

Emulasi Komputer dengan Bochs

Di dunia *open source*, kita mengenal salah satu emulator komputer yang hebat. Namanya adalah Bochs. Dengan menggunakan Bochs, kita bisa mengemulasikan komputer di atas komputer. Kita bisa menginstal sistem operasi lain di atas sistem operasi yang sedang berjalan. Tulisan ini akan membahas tentang Bochs dan beberapa isu seputar penggunaannya.

Bagi pengguna komputer yang bekerja di lebih dari satu sistem operasi, instalasi dua (katakanlah Windows dan Linux) atau lebih sistem operasi dalam satu harddisk adalah hal yang wajar. Terantung sistem operasi mana yang diperlukan, sistem operasi tersebutlah yang di boot. Apabila ingin berganti sistem, maka sistem yang aktif akan di-*reboot* untuk selanjutnya berpindah ke sistem lainnya.

Kondisi semacam ini memang kondisi yang sangat umum ditemukan bagi pengguna lebih dari satu sistem operasi. Tak jarang, sebuah partisi tambahan dibuat sebagai partisi untuk menyimpan data bersama yang bisa ditulis oleh sistem operasi-sistem operasi yang digunakan.

Masalah baru akan timbul ketika ada kebutuhan untuk menggunakan dua sistem operasi bersama-sama. Hal ini tidak dimungkinkan dengan sistem dual boot (atau lebih) tersebut. Pendekatan dengan menggunakan wine (atau pun variannya) terkadang juga tidak bisa membantu banyak. Tidak semua aplikasi bisa diemulasikan dengan wine. Walaupun bisa, bagaimanapun juga, aplikasi yang berjalan di Wine tidak akan sebaik ketika berjalan di sistem asalnya. Selain itu, bagaimana kalau sistem operasi utama adalah Windows?

Di sinilah emulator komputer diperlukan. Dengan menggunakan emulasi komputer, kita bagaikan memiliki banyak komputer yang sama-sama sedang berjalan, namun tetap bisa saling bekerja sama. Dan, secara fisik, kita hanya memiliki satu komputer saja. Di dalam setiap emulator, kita bisa menginstall sistem operasi sendiri pada harddisk virtual (yang bagi sistem operasi yang menjalankan emulator, umumnya berupa sebuah file berukuran besar).

Di dunia ini, kita mengenal banyak sekali emulator komputer. Salah satu yang terbaik yang mungkin pernah ada adalah VMWare. Namun, *software* seharga 200-an dollar Amerika Serikat tersebut bagi beberapa kalangan pengguna sangatlah mahal. Terutama, kalau emulasi digunakan untuk sistem operasi yang sudah lama (dan harganya sendiri juga sudah murah) seperti DOS ataupun Windows 95.

Bagi pengguna yang mencari alternatif emulator yang murah (bahkan gratis!), dapat diinstal pada berbagai sistem operasi (multiplatform), dan dapat mengemulasikan beberapa sistem operasi, maka Bochs adalah salah satu alternatif yang terbaik yang ada.

Pengenalan Bochs

Bochs adalah emulator komputer. Sebuah *software* yang mampu mengemulasikan berbagai *hardware* yang menyusun sebuah komputer yang bekerja penuh. Bochs mampu mengemulasikan processor, memory, harddisk, sound card, vga card, dan lain sebagainya. Ketika dijalankan, Bochs akan menampilkan sebuah konsol yang mensimulasikan komputer yang dinyalakan. Kita bisa memasukkan CDROM installer suatu sistem operasi, kita bisa menginstall sistem operasi tersebut di harddisk yang telah diemulasikan oleh Bochs. Setelah instalasi selesai, Bochs bisa boot ke dalam sistem operasi tersebut dan sistem operasi tersebut bekerja seolah-olah sedang berjalan di komputer yang sesungguhnya.

Arsitektur komputer

Lebih lanjut lagi, Bochs adalah emulator komputer x86. Bochs dapat dikonfigur un-

tuk bertingkah laku seperti komputer 386, 486, Pentium ataupun Pentium Pro (dukungan tidak lengkap). Bochs tidak dapat digunakan untuk mengemulasikan mesin Sparc atau mengemulasikan mesin Apple misalnya.

