Lisensi Dokumen:

Copyright © 2004 IlmuKomputer.Com

Seluruh dokumen di IlmuKomputer.Com dapat digunakan, dimodifikasi dan disebarkan secara bebas untuk tujuan bukan komersial (nonprofit), dengan syarat tidak menghapus atau merubah atribut penulis dan pernyataan copyright yang disertakan dalam setiap dokumen. Tidak diperbolehkan melakukan penulisan ulang, kecuali mendapatkan ijin terlebih dahulu dari IlmuKomputer.Com.

BAB I

Pendahuluan

1.1 Latar Belakang

Sebuah Sistem Operasi adalah kumpulan program-program (software/perangkat lunak) yang membantu para pemakai komputer untuk berkomunikasi dengan komputernya. Bisa dianalogikan sebagai “Suatu Organisasi Pemerintahan” pada kumpulan komunitas yang ada di dalam komputer kita (misalnya: harddisk, mouse, VGA card, CPU dan lain-lain), dimana Program Utama memberitahu kepada program lain apa dan akan mereka lakukan untuk memberikan layanan yang mereka butuhkan. Sebuah komputer dapat dimungkinkan mempunyai lebih dari sebuah sistem operasi, tetapi hanya satu sistem operasi komputer yang dapat jalan pada satu saat .

Ketika komputer pertama kali menyala atau menjalankan sistem operasi itu dikenal dengan proses booting pada komputer.

Sistem Operasi Linux adalah salah satu contoh dari sebuah model sistem operasi dewasa ini. Windows XP juga adalah sebuah sistem operasi, begitupun juga

MS-DOS. Kesemuanya mempunyai tugas yang hampir sama. Namun perlu diketahui bahwa hanya satu sistem operasi yang dapat digunakan pada satu saat.

Istilah-istilah yang perlu diketahui dalam memahami sistem operasi:

Memori

Memori berfungsi untuk menyimpan data dan program.

Kernel

Kernel adalah program inti yang mengatur komponen penting komputer (processor dan memori sebagai contoh); serta beberapa device driver yang mengatur sistem perangkat keras yang lain (kartu jaringan, kartu suara dll). Kernel mengatur segala sesuatu yang berhubungan dengan program-program untuk dapat berkomunikasi dengan perangkat kerasnya.

Shell

Sebuah program yang dapat berkomunikasi dengan pengguna komputer (manusia) dan memungkinkan pengguna komputer untuk berinteraksi dengan komputer. Shell adalah salah satu bagian dari sebuah sistem operasi yang dapat dilihat. Dimana setiap sistem operasi menampilkan beragam antarmuka yang berbeda atau lingkungan kerja yang berbeda.

Proses

Proses adalah program yang sedang dieksekusi. Proses merupakan unit kerja terkecil yang secara individu memiliki sumber daya-sumber daya dan dijadwalkan sistem operasi.

Multiprogramming (multitasking)

Manajemen banyak proses pada satu pemroses. Banyak proses dijalankan bersamaan, masing-masing proses mendapat bagian memori dan kendali tersendiri. Sistem operasi mengalih-alihkan pemroses diantara proses-proses tersebut.

Multiprocessing

Manajemen banyak proses di komputer *multiprocessor* (banyak proses di dalamnya).

Distributed Processing

Manajemen banyak proses yang dieksekusi di banyak sistem komputer yang tersebar (terdistribusi).

1.2 Batasan Masalah

Makalah ini membahas manajemen proses dan manajemen memori pada sistem operasi Mandrake 8.2 kernel.... sebagai salah satu distro Linux.

1.3 Tujuan

Untuk lebih memahami Linux yang bersifat *open source* sebagai sarana yang tepat untuk belajar.

BAB II

Pembahasan

Linux adalah tiruan (clone) UNIX. Pengembangan Linux pertama kali dilakukan Linus Benedict Torvalds, Universitas Helsinki, Finlandia sebagai proyek hobi. Seluruh kode sumber Linux termasuk kernel, device drivers, libraries, program dan tool pengembangan disebarakan secara bebas dengan lisensi GPL (General Public License) versi kedua kemudian berkembang cepat melalui bantuan seluruh programmer di dunia melalui jaringan internet.

