

Tutorial Praktis: Membuat CD Live Linux dengan Kernel Sendiri

Kemas Yunus Antonius*

Hadirnya teknologi CD live linux saat ini memberikan kemudahan kepada para pengguna untuk mencoba sistem operasi linux tanpa perlu menginstalnya terlebih dahulu di komputer. Dalam tutorial ini akan dijelaskan secara gamblang langkah-langkah untuk membuat CD live berbasis Slackware linux dengan menggunakan kernel sendiri.

Target pembaca: Pengguna linux lanjutan hingga berpengalaman.

Waktu: 4 jam atau lebih (tergantung kepada spesifikasi PC yang dimiliki).

Apa itu CD live linux?

Mungkin itulah pertanyaan pertama yang hadir di kepala kita semua. Sebenarnya CD live linux adalah sistem operasi linux yang disimpan di sebuah *bootable* CD atau DVD yang kemudian dapat dijalankan langsung dari CD atau DVD *drive*, tanpa harus menginstalnya secara permanen di *hard disk* komputer. Pengguna cukup memasukkan CD live tersebut ke dalam CD atau DVD *drive* dan selanjutnya dapat menjalankan linux tanpa harus mengubah satupun konfigurasi komputernya. Tentunya ini akan menolong pengguna dalam mencoba sistem operasi lain di samping sistem operasi utama yang dimiliki.

Bagaimana cara kerja CD live linux?

Karena sistem operasi linux tersimpan di CD atau DVD, semua proses mulai dari *boot* sampai *shutdown* dilakukan di CD atau DVD tersebut. Ketika komputer *boot* dari CD live, skrip yang terdapat di CD live akan memuat *image* kernel (*vmlinuz*). Selanjutnya, sejumlah *ramdisk* dibuat dalam RAM (*random access memory*) komputer anda. Gunanya sebagai tempat penyimpanan data sementara yang dibutuhkan oleh CD live untuk *boot*. *Image rootdisk* (*initrd.gz*) kemudian dibongkar ke dalamnya dan di-*mount* sebagai sistem *file root*. Sementara direktori-direktori besar (seperti */bin*, */etc*, */lib*, */usr*, */opt*, dan lain-lain) akan di-*mount* langsung dari CD atau DVD *drive*. Dengan begitu beban kerja komputer akan berkurang karena tidak semua data disimpan di memori. Walaupun demikian, sejumlah varian CD live linux menyarankan penggunaanya untuk memiliki RAM sekitar 256 MB – 512 MB untuk menjamin kelancaran dalam pemakaian.

Sekarang hampir semua *distro* linux besar seperti Slackware (<http://www.slackware.com>), Debian (<http://www.debian.org>), Mandriva (<http://www.mandrivalinux.com>), Gentoo (<http://www.gentoo.org>) dan sejumlah turunannya menyediakan CD live siap pakai yang dapat di-*download* dan dicoba oleh para pemula dan pengguna setianya. Biasanya ini dilakukan untuk menguji coba sejumlah *feature* baru yang dintroduksi di versi terbaru distro tersebut secara instan tanpa harus merepotkan penggunaanya mempartisi *hard disk* dan menginstalnya.

CD live linux dengan kernel sendiri

Membuat sebuah CD live linux dengan kernel sendiri dan menggunakan distro linux yang diinginkan tidaklah sesulit yang dibayangkan kebanyakan orang. Dalam kesempatan ini, penulis hendak mengajak pembaca bersama-sama membuat sebuah CD live linux berbasis Slackware linux 10.2 (<http://www.slackware.com>) dengan menggunakan kernel linux (<http://www.kernel.org>) 2.6.16.

Perlu digarisbawahi, metode yang diterapkan di sini merupakan salah satu metode atau teknik yang terdapat saat ini untuk membuat CD live linux. Informasi selanjutnya mengenai metode atau teknik lain dapat dibaca secara lengkap per jenis distro dan sistem operasi di sini, **FrozenTech's LiveCD List** (http://www.livedclist.com/wiki/index.php/LiveCD_Creation_Resources)

Yang dibutuhkan

Sebelum mulai membuat CD live linux, ada baiknya kita mendata sejumlah perkakas yang dibutuhkan:

- Pengetahuan linux dasar
- Sistem operasi linux yang telah diinstal dengan baik (Slackware 10.2), lengkap dengan *build-tools* seperti compiler, linker, make, dan lain-lain.
- Skrip Linux Live yang dapat diperoleh di <http://linux-live.org> (Linux Live 5.4.9)
- *mkisofs* yang dibutuhkan oleh skrip Linux Live untuk membuat file iso dari sistem operasi linux yang ada. Salah satu cara memperoleh mkisofs dengan mudah adalah dengan menginstal paket *cdrtools*.
- Kernel linux 2.6.16
- Unionfs 1.2 (<http://www.fsl.cs.sunysb.edu/project-unionfs.html>)
- Squashfs 3.0 (<http://squashfs.sourceforge.net/>)

Langkah-langkah

Setelah semua perkakas didata, saatnya sekarang kita memulai pekerjaan yang sesungguhnya.