Namun, Bochs sendiri adalah emulator yang dapat berjalan pada berbagai sistem operasi. Bochs dapat diinstal pada berbagai sistem operasi yang berjalan pada berbagai hardware ataupun arsitektur komputer. Misal, Bochs dapat berjalan pada Linux yang berjalan pada mesin Sun. Namun, pada proses emulasi, Bochs tetap akan mengemulasikan hardware dengan arsitektur x86.

Emulasi hardware

Dengan kata lain, Bochs tidak peduli akan hardware yang berjalan sistem operasi host (sistem operasi yang menjalankan bochs). Selama hardware host dapat dikenali dengan baik oleh sistem operasi, maka bochs dapat berkomunikasi dengan hardware tersebut (secara teoritis).

Kondisi demikian bisa dilakukan karena Bochs telah mengemulasikan jenis hardware tertentu. Bochs tidak perlu mengemulasikan hardware host. Bochs memiliki 'hardware' sendiri. Berikut ini adalah beberapa hardware yang diemulasikan oleh Bochs:

- Processor: 386, 486, 586 atau yang kompatibel
- Harddisk: IDE Harddisk ATA-2
- CDROM: IDE CDROM ATAPI-4
- Keyboard: Keyboard PS/2 standar dengan keymap Amerika Utara
- Mouse: Mouse PS/2
- Sound card: Sound Blaster 16

- Network card: NE2000 atau yang kompatibel

Satu hal yang luar biasa tentang Bochs adalah kemampuan untuk mengemulasikan SMP (Symetric Multi Processor) sampai 15 processor. Walaupun, harus diakui, fitur ini masih sangat eksperimental. Satu catatan untuk emulasi SMP adalah, walaupun kita bisa menentukan jumlah processor, tidak berarti kalau emulasi akan berjalan lebih cepat.

Sebagai emulator open source, bochs memang sudah cukup hebat dalam melakukan emulasi berbagai hardware. Memang, harus diakui pula, tidak semua emulasi hardware dapat berjalan dengan baik, di semua sistem operasi host.

Kemampuan emulasi CDROM di Bochs misalnya. Emulasi CDROM cukup bagus di Bochs. CDROM dapat dibaca dari ISO image untuk berbagai sistem operasi. Di berbagai sistem operasi, Bochs juga mampu untuk membaca langsung *device* CDROM. Bahkan, Bochs juga bisa melakukan booting dari bootable CDROM ataupun bootable CDROM image.

Namun, kemampuan Bochs dalam mengemulasikan Sound card tidaklah sebaik CDROM. Memang, di beberapa platform seperti Windows dan Linux, sound bisa bekerja. Namun, tidak dengan sound di Mac OSX.

Dukungan untuk USB bahkan lebih jelek lagi. Saat ini, dukungan USB masih belum lengkap.

Jadi, kita tidak bisa mengharapkan Bochs untuk mampu mengemulasi semua harddisk yang ada dengan baik, di semua *platform*. Terkadang, sebuah hardware diemulasikan

dengan baik di satu platform, namun tidak di platform lain.

Fitur lainnya

Berikut ini adalah fitur Bochs selain yang dibicarakan sebelumnya:

- Memiliki command line debugger, yang dapat digunakan untuk melakukan debugging. Sangat berguna bagi yang ingin mempelajari sistem operasi lebih jauh lagi.
- Mendukung floating point melalui software floating point.
- Dukungan VGA.
- Dukungan VESA sampai 1024x768 32bpp.
- Mendukung floppy disk sampai 2,88MB.
- Mendukung 4 channel ATA, atau 8 device ATA (8 harddisk pada umumnya).
- Mendukung paralel port.
- Mendukung serial port.
- Mendukung game port.
- Mendukung 16/32 bit addressing.
- Bisa copy dan paste antara host OS dan guest OS.

Istilah yang dipergunakan

Bochs dan emulator komputer lainnya umumnya mengenal istilah-istilah:

- Host, sistem operasi atau mesin yang menjalankan bochs
- Guest, sistem operasi atau mesin yang diemulasikan.

Lisensi Bochs

Pada awalnya (Bochs dimulai pada tahun 1994), Bochs merupakan proyek komersial.