Linux pertama kali dipublikasikan sekitar november 1991, dikenal dengan versi 0.10 kemudian disusul versi 0.11 pada desember 1991. Pada versi 0.13 Linux sudah lebih stabil dan Linus memutuskan megubah versinya menjadi versi 0.95. Sifat Linux yang terbuka membuatnya masih terus dikembangkan oleh kelompok-kelompok tanpa dibayar, yang banyak dijumpai di Internet. Mereka saling tukar-menukar kode, melaporkan bug, dan membenahi segala masalah yang ada. Setiap orang yang tertarik dipersilahkan untuk bergabung dalam pengembangan Linux.

Linux mempunyai kelebihan dibanding sistem operasi yang lain:

- Full Multitasking dan Full 32-bit. Linux seperti halnya versi UNIX yang lain mendukung penuh multitasking, sehingga pengguna dapat menjalankan banyak program pada saat bersamaan. Linux mendukung manajemen memori protected-mode pada platform processor sekelas Intel 80386 ke atas.
- X Window system. X Windows merupakan standar tampilan grafis dari mesin-mesin UNIX. Versi terlengkap dari X Window yang dikenal dengan Xfree86 telah tersedia untuk Linux. X Window dengan tampilan grafis yang menawan dapat mendukung banyak aplikasi.
- Implementasi TCP/IP Networking. Implementasi TCP/IP (Transmission Control Protocol/Internet Protocol) yang lengkap sebagai penghubung ke dunia internet. Banyak aplikasi yang tersedia seperti: SLIP/CSLIP, PLIP, PPP, NFS, FTP, Telnet, NNTP, SMTP dan sebagainya. Tersedia protokol dasar di kernel termasuk TCP, Ipv4, Ipv6, AX.25, X.25, DDP(AppleTalk), NetBEUI, Netrom dan sebagainya.
- Mendukung virtual memori dan shared library. Virtual memori memungkinkan penggunaan ruang pada harddisk sebagai memori, sehingga dapat mengatasi

kekurangan RAM untuk menjalankan suatu proses. Shared library memungkinkan program untuk menggunakan library bersama-sama sehingga file executable dapat lebih sedikit menggunakan ruang pada disk.

- Dukungan GNU Software. Linux memiliki banyak aplikasi pendukung yang powerful dimana aplikasi ini dibuat oleh GNU –sebuah badan pembuat *free software*.
- Dukungan penuh terhadap Networking
- Lebih murah. Sebenarnya linux sistem operasi yang dapat diperoleh secara gratis. Biaya yang dikeluarkan mungkin hanya untuk pengganti CD atau pulsa telpon jika mendapatkannya dari internet.

Arsitektur Dasar Sistem Operasi Linux

Bagian terpenting sistem operasi adalah *kernel*, merupakan jantung sistem operasi.

