

BAB 12

KOMPILASI KERNEL

12.1 Anatomi Direktori/file Kernel Linux

Sebuah *source* kernel Linux biasanya tersedia dalam format kompresi *.gz atau *.bz2. Saat kernel tersebut telah diekstrak, akan terdapat sebuah direktori baru berisi *source* kernel yang telah terekstrak (Misal: kernel-source-2.4.18). Di dalam direktori tersebut terdapat beberapa file dan subdirektori. setiap file/direktori yang terdapat di dalam kernel memiliki fungsi khusus untuk menangani kompilasi, konfigurasi, serta instalasi kernel Linux ke sistem.

```

kari@debian: /usr/src/kernel-source-2.4.18 - Shell - Konsole
Session Edit View Bookmarks Settings Help
Shell Root Console
kari@debian: /usr/src/kernel-source-2.4.18$ ls
COPYING      README      drivers     kernel
CREDITS      README.Debian  fs         lib
Documentation REPORTING-BUGS include     mm
MAINTAINERS  Rules.make   init       net
Makefile     arch        ipc        scripts
kari@debian: /usr/src/kernel-source-2.4.18$

```

Gambar 12.1 Tampilan file/direktori /usr/src/kernel-source-2.4.8

Berikut penjelasan masing-masing *file* dan direktori yang terdapat di dalam direktori *kernel-source* Linux.

Documentation: Berisi informasi tentang kernel Linux dan subsistemnya. Beberapa informasi diperuntukkan bagi *end user* (pengguna). Selain itu, terdapat juga beberapa *file* yang khusus diperuntukkan bagi developer Linux.

arch: Tipe dari arsitektur *processor* komputer yang didukung oleh kernel Linux. Isi dari sub direktori tersebut antara lain: alpha cris ia64 mips parisc s390 sh sparc64 arm i386 m68k mips64 ppc s390x sparc.

drivers: Berisi kode-kode program *driver* perangkat keras seperti *driver USB drive*, kartu jaringan, *video card*, dll.

fs: Berisi kode program yang digunakan untuk mendukung *file system* yang ada (misal: ntfs, ext2, ext3, msdos, minix, dll.).

- include:** Berisi *header file* (*.h) yang akan digunakan oleh *source* kernel.
- init:** Kode inisialisasi kernel.
- ipc:** Kode-kode yang diimplementasikan pada *System V InterProcess Communication APIs* (misal: *semaphores*, *message queues*, dan *shared memory*).
- kernel:** Berisi kode-kode dasar dari kernel Linux. Berisi kode program untuk proses penjadwalan (*scheduling*), pengalokasian *resource*, modul-modul kernel, dll.
- lib:** Standar pustaka C (misal: *ctype*, *sprintf*, dll)
- mm:** Berisi kode sumber untuk sistem manajemen memori Linux.
- net:** Berisi kode-kode program untuk diimplementasikan pada protokol jaringan seperti *ipv6*, *ipv4*, dll.
- scripts:** Berisi *script-script* untuk konfigurasi kernel (misal: *xconfig*, dll).

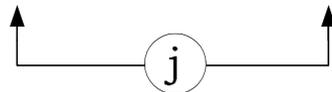
12.2 Instalasi Kernel

Tujuan dari *upgrade* kernel adalah untuk menambah dukungan modul *driver* terbaru dari perangkat keras (*hardware*). *Source* kernel GNU/Linux dapat didownload secara gratis di website <http://www.kernel.org>. Berikut tahap-tahap dalam instalasi kernel Linux:

1. Ekstraksi kernel Linux

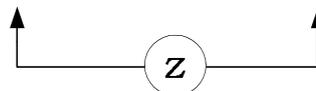
Source kernel Linux biasanya terkompresi dalam format *.gz atau *.bz2. Untuk mengekstraknya, gunakan perintah berikut.

```
debian:~# tar xjvf kernel-source-2.4.18.tar.bz2
```



atau

```
debian:~# tar xzvf kernel-source-2.4.18.tar.gz
```



Kemudian pindah ke direktori *kernel-source-2.4.18* dengan menggunakan perintah berikut.

```
debian:~# cd kernel-source-2.4.18
```

2. Konfigurasi kernel

Tahap ini merupakan tahap yang paling menyenangkan karena pengguna (*user*) melakukan konfigurasi kernel Linux. Terdapat banyak opsi yang tersedia pada *source* kernel Linux dan pilihan opsinya tergantung dari jenis perangkat keras (*hardware*) yang terdapat pada sistem.

Metode konfigurasi

a. make config

Metode ini banyak dihindari oleh pemula karena bekerja dalam mode teks.