Pada tahun 2000, MandrakeSoft membeli Bochs dan menjadikan Bochs proyek open source di bawah lisensi LGPL. Lisensi ini menjadikan Bochs sangat fleksibel baik bagi kalangan pengguna biasa ataupun kalangan industri.

Satu yang terakhir, sebelum kita berpindah dari pengenalan, Bochs bisa diucapkan sebagai box. Nama Bochs merupakan permainan kata-kata dari Box. Nama Bochs ini terpikir karena umumnya pengguna menyebut sistem operasinya sebagai box, misal: Linux box, Windows box, dan lain sebagainya.

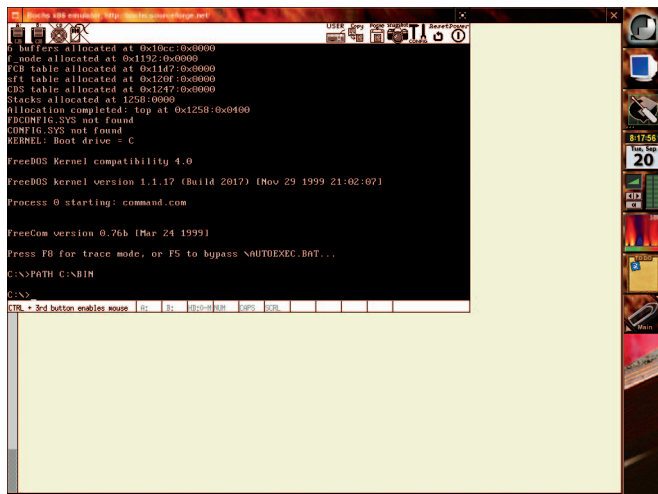
Kegunaan Bochs

Berikut ini, kita akan melihat kapan kita membutuhkan Bochs. Yang pertama adalah seperti yang disebutkan sebelumnya. Kita perlu bekerja pada dua sistem operasi sekaligus.

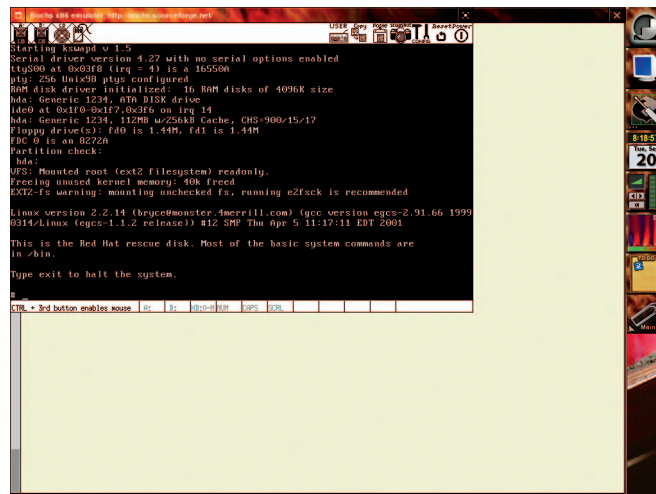
Bekerja pada dua sistem operasi sekaligus tidak harus selalu dalam pengertian yang serius. Misal, Anda ingin bermain game Windows sementara sedang bekerja di Linux (*refreshing*).

Alasan kedua adalah katakanlah Anda masih ingin mempertahankan program lama sementara, sistem operasi baru di komputer baru Anda sudah tidak mendukung lagi. Misal, Anda ingin menjalankan aplikasi DOS yang lama sementara untuk aplikasi tersebut, Windows XP yang Anda gunakan sudah tidak mendukung.

Alasan ketiga adalah belajar. Bagi yang ingin mendalami sistem operasi di era sekarang, menggunakan Bochs merupakan salah satu cara yang sangat membantu. Bochs bisa



Emulasi FreeDOS.



Emulasi Linux Redhat Rescue disk.

mengemulasikan banyak sistem operasi. Proses belajar tidak harus selalu diartikan secara akademis. Anda bisa menggunakan Bochs untuk *me-reverse engineering* sebuah driver misalnya.