1. Instal dan konfigurasi sistem operasi linux sesuai dengan kebutuhan dan keinginan. Karena apa yang telah dikonfigurasi ini yang akan menjadi basis CD live linux anda nantinya. Jika anda menginginkan CD live linux yang akan dibuat ini berfungsi sebagai *desktop*, maka aplikasi-aplikasi rumah dan perkantoranlah yang diinstal dan dikonfigurasi. Demikian pula untuk fungsi-fungsi lainnya disesuaikan dengan tujuan anda membuat CD live linux tadi.
2. Selanjutnya mempersiapkan direktori kerja.
mkdir /tmp/remaster
cd /tmp/remaster
3. Kemudian download sejumlah komponen yang dibutuhkan untuk seterusnya diekstrak di direktori kerja.
 - a. Skrip Linux Live 5.4.9
wget http://www.linux-live.org/dl/linux-live-5.4.9.tar.gz
tar -xvzf linux-live-5.4.9.tar.gz
 - b. Unionfs 1.2
wget ftp://ftp.fsl.cs.sunysb.edu/pub/unionfs/unionfs-1.2.tar.gz
tar -xvzf unionfs-1.2.tar.gz
 - c. Squashfs 3.0
wget http://jaist.dl.sourceforge.net/sourceforge/squashfs/squashfs3.0.tar.gz
tar -xvzf squashfs3.0.tar.gz

- d. Kernel 2.6.16
wget http://kernel.org/pub/linux/kernel/v2.6/linux-2.6.16.tar.bz2
tar -xvzf linux-2.6.16.tar.bz2

4. Mengkompilasi kernel.

Sebelum proses kompilasi kernel dimulai, kita harus melakukan sejumlah *patch* yang diperlukan agar kernel dapat bekerja di lingkungan CD live.

- a. *Patch* unionfs.
cd /tmp/remaster/unionfs-1.2
./patch-kernel.sh /tmp/remaster/linux-2.6.16

Setelah *patch* berhasil, anda akan menemukan sebuah direktori baru yang berisikan banyak file di /tmp/remaster/linux-2.6.16/fs/unionfs

- b. *Patch* squashfs.
cd /tmp/remaster/linux-2.6.16
patch -p1 < /tmp/remaster/squashfs3.0/linux-2.6.15/squashfs3.0-patch

- c. Mengkonfigurasi kernel.
Untuk ini kita memerlukan sejumlah pengaturan sehingga kernel dapat bekerja dalam lingkungan CD live. Cara paling mudah dan aman untuk mengkonfigurasi kernel adalah dengan menggunakan berkas *config* kernel yang ada di skrip Linux Live. Pengaturan yang terdapat di sana akan mengaktifkan unionfs, squashfs dan berapa komponen lainnya yang dibutuhkan oleh CD live.

```
# cp /tmp/remaster/linux-live-5.4.9/ DOC/config/.config.kernel /tmp/remaster/linux-2.6.16  
# make menuconfig
```

Sekarang lakukan perubahan yang kita inginkan pada kernel, tapi pastikan sejumlah parameter memiliki nilai seperti yang ditunjukkan di bawah ini:

```
CONFIG_EXT2_FS=y  
CONFIG_TMPFS=y  
CONFIG_BLK_DEV_RAM=y  
CONFIG_BLK_DEV_INITRD=y
```

Setelah selesai melakukan konfigurasi, kita simpan perubahan-perubahan yang telah dibuat dengan memilih opsi keluar *config-tool*. Selanjutnya akan ditanya apakah kita ingin menyimpan konfigurasi yang baru, konfirmasilah dengan jawaban **Y(es)**.

Kembali ke *shell*, masukkan perintah berikut untuk melakukan proses kompilasi:

```
# make
```

Proses kompilasi pun dimulai dan biasanya akan memakan cukup lama bergantung kepada kinerja dari komputer. Oleh karena itu kita dapat melakukan aktivitas lainnya sambil menunggu proses kompilasi selesai.