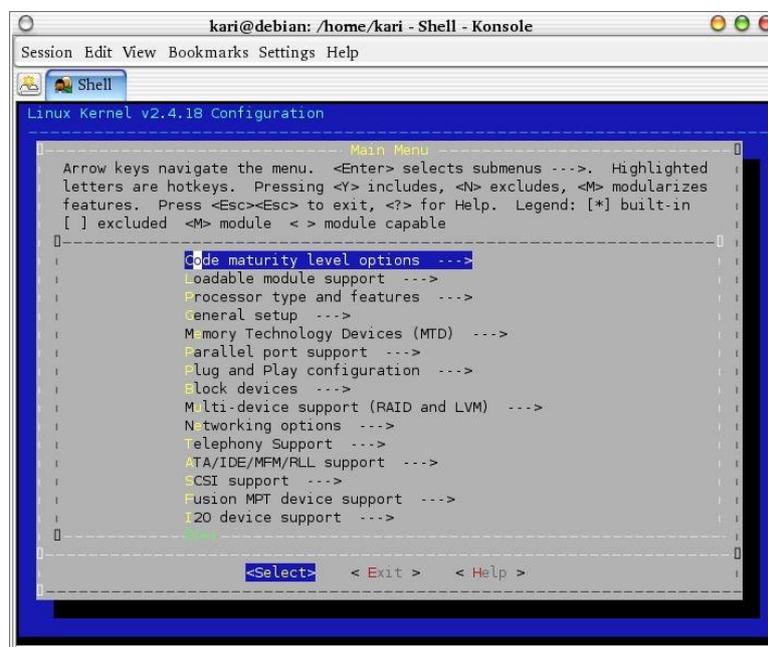
b. make menuconfig

Metode ini paling banyak digunakan oleh pengguna secara umum. Hanya saja tampilan masih dalam mode text namun lebih interaktif dibanding dengan metode pertama.

c. make xconfig

Metode ini menggunakan tampilan grafis (GUI) sehingga lebih memudahkan pengguna. Anda hanya menggunakan mouse untuk memilih option pada kernel-source yang anda butuhkan. Untuk menggunakan metode ini dibutuhkan paket software **tk8.3**

```
debian:~# make menuconfig
```



Gambar 12.2 Tampilan menuconfig kernel linux

Pilihlah setiap opsi yang akan diikutkan dalam kompilasi kernel seperti *driver sound card*, *driver ethernet card*, *driver usb drive*, dll. Opsi-opsi tersebut dapat anda *load* dengan permanen yang ditandai dengan tanda (*) atau sebagai modul saja dan sewaktu-waktu jika dibutuhkan akan diaktifkan (ditandai dengan huruf **M**).

```
debian:~# make dep && make clean
```

3. Instalasi kernel

Tahap ini merupakan bagian terakhir dari proses instalasi kernel Linux beserta modul-modul yang akan diikutsertakan.

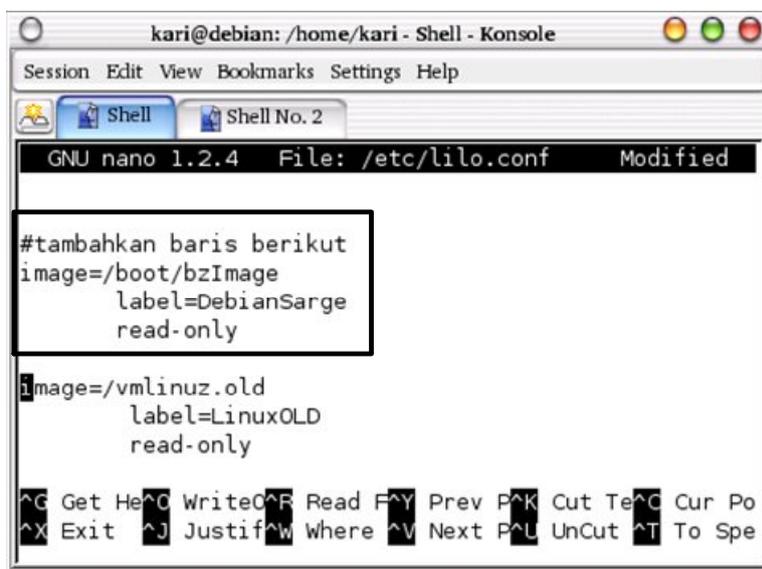
```
debian:~# make bzImage
debian:~# make modules && make modules_install
```