Alasan keempat adalah alasan kesenangan. Kengengahan Anda akan sistem DOS yang Anda gunakan ketika kali pertama belajar komputer? Atau, barangkali Anda ingin menginstall kembali Slackware 4.0 atau Debian 2 atau Redhat 6 Anda yang beberapa tahun lalu Anda gunakan? Gunakan Bochs.

Bochs dan VMWare

Bochs mirip-mirip dengan VMWare. Namun, secara umum, Bochs tidak bisa dibandingkan dengan VMWare. VMWare sendiri adalah produk dari perusahaan besar dengan dana dan sumber daya manusia serta perencanaan yang telah diusahakan dengan baik. Ada target untuk rilis-rilis VMWare. Sebagai hasilnya, VMWare dijual dengan sangat mahal.

Sementara, Bochs adalah salah satu proyek emulator open source yang mungkin tidak memiliki dana sekencah perusahaan pembuat VMWare. Sumber daya manusianya pun, walaupun hebat, mungkin tidak bekerja fulltime untuk Bochs. Dan, perencanaan Bochs pun bergantung pada para pengembangnya ataupun komunitas. Tidak ada target keras tertentu untuk rilis-rilis Bochs. Sebagai hasilnya, Bochs bisa digunakan tanpa harus membayar sepeser pun.

Bochs tidak datang dengan *graphical user interface* sebagai VMWare. Sebagai gantinya, Bochs datang dengan command line interface.

Bochs juga tidak memiliki dukungan hardware yang matang untuk semua platform. Terkadang, perbedaannya terlalu kentara. Bahkan, USB pun tidak didukung dengan baik, untuk semua platform.

Bochs juga tidak memiliki integrasi yang tinggi dengan sistem operasi host. VMWare selain datang dengan modul untuk kernel berbagai distro (sampai versi-versinya), masih mampu mengkompilasi modul untuk kernel yang tidak standar atau yang lebih baru.

Singkat kata, Bochs dan VMWare dalam beberapa hal memang tidak bisa dibandingkan. Namun, Bochs juga memiliki sejumlah kelebihan seperti gratis, source code yang terbuka, lisensi yang fleksibel, dan hal-hal lainnya. Yang terpenting, kalau permasalahan bisa diselesaikan dengan Bochs, kita tidak perlu menggunakan VMWare.

Instalasi Bochs

Bochs bisa di-download di website-nya di bochs.sf.net. Namun, beberapa distribusi sudah datang dengan Bochs di dalam CDROM atau DVDROM distro. Sebelum men-download source dan melakukan kompilasi sendiri, cobalah untuk menggunakan paket yang telah disiapkan untuk distribusi Anda.

Sebagai catatan, tulisan ini dibuat di distro Linux SUSE Linux Pro 9.3, dengan Bochs yang di-upgrade menjadi versi 2.2.1.

Setelah terinstal, Bochs akan datang dengan tiga program:

- Bochs: emulator itu sendiri.
- bxImage: tool untuk membuat disk image untuk digunakan pada Bochs. Ketika

menggunakan Bochs, kita perlu membuat harddisk 'virtual' di mana Bochs akan menganggapnya sebagai harddisk. Dari sisi pandang host, harddisk virtual tersebut hanyalah sebuah file image biasa, yang dibuat dengan bximage.

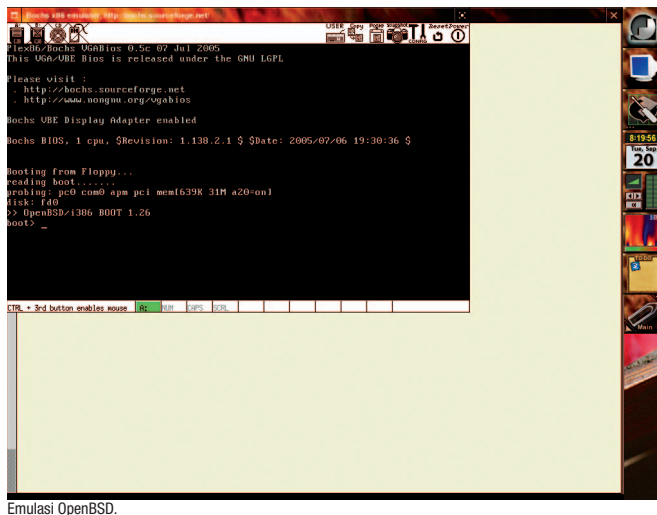
- bxcommit: tool interaktif untuk mengcommit redolog ke disk image untuk digunakan oleh Bochs.
- Bochs-dlx: Menjalankan demo DLX Linux

Setelah bochs terinstal, kita bisa melanjutkan ke langkah berikutnya: memulai bochs dengan image contoh DLX Linux.