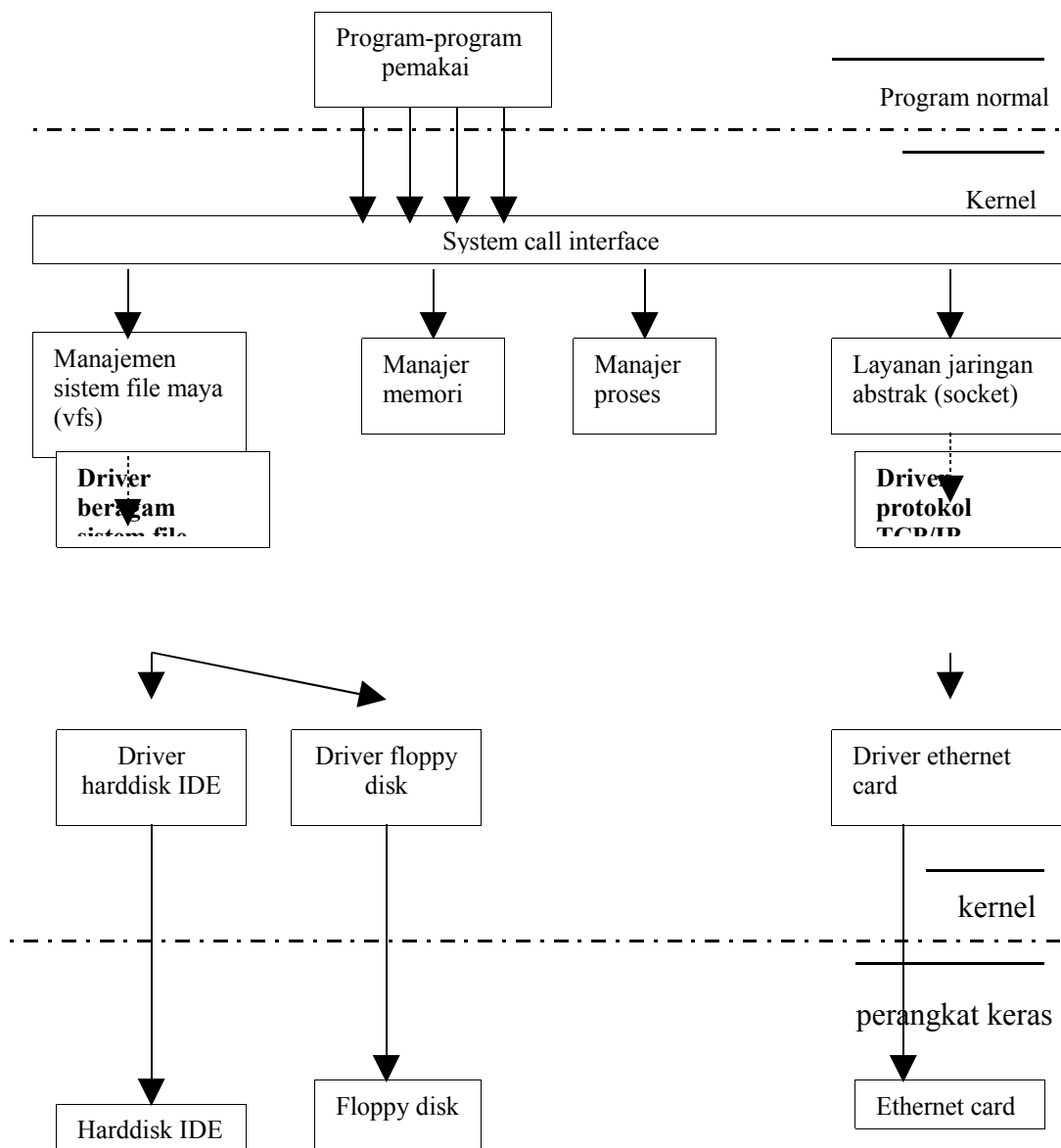
- *Kernel* menyediakan *tool* dimana semua layanan sistem komputer disediakan.
- *Kernel* mencegah proses aplikasi mengakses perangkat keras secara langsung, memaksa proses menggunakan *tool* yang disediakan.
- *Kernel* memberi proteksi kepada pemakai dari gangguan pemakai lain

Tool Kernel digunakan melalui panggilan sistem (*system call*). Program sistem menggunakan *tool kernel* untuk implementasi beragam layanan. Program sistem dan semua program lain berjalan diatas kernel. Program pemakai berjalan di mode berbeda dengan kernel, disebut mode pemakai.

Kernel berisi beberapa bagian penting, yaitu:

- Manajemen proses
- Manajemen memori
- Driver-driver perangkat keras
- Driver-driver sistem file
- Manajemen jaringan
- Dan beragam subsistem lain

Berikut adalah arsitektur sistem operasi Linux :



PROSES

Proses adalah program tunggal yang berjalan pada alamat virtual, berarti setiap hal yg berjalan dibawah Linux adalah proses. Proses ini berjalan melalui perintah-perintah yang ada dalam shell. Satu baris perintah dalam shell terkadang mampu mempengaruhi lebih dari satu proses, khususnya jika terdapat perintah *pipe*. Contoh:

```
nroff -man ps.1 | grep kill | more
```

perintah ini melakukan tiga proses, satu proses untuk setiap perintahnya.