5. Menginstal kernel.

Setelah kernel terkompilasi, kita instal kernel yang baru untuk menggantikan kernel yang lama. (Perintah di bawah ini dapat berbeda dari situasi yang sebenarnya. Disarankan untuk mengecek terlebih dahulu semua file yang ada di direktori **/boot** sebelum melakukan penggantian. Sesuaikanlah nama file yang ditemui untuk setiap perintah yang dijalankan.)

```
# mv /boot/System.map /boot/System.map.old
# cp /tmp/remaster/linux-2.6.16/System.map /boot/System.map
# mv /boot/vmlinuz /boot/vmlinuz.old
# cp /tmp/remaster/linux-2.6.16/arch/i386/boot/bzImage /boot/vmlinuz
# mv /boot/config/boot/config.old
# cp /tmp/remaster/linux-2.6.16/.config /boot/config
```

Kemudian instal semua modul kernel dengan menjalankan perintah berikut:

```
# make modules_install
```

Langkah selanjutnya adalah mengkonfigurasi ulang *boot manager* linux (contoh: **lilo**) guna mendukung kernel yang baru. Berikut contoh sebuah file **lilo.conf** yang dapat ditemukan di direktori **/etc**.

```
boot = /dev/hda
prompt
timeout = 300
vga=791
```

```
image = /boot/vmlinuz
  root = /dev/hda1
  label = Slackware
  read-write
```

Perlu diperhatikan, setiap selesai mengedit file **lilo.conf**, jangan lupa untuk menyimpannya dan kemudian mengeksekusi perintah:

```
# lilo -v
```

untuk mengaktifkan perubahan.

6. Menguji coba kernel baru.

Setelah kompilasi dan instalasi kernel di atas, kinilah saatnya untuk mengadakan uji coba kernel yang baru. *Reboot* komputer, tunggu sampai mesinnya hidup kembali dan menjalankan sistem operasi linux sampai selesai. Lalu di *shell*, cek versi kernel linux yang dipakai:

```
# uname -a
```

Jika semua berjalan dengan lancar, perintah tadi akan menunjukkan versi kernel linux yang benar, yaitu 2.6.16.

7. Mengkompilasi dan menginstal unionfs.

```
# cd /tmp/remaster/unionfs-1.2  
# touch fistdev.mk
```

Kemudian dengan bantuan sebuah editor teks, *copy paste* perintah berikut ke dalam file **fistdev.mk**.

```
EXTRACFLAGS=-DUNIONFS_NDEBUG  
UNIONFS_DEBUG_CFLAG=
```

Perintah ini sengaja kita pakai supaya modul unionfs yang akan dikompilasi berukuran kecil dengan menghilangkan hal-hal yang tidak diperlukan. Selanjutnya proses kompilasi unionfs dan instalasi dapat dilakukan.

```
# make  
# make install
```

8. Mengkompilasi dan menginstal squashfs-tools.

Sebenarnya *squashfs-tools* yang diperlukan telah dibuat secara otomatis sewaktu men-*patch* kernel seperti dideskripsikan pada langkah **4b** di atas. Kita dapat memeriksanya dengan membuka direktori **/usr/sbin** dan mencari file yang bernama **mksquashfs**. Jika ditemukan, kita dapat langsung menuju ke langkah 9. Jika tidak ditemukan, kita harus mengkompilasi dan menginstalnya secara manual.

```
# cd /tmp/remaster/squashfs3.0/squashfs-tools  
# make  
# make install  
# cp /tmp/remaster/squashfs3.0/squashfs-tools/mksquashfs /usr/sbin/mksquashfs
```

9. Memodifikasi skrip Linux Live dan lain-lain.

Saat ini kita sudah mendekati langkah-langkah terakhir dalam pembuatan CD live linux sendiri. Adapun modifikasi yang dimaksud adalah:

- a. Memindahkan unionfs dan squashfs ke skrip Linux Live.

```
# mkdir /tmp/remaster/linux-live-5.4.9/initrd/kernel-modules/2.6.16
# cp /lib/modules/2.6.16/kernel/fs/unionfs.ko /tmp/remaster/linux-live-
5.4.9/initrd/kernel-modules/2.6.16/
# cp /lib/modules/2.6.16/kernel/fs/squashfs/squashfs.ko /tmp/remaster/linux-live-
5.4.9/initrd/kernel-modules/2.6.16/
```