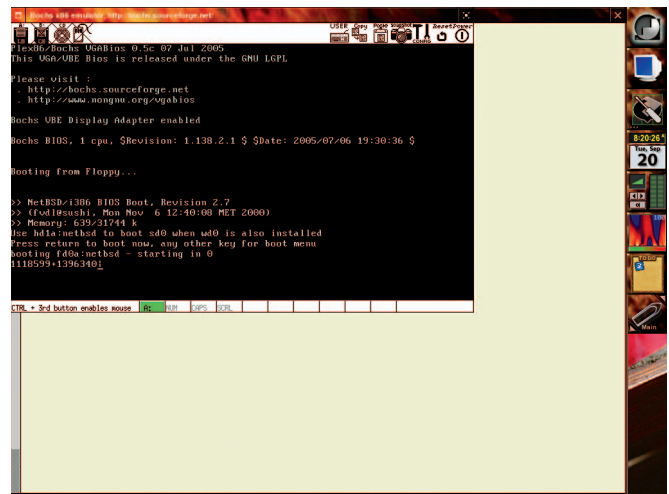
Menjalankan DLX Linux

Bochs datang dengan sebuah image DLX Linux yang siap untuk dijalankan. Untuk menjalankan, berikanlah perintah `bochs-dlx`.

```
$ bochs-dlx
-----
DLX Linux Demo, for
Bochs x86 Emulator
-----
Checking for bochs binary...ok
Checking for DLX linux
directory...ok
Checking for /usr/bin/gzip...ok
Checking for /home/nop/.bochsdlx
directory...
-----
To run the DLX Linux demo, I
```



Emulasi OpenBSD.



Emulasi NetBSD.

```
need to create a directory
called
/home/nop/.bochsdlx, and copy
some configuration files
and a 10 megabyte disk image
into the directory.
-----
-----
Is that okay? [y/n]
```

Kali pertama dijalankan, script bochsdlx akan membuat sebuah direktori ~/.bochsdlx dan mengekstrak file-file yang dibutuhkan ke dalamnya. Jawablah y pada saat konfirmasi untuk membuat direktori diminta. Setelah itu, secara otomatis emulator yang berisi DLX Linux akan dijalankan.

Sebuah window akan ditampilkan. Emulasi pun akan dimulai. Setelah DLX Linux selesai boot, pengguna akan menjumpai sebuah login prompt. Login-lah sebagai root (tanpa *password*) untuk memulai penggunaan. Pengguna kemudian bisa bekerja seperti biasa.

Untuk menutup sesi sistem, pengguna dapat *logout* dan klik langsung pada tombol Power yang terletak pada kanan atas dialog.

Sampai di sini, pengguna telah dibawa untuk melihat kemampuan Bochs untuk mengemulasi Linux di atas Linux. Berikut ini, kita akan melanjutkan ke penggunaan yang lebih menarik, yaitu mencoba berbagai disk image yang disediakan.

Memulai Bochs dengan image yang disediakan

Berkat kontribusi dari berbagai pengguna dan pengembang Bochs di seluruh dunia, di website Bochs, kita bisa menemukan berbagai disk images siap pakai. Ukuran dari disk images tersebut beraneka ragam, mulai dari yang berukuran di bawah 5 MB sampai yang mendekati 100 MB. Sistem operasinya pun macam-macam, mulai dari Linux, *BSD sampai GNU Hurd.

Mencoba Bochs yang paling mudah adalah dengan menggunakan disk image yang sudah ada. Dengan demikian, kita tidak perlu:

- Membuat disk image sendiri.
- Melakukan instalasi dan konfigurasi sendiri.