Tipe-tipe proses dalam Linux:

Ada beberapa tipe proses dalam Linux . Setiap proses mempunyai kekhususan dan atribut tersendiri.:

- Interactive processes: Proses yang dimulai (dan dikontrol oleh) shell. Bisa tampak diluar (foreground) ataupun hanya didalam (background).
- Batch processes: Proses yang tidak berhubungan dengan terminal tetapi menunggu untuk dieksekusi secara sequent.
- Daemon processes: Proses yang dimulai ketika Linux booting dan berjalan secara background.

Cara termudah untuk mengetahui proses apa yang sedang berjalan pada sistem adalah menggunakan perintah ps (process status). Perintah ps mempunyai beberapa option dan argumen, tersedia untuk seluruh user dan root, walau tentu hasilnya akan berbeda tergantung *login* anda. Jika anda login sebagai user biasa (bukan root), jika perintah ps ditulis akan tampak hasil seperti:

```
$ ps
```

```
PID TTY STAT TIME COMMAND
41 v01 S 0:00 -bash
134 v01 R 0:00 ps
```

Hasil dari perintah ps selalu disusun dalam bentuk kolom. Kolom pertama berlabel PID (Process Identification Number); adalah penomoran dari Linux untuk menandai sebuah proses, dimulai dari nol dan bertambah satu untuk tiap proses hingga nomor tertinggi. Jika Linux mencapai nomor tertinggi, penomoran akan dimulai lagi dari nomor terendah dengan melewati nomor yang sedang digunakan proses yang aktif. Biasanya nomor terendah digunakan oleh proses kernel dan daemon yang dimulai saat booting hingga Linux berjalan. Untuk memanipulasi proses kita dapat menggunakan PID. Kolom TTY menunjukkan terminal yang digunakan saat proses dimulai. Kolom STAT menunjukkan kondisi terkini dari proses, entri yang paling banyak terdapat adalah S untuk sleeping dan R untuk running. Kolom TIME menunjukkan CPU time yang digunakan oleh proses. Kolom NAME menunjukkan nama proses yang berjalan, biasanya adalah nama perintah yang dimasukkan. Beberapa perintah digunakan untuk memulai perintah yang

lain, proses ini dikenal dengan *child process*. Terdapat banyak versi perintah `ps` tergantung versi Linux yang digunakan. Kebanyakan admin (jika login sebagai root) menggunakan perintah `ps` dibawah ini untuk menampilkan keseluruhan informasi sistem:

```
ps -ef
```

```
ps -le
```

Proses yang terkunci dalam terminal dan tidak dapat melakukan apa pun disebut hang. Terkadang adapula sebuah proses yang berahir tidak sempurna. Hal ini disebut *runaway process*. Untuk mengatasi hal ini dan mengembalikan sistem ke keadaan normal digunakan perintah `kill`. Untuk menggunakan perintah ini kita harus mengakses konsol lain, jika terminal benar-benar terkunci kita harus login kembali. Sebagai seorang user biasa hanya dapat meng*kill* proses user itu sendiri dan tak dapat mempengaruhi user lain dalam sistem. Sebagai root anda mempunyai kuasa penuh menggunakan perintah `kill`. Diperlukan PID untuk mengetahui proses yang berjalan sehingga dapat meng*kill*nya.

Memori

Organisasi dan manajemen memori sangat mempengaruhi kinerja komputer. Manajemen memori melakukan tugas penting dan kompleks berkaitan dengan:

- Memori utama sebagai sumber daya yang harus dialokasikan dan dipakai bersama diantara sejumlah proses yang aktif. Agar dapat memanfaatkan pemroses dan fasilitas masukan/keluaran secara efisien, maka diinginkan memori yang dapat menampung sebanyak mungkin proses.
- Upaya agar pemrogram atau proses tidak dibatasi kapasitas memori fisik di sistem komputer.

Linux memanfaatkan virtual memori untuk mendukung kinerja sistem. Sebagai sistem operasi multiprogramming, virtual memori dapat meningkatkan efisiensi sistem. Sambil proses menunggu bagiannya di*swap* masuk ke memori, menunggu selesainya operasi masukan/keluaran dan proses di*blocked*. Jatah waktu pemroses dapat diberikan ke proses-proses lain.