- b. Mengedit sejumlah pengaturan yang terdapat di skrip Linux Live

Berdasarkan pengalaman pribadi penulis, terkadang CD live linux buatan sendiri tidak dapat bekerja dengan baik disebabkan ukuran *ramdisk* yang terlalu kecil. Oleh karena itu, kita harus memodifikasi dua file berikut:

- Buka file **/tmp/remaster/linux-live-5.4.9/config** dengan sebuah editor teks dan ubah garis

```
RAM0SIZE=4444
menjadi
RAM0SIZE=8888
```

- Buka file **/tmp/remaster/linux-live-5.4.9/cd-root/isolinux.cfg** dan ubah pernyataan

```
ramdisk_size=4444
menjadi
ramdisk_size=8888
```

- c. Kalau anda ingin menggunakan resolusi layar yang lebih tinggi, anda dapat mengubah mode grafik sewaktu *boot*. Sebagai contoh untuk resolusi 1024x768x256, cukup tambahkan **vga=773** pada garis **append** dalam file *isolinux.cfg* di skrip Linux Live.

10. Eksekusi skrip Linux Live.

```
# cd /tmp/remaster/linux-live-5.4.9/
# ./runme.sh
```

Tergantung pada kinerja komputer yang dimiliki, cepat atau lambat file **livecd.iso** akan dihasilkan di **/tmp**.

11. Bakar dan jalankan!

Dengan menggunakan program pembakar CD/DVD yang ada, bakarlah (*burning*) file **livecd.iso** ke CD kosong. Selanjutnya *reboot* komputer dengan CD masih di dalam CD atau DVD *drive*. Namun sebelumnya, pastikan komputer mendukung *boot* dari CD atau DVD. Jika semua berjalan dengan baik, sistem operasi linux yang disimpan dalam CD tadi akan dijalankan.

Alternatif lain adalah dengan menggunakan program emulasi seperti QEMU (<http://fabrice.bellard.free.fr/qemu/>) di sistem operasi yang masih dipakai. File iso tadi akan dijalankan seperti di lingkungan yang sebenarnya. Bedanya kita tidak perlu membakar ke CD terlebih dahulu dan me-*restart* komputer. Perintah sederhana yang dapat dipakai dalam *shell* berupa:

```
# qemu -m 128 -cdrom /tmp/livecd.iso -boot d -user-net
```

Kesimpulan

Berkat teknologi CD live kita dapat menjalankan sistem operasi linux dari sebuah CD atau DVD. Satu hal yang dulunya mungkin di luar bayangan kebanyakan orang, sekarang telah menjadi kenyataan. Berbagai macam teknik yang tersedia dewasa ini memberikan kebebasan kepada pengguna memilih dan menggunakan metode mana yang terbaik untuknya. Tutorial ini mencoba menuntun pembaca membuat CD live sendiri dengan menerapkan teknik yang diperoleh di <http://www.linux-live.org> dan menggunakan Slackware linux sebagai basis distro. Akhir kata, selamat mencoba dan semoga berhasil!

Disclaimer

Tutorial ini disusun sedemikian rupa dengan penuh kehati-hatian dan telah diuji coba sebelumnya. Walaupun demikian, penulis tidak bertanggung jawab terhadap apapun yang terjadi setelah pembaca mengikuti semua langkah yang dijelaskan dalam tulisan ini. Semua akibat dan konsekuensinya merupakan tanggung jawab pribadi pembaca.



Info Penulis

* **Kemas Yunus Antonius** adalah seorang konsultan TI. Merupakan salah seorang aktivis IlmuKomputer.Com dan kontributor di sejumlah komunitas open source. Pengembang distro linux SLAMPP (<http://slampp.abangadek.com>), solusi sederhana untuk server rumah.

Website: <http://www.kyantoni.us>

Jakarta, 8 Agustus 2006

Referensi:

- Slackware Linux: <http://www.slackware.com>
- Kernel Linux: <http://www.kernel.org>
- Skrip Linux Live: <http://www.linux-live.org>
- SLAX: <http://www.slax.org>
- Unionfs: <http://www.unionfs.org/>
- Squashfs: <http://squashfs.sourceforge.net/>
- CD live di Wikipedia: http://en.wikipedia.org/wiki/Live_CD
- Teknik pembuatan CD live:
http://www.livecdlist.com/wiki/index.php/LiveCD_Creation_Resources
- QEMU: (<http://fabrice.bellard.free.fr/qemu/>)