FreeDOS

Sebagai contoh pertama, kita akan mencoba disk image berisikan sistem freedos. Download-lah file freedos-img.tar.gz dan lakukanlah langkah-langkah berikut ini untuk memulai emulasi:

- Ekstraklah freedos-img.tar.gz tersebut dengan perintah:

```
$ tar zxvf freedos-img.tar.gz
freedos-img/
freedos-img/a.img
freedos-img/b.img
freedos-img/c.img
freedos-img/bochsrc
```

- Masuklah ke direktori freedos-img

```
$ cd freedos-img
```

- Jalankanlah perintah berikut untuk menjalankan Bochs:

```
$ bochs -q
```

Selanjutnya, sebuah window akan ditampilkan. Setelah FreeDOS selesai booting, pengguna segera bisa bekerja dengan DOS. Kliklah pada tombol Power begitu penggunaan selesai.

Linux

Image kedua yang akan kita coba adalah image distribusi Linux (rescue disk Red Hat dengan kernel 2.2.14). Image linux-img.tar.gz ini berukuran cukup kecil, yaitu 2 MB. Lakukanlah langkah-langkah berikut ini setelah download linux-img.tar.gz selesai dilakukan:

- Ekstraklah linux-img.tar.gz tersebut dengan perintah:

```
$ tar zxvf linux-img.tar.gz
linux-img/
linux-img/README
linux-img/TESTFORM.txt
linux-img/minibootable.img
linux-img/bochsrc
```

- Masuklah ke direktori hasil ekstraksi:

```
$ cd linux-img
```

- Berikan perintah berikut untuk menjalankan emulator:

```
$ bochs -q
```

OpenBSD

Berikut ini, kita akan membuktikan kemam-

puan bochs dalam mengemulasi OpenBSD. Download-lah terlebih dahulu image openbsd-img.tar.gz yang hanya berukuran sekitar 1.3 M. Setelah itu, lakukanlah langkah-langkah berikut:

- Ekstraklah openbsd-img.tar.gz tersebut dengan perintah:

```
$ tar zxvf openbsd-img.tar.gz
openbsd-img/
openbsd-img/a.img
openbsd-img/README
openbsd-img/TESTFORM.txt
openbsd-img/bochsrc
```

- Masuklah ke direktori hasil ekstraksi:

```
$ cd openbsd-img
```

- Berikanlah perintah berikut ini untuk menjalankan emulator:

```
$ bochs -q
```

Sebuah dialog akan ditampilkan. Sedikit berbeda, pada prompt yang ditampilkan, tekanlah tombol ENTER. Booting akan dilakukan menggunakan image disket.

NetBSD

Masih berada dalam keluarga BSD, kali ini kita akan mencoba emulasi NetBSD. Download-lah image netbsd-img.tar.gz yang berukuran 1.2 MB. Setelah itu, lakukanlah langkah-langkah berikut:

- Ekstraklah netbsd-img.tar.gz tersebut dengan perintah:

```
$ tar zxvf netbsd-img.tar.gz
netbsd-img/
netbsd-img/a.img
netbsd-img/README
netbsd-img/TESTFORM.txt
netbsd-img/bochsrc
```

- Masuklah ke direktori hasil ekstraksi:

```
$ cd netbsd-img
```

- Berikanlah perintah berikut ini untuk menjalankan emulator:

```
$ bochs -q
```

Sebuah dialog akan ditampilkan. Proses booting akan dilakukan dengan media floppy. Setelah itu, NetBSD mini siap digunakan. Setelah mencoba beberapa image yang disediakan, kita akan mencoba untuk membuat disk image sendiri,

kemudian menginstalasi sendiri sistem operasi yang diinginkan, untuk kemudian menikmati kemampuan emulasi Bochs.

Membuat disk image dan menginstalasi OS

Pada bagian ini, kita akan mempergunakan Bochs untuk menginstal dan menjalankan Linux Debian 3.1 (hanya base system saja). Sebelum kita bisa memulai, kita perlu membuat disk image terlebih dahulu. Untuk kebutuhan sistem Debian 3.1, kita akan menyediakan disk space sekitar 150 MB.