Manajemen memori Linux menyediakan:

1. Ruang alamat besar

Ruang alamat dapat lebih besar dibanding memori fisik yang tersedia

2. Proteksi

Tiap proses di sistem mempunyai ruang alamat maya tersendiri. Ruang-ruang alamat maya itu sepenuhnya terpisah. Proses yang berjalan di satu aplikasi tidak dapat mengganggu proses lainnya.

3. Pemetaan memori

Dilakukan pemetaan antara memori maya ke memori fisik yang tersedia.

4. Memori maya bersama (shared virtual memory)

Memori maya bersama ini untuk menghemat ruang memori, seperti pustaka dinamis bagi beberapa proses.

Karena memori fisik lebih sedikit dibanding memori maya, maka sistem hanya memuatkan *page-page* maya yang saat itu sedang digunakan proses. Linux memuatkan *page* maya begitu diperlukan. Teknik ini disebut dengan *demand paging*. Saat pemroses berusaha mengakses alamat maya yang tidak di memori fisik, pemroses tidak dapat menemukan isian di tabel *page* maya. Pemroses menerbitkan *page fault*.

- Jika alamat maya yang dituju tak absah (yaitu proses berusaha mengakses alamat maya yang tidak dibolehkan), maka sistem operasi mengakhiri proses itu untuk memproteksi proses-proses lain.
- Jika alamat maya yang dituju absah tapi tidak sedang di memori fisik, maka sistem operasi harus membawa *page* ke memori dari *disk*. Pengaksesan disk memerlukan waktu lama. Jika terdapat proses lain yang dapat dijalankan, maka sistem operasi memilih proses lain untuk dijalankan. Proses dimulai kembali di instruksi dimana *page fault* terjadi. Pemroses dapat memetakan memori maya ke memori fisik saat dilakukan pengaksesan memori maya, sehingga proses terus berjalan. Jika tidak terdapat memori bebas, sistem operasi harus membuat ruang bagi *page* yang akan dimasukkan dengan membuang *page* lain dari memori.
 - Jika *page* telah dimodifikasi (*dirty-page*), sistem operasi harus menjaga isi *page*. Ketika dipindahkan dari memori, disimpan di ruang khusus (*swap space*).
 - Jika *page* yang dibuang berupa *page* belum ditulis maka *page* tidak perlu dituliskan ke *disk*.

Berikut code program memory manager-nya:

```
#ifndef _MEMORY_H
```

```
#define      _MEMORY_H

#include <features.h>
#ifndef     _STRING_H
# include <string.h>
#endif
#endif

#ifndef     _LINUX_MM_H
#define     _LINUX_MM_H

#include <linux/sched.h>
#include <linux/errno.h>

#ifdef     __KERNEL__

#include <linux/config.h>
#include <linux/string.h>
#include <linux/list.h>
#include <linux/mmzone.h>
#include <linux/swap.h>
#include <linux/rbtree.h>

extern unsigned long max_mapnr;
extern unsigned long num_physpages;
extern void * high_memory;
extern int page_cluster;
extern struct list_head active_list;
extern struct list_head inactive_list;

#include <asm/page.h>
#include <asm/pgtable.h>
#include <asm/atomic.h>

struct vm_area_struct {
    struct mm_struct * vm_mm;
    unsigned long vm_start;
    unsigned long vm_end;
within vm_mm.
    * vm_flags..
#define VM_DENYWRITE      0x00000800
#define VM_EXECUTABLE    0x00001000
#define VM_LOCKED 0x00002000
#define VM_IO             0x00004000

#define VM_SEQ_READ      0x00008000
#define VM_RAND_READ     0x00010000
#define VM_DONTCOPY      0x00020000
#define VM_DONTEXPAND    0x00040000
#define VM_RESERVED      0x00080000
#if defined(CONFIG_GRKERNSEC_PAX) || defined
(CONFIG_GRKERNSEC_PAX_RANDMMAP)
#define VM_STACK_FLAGS  0x00000133
#else
#define VM_STACK_FLAGS  0x00000177
#endif

#define VM_READHINTMASK      (VM_SEQ_READ | VM_RAND_READ)
#define VM_ClearReadHint(v) (v)->vm_flags &= ~VM_READHINTMASK
#define VM_NormalReadHint(v) (!((v)->vm_flags &
VM_READHINTMASK))
#define VM_SequentialReadHint(v) ((v)->vm_flags & VM_SEQ_READ)
```

```
#define VM_RandomReadHint(v)          ((v)->vm_flags & VM_RAND_READ)

extern int vm_min_readahead;
extern int vm_max_readahead;

extern pgprot_t protection_map[16];

struct vm_operations_struct {
    void (*open)(struct vm_area_struct * area);
    void (*close)(struct vm_area_struct * area);
    struct page * (*nopage)(struct vm_area_struct * area, unsigned
long address, int unused);
};

typedef struct page {
    struct list_head list;
    struct address_space *mapping;
    unsigned long index;
    struct page *next_hash;
    atomic_t count;
    wait_queue_head_t wait;
    struct page **pprev_hash;
    void *virtual;
    struct zone_struct *zone;
} mem_map_t;
```

