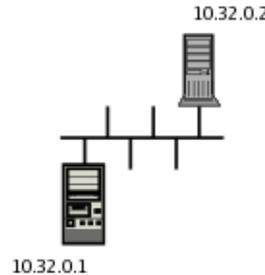




## LF Tip: Mengkloning seluruh PC melalui jaringan



oleh Gerrit Renker

<gerrit.renker(at)gmx.de>

### Tentang Penulis:

Obtained a computer science degree in 2001.

### Diterjemahkan Ke Indonesia

#### Oleh:

Ahmad SS Ramadhana