Langkah 1: Membuat disk image

Gunakan program bimage untuk membuat disk image. Buatlah hd image (flat) dengan ukuran 150 MB dan disimpan sebagai debian31.img. Kemudian, catatlah baris-baris terakhir dari bimage seperti contoh berikut ini:

```
ata0-master: type=disk,
path="debian31.img", mode=flat,
cylinders=304, heads=16, spt=63
```

Setelah itu, perhatikanlah direktori aktif Anda. Seharusnya, akan terdapat sebuah file debian31.img yang berukuran 150 MB.

Sampai di sini, langkah 1 untuk membuat disk image telah selesai. Kita siap melanjutkan ke langkah berikutnya.

Langkah 2: Konfigurasi bochsrc

Apabila diperhatikan, pada pembahasan mengenai penggunaan disk image yang tersedia:

- Kita selalu hanya menjalankan bochs dengan argumen -q. Sederhana sekali, tanpa memberikan argumen yang berarti.
- Di setiap arsip, selalu tersedia file dengan nama .bochsrc, bochsrc, atau bochsrc.txt. File-file tersebut adalah file konfigurasi bochs untuk setiap image. Untuk image yang berbeda, file konfigurasi yang digunakan juga tentu berbeda. Kita bisa berpikir file konfigurasi tersebut merupakan file kontrol untuk setiap sistem yang ingin kita emulasikan. Dengan menggunakan file konfigurasi, kita bisa menjalankan Bochs tanpa argumen sama sekali.

File konfigurasi akan dicari dalam urutan berikut ini:

- Mencari .bochsrc di dalam direktori aktif.
- Mencari bochsrc di dalam direktori aktif.
- Mencari bochsrc.txt di dalam direktori aktif.
- Mencari bochsrc.bxrc di dalam direktori aktif (hanya untuk Windows).
- Mencari .bochsrc di dalam home directory user (hanya untuk Unix).
- Mencari bochsrc di /etc (hanya untuk Unix).

Apabila diinginkan, kita pun bisa memberikan nama yang berbeda untuk file konfigurasi. Namun, dengan demikian, kita perlu memberikan argumen -f disertai lokasi file konfigurasi yang diinginkan. Bochs sangatlah fleksibel.

File konfigurasi Bochs harus diakui cukup rumit. Namun, pada dasarnya, kita tidak perlu menggunakan semua opsi yang tersedia. Dalam sistem yang umum, kita hanya perlu melakukan beberapa pengaturan berikut:

- Jumlah memori yang diinginkan. Sebaiknya, kita memberikan nilai yang masuk akal. Sepertiga dari memory sistem adalah nilai yang masuk akal. Opsi yang dipergunakan adalah megs, sementara, nilai yang diperbolehkan adalah integer. Contoh:

```
megs: 64
```

- Nama file ROMBIOS

Opsi yang dipergunakan adalah romimage, sementara, nilai yang diperbolehkan adalah path ke rombios yang disertakan dalam distribusi bochs. Contoh:

```
romimage: file=/usr/share/bochs/BIOS-bochs-latest,
address=0xf0000
```

- Nama file VGABIOS

Opsi yang dipergunakan adalah vga-romimage. Nilai yang diperbolehkan adalah path ke vgarombios yang disertakan dalam distribusi bochs. Contoh:

```
vgaromimage: file=/usr/share/bochs/VGABIOS-lgpl-latest
```

- Ekstensi VGA yang akan dipergunakan
- Opsi yang dipergunakan adalah vga: extension=<nama ekstensi>. Umumnya, pengguna akan mengaktifkan ekstensi VBE. Contoh:

```
vga: extension=vbe
```

- Floppy (apabila diperlukan)

Opsi yang dipergunakan adalah floppy (untuk floppy pertama). Nilai yang diperbolehkan adalah path ke image floppy ataupun lokasi device floppy pertama di Linux (/dev/fd0). Kita bisa pula memberikan opsi status dengan nilai inserted (floppy telah dimasukkan) ataupun ejected. Contoh:



Instalasi Debian 3.1 di atas Bochs.

```
floppya: 1_44=/dev/fd0,
status=ejected
```

- **Harddisk utama**

Opsi yang dipergunakan adalah ata0-master (umumnya). Gunakan keluaran dari perintah `bximage`. Contoh:

```
ata0-master: type=disk,
path="debian31.img",
mode=flat, cylinders=304,
heads=16, spt=63
```

Contoh ini menunjukkan kita akan menggunakan harddisk primary master dengan nama file `debian31.img`.