Linux menempatkan proses pada memori yang dibagi menjadi sejumlah partisi. Pemartisian ini bersifat dinamis maka jumlah, lokasi dan ukuran proses di memori dapat beragam sepanjang waktu secara dinamis. Proses yang akan masuk ke memori segera dibuatkan partisi sesuai kebutuhan. Linux menciptakan ruang disk tempat swap terlebih dahulu, saat proses diciptakan, ruang swap pada disk dialokasikan. Ketika proses harus dikeluarkan dari memori utama, proses selalu ditempatkan ke ruang yang telah dialokasikan, bukan ke tempat-tempat berbeda setiap kali terjadi swap-out. Ketika proses berakhir, ruang swap pada disk didealokasikan.

BAB III

Kesimpulan

Linux merupakan tiruan dari UNIX. Linux memiliki kompatibilitas tinggi terhadap UNIX, bahkan GNU utilitasnya memiliki kemampuan yang lebih baik dibandingkan milik UNIX. Linux dirancang secara spesifik untuk platform PC. Linux merupakan sistem operasi multitasking, multiuser, mendukung network secara langsung, mendukung protected-mode.

Linux memiliki hierarki proses yang mirip dengan UNIX. Ada beberapa tipe proses dalam Linux . Setiap proses mempunyai kekhususan dan atribut tersendiri.:

- Interactive processes: Proses yang dimulai (dan dikontrol oleh) shell. Bisa tampak diluar (foreground) ataupun hanya didalam (background).
- Batch processes: Proses yang tidak berhubungan dengan terminal tetapi menunggu untuk dieksekusi secara sequent.
- Daemon processes: Proses yang dimulai ketika Linux booting dan berjalan secara background.

Linux memanfaatkan virtual memori sehingga kerja sistem dapat lebih efisien. Manajemen memori Linux menyediakan:

1. Ruang alamat besar

Ruang alamat dapat lebih besar dibanding memori fisik yang tersedia

2. Proteksi

Tiap proses di sistem mempunyai ruang alamat maya tersendiri. Ruang-ruang alamat maya itu sepenuhnya terpisah. Proses yang berjalan di satu aplikasi tidak dapat mengganggu proses lainnya.

3. Pemetaan memori

Dilakukan pemetaan antara memori maya ke memori fisik yang tersedia.

4. Memori maya bersama (shared virtual memory)

Memori maya bersama ini untuk menghemat ruang memori, seperti pustaka dinamis bagi beberapa proses.

BIOGRAFI PENULIS

M



Zainal Arifin, lahir di Malang, 7 agustus 1980. Mengikuti pendidikan SD sampai Kuliah di Malang. Masih menempuh kuliah jurusan Matematika di Universitas Brawijaya Malang.

Menggeluti dunia pemrograman sejak tahun 1997 dan sekarang telah menulis beberapa artikel tentang jaringan dan pemrograman di majalah komputer. Sekarang sedang melakukan penelitian tentang teknologi suara dengan program java dengan teman-teman. Sangat tertarik dengan teknologi java terutama pada jaringan komputer dan mobile.

Penulis dapat dihubungi melalui :
Email: agendogget@yahoo.com