Jika tidak terdapat pesan kesalahan saat instalasi berlangsung, maka akan terdapat sebuah *file* **bzImage** pada direktori */usr/src/kernel-source-2.4.18/arc/i386/boot*. Salin *file* tersebut ke direktori */boot* dengan perintah berikut.

```
debian:~# cp arc/i386/boot/bzImage /boot/
```

Kemudian edit *file* konfigurasi */etc/lilo.conf* jika menggunakan *bootloader* lilo.

```
debian:~# pico /etc/lilo.conf
```



Gambar 12.3 File konfigurasi bootloader lilo

12.2 Instalasi Kernel ala Debian GNU/Linux

Instalasi kernel pada Debian GNU/Linux dapat dilakukan dengan 2 cara. Cara pertama adalah cara konvensional yang lazim digunakan oleh seluruh distro Linux sebagaimana yang telah dijelaskan pada awal bab ini. Dan cara kedua yang akan dijelaskan pada subbab ini.

Debian GNU/Linux memiliki cara khusus untuk membuat segala sesuatu nampak lebih indah dan menawan, termasuk instalasi kernel. Debian memiliki sebuah sistem sendiri untuk instalasi kernel dan sekaligus membangun paket debian secara otomatis. Cara ini sama dengan instalasi paket *software* pada Debian GNU/Linux. Maksudnya, anda dapat menginstal kernel pada mesin (PC) lain yang memiliki tipe atau jenis yang sama dengan PC sebelumnya tanpa harus mengkompilasi kernel dari awal lagi.

Software tambahan yang dibutuhkan

Untuk melakukan konfigurasi, kompilasi, dan instalasi ala Debian GNU/Linux, ada beberapa *software* tambahan yang anda butuhkan sehingga proses tersebut dapat berjalan lancar.

- kernel-package
- libncurses5-dev
- modutils

```
debian:~# apt-get install kernel-package libncurses5-dev
modutils
```

Kompilasi dan pemaketan kernel dan modul

Proses awal dari instalasi kernel Linux tidak berbeda dengan cara konvensional. Download *source* kernel Linux di <http://www.kernel.org>, kemudian ekstrak dan konfigurasi kernel linux sesuai dengan urutan langkah pada awal bab ini.

Khusus untuk tahap “**make dep && make clean && make && make install**” yang biasa dilakukan dengan cara kompilasi konvensional tidak dibutuhkan lagi saat anda mengkompilasi kernel versi Debian GNU/Linux. Anda cukup menyetikkan perintah berikut:

```
debian:~# make-kpkg kernel-image
debian:~# make-kpkg modules-image
```

Perintah di atas akan mengkompilasi kernel dan modul-modul yang akan disertakan kemudian membangun paket kernel debian “**kernel-image-<versi>_<architecture>.deb**” di luar tempat *source code*nya berada (*/usr/src/*).

Instalasi paket kernel dan modul yang disertakan

Untuk instalasi paket kernel dan modul tersebut, cukup dengan perintah berikut:

```
debian:~# dpkg -i kernel-image-2.4.20_10.00.Custom_i386.deb
```

Selanjutnya sistem akan menanyakan apakah akan membuat *floppy boot disk* <default: no>. Kemudian sistem menanyakan apakah akan menginstal kernel pada blok boot /etc/lilo.conf <default: yes>.

Perintah **dpkg** akan menghasilkan *kernel image* baru (*vmlinuz*) pada direktori */boot*.

Setting Lilo Boot Loader

Untuk menggunakan kernel baru tersebut anda dapat menyettingnya di *path* **/etc/lilo.conf**.

```
image=/boot/vmlinuz
        label=DebianSarge
        read-only
```

Setting Grub Boot Loader

Dengan menggunakan teks editor, buka *file* konfigurasi **/boot/grub/menu.lst**, kemudian tambahkan baris berikut:

```
title GNU/Linux
root (hd0,0)
kernel /boot/vmlinuz root=/dev/hda1 ro vga=791
savedefault
boot
```

Reboot sistem Debian GNU/Linux

Setelah konfigurasi anda lakukan, reboot komputer sehingga anda dapat mencoba menggunakan kernel yang baru saja anda install.

```
debian:~# reboot
```

Anda dapat memeriksa kernel tersebut untuk memastikan apakah telah berjalan dengan sempurna dengan perintah berikut:

```
debian:~# less /var/log/dmesg
```

257

Instalasi kernel pada PC lain dengan tipe sama

Salah satu kelebihan Debian GNU/Linux adalah kepraktisannya. Anda tidak perlu lagi mengkompilasi kernel baru dari awal untuk komputer yang sejenis. Anda hanya mengkopi kernel-imagenya saja kemudian jalankan perintah **dpkg** dan siap untuk digunakan.