- **CDROM**

Opsi yang dipergunakan adalah salah satu dari `ata<n>-<master/slave>`. N menunjukkan urutan device. Nilai yang mungkin adalah 0 sampai 3. Sementara, master atau slave menunjukkan status disk pada sebuah ATA channel.

Dengan demikian, apabila cdrom diumpamakan ditancapkan pada primary slave, maka berikanlah opsi ini:

```
ata0-slave: type=cdrom,
path="netinst.iso",
status=inserted
```

Perhatikanlah opsi type antara harddisk (disk) dan cdrom (cdrom). Selanjutnya, pengguna bisa memberikan lokasi ISO image CDROM pada opsi path. Opsi status akan menunjukkan apakah CDROM dimasukkan (*inserted*) atau tidak (*ejected*).

- **Boot sequence**

Opsi yang dipergunakan adalah boot. Contoh:

```
boot: cdrom, c
```

- **Status mouse (digunakan atau tidak, serta protokol mouse apabila digunakan)**

Opsi yang dipergunakan adalah mouse. Nilai yang mungkin adalah opsi `enabled=0` atau `enabled=1`. Protokol dapat disebutkan secara eksplisit dengan opsi type. Contoh:

```
mouse: enabled=0
```

Berikut ini adalah file konfigurasi (de-

ngan nama `bochsrc`) yang menggunakan semua opsi yang telah dibahas sebelumnya:

```
$ cat bochsrc
megs: 64
romimage: file=/usr/share/
bochs/BIOS-bochs-latest,
address=0xf0000
vgaromimage: file=/usr/share/
bochs/VGABIOS-lgpl-latest
vga: extension=vbe
floppya: 1_44=/dev/fd0,
status=ejected
ata0-master: type=disk,
path="debian31.img",
mode=flat, cylinders=304,
heads=16, spt=63
ata0-slave: type=cdrom,
path="netinst.iso",
status=inserted
boot: cdrom, c
mouse: enabled=0
```

Langkah 3: Selesai

Sampai di sini, seharusnya kita memiliki tiga file di dalam direktori aktif:

- `bochsrc`, file konfigurasi.
- `Debian31.img`, yang merupakan hard-disk primary master.
- `Netinst.iso`, yang merupakan image CDROM. Image ini dibuat dengan sumber adalah CDROM Net Install yang bisa didownload di Debian.org.

Sampai di sini, kita telah siap untuk menjalankan bochs. Dengan boot sequence adalah `cdrom, c`, maka pertama-tama CDROM installer akan di-boot. Berikan perintah berikut ini untuk menjalankan Bochs:

```
$ bochs -q
```

Sebuah dialog akan ditampilkan. Setelah itu, instalasi dapat dilakukan seperti biasa. Setelah instalasi selesai, maka Debian 3.1 yang diemulasikan pun siap untuk digunakan.

Sampai di sini dulu pembahasan kita mengenai Bochs, emulator open source yang sungguh-sungguh luar biasa. Selamat mencoba dan sukses! 🙌

Noprianto (noprianto@infolinux.co.id)

Be-I NET

LINUX HOSTING	
EKONOMI 5 MB	Rp. 24.000,-/TAHUN
BISNIS 25 MB	Rp. 30.000,-/3 BULAN
50 MB	Rp. 54.000,-/3 BULAN
100 MB	Rp. 90.000,-/3 BULAN
PRO 250 MB	Rp. 60.000,-/BULAN
500 MB	Rp. 100.000,-/BULAN
1.000 MB	Rp. 150.000,-/BULAN

Intel Pentium 4 2.8 Ghz
Dual Xeon
Multi Backbone:
- OC3 (155mbps)
- DS3 (45mbps)
- OC3 (155mbps)
Unlimited Email
Unlimited MySQL
Web Statistics
FTP, Web Control Panel
Script Instant
PHP, CGI, SSI

DOMAIN(.COM/.NET/.ORG)
Rp. 90.000,-/TAHUN

* Khusus Hosting

www.be-i.net
Be-I NET, Komplek Batan Indah Blok I 1
Tangerang - Banten 15313
Telp. 021.92757548 - HP: 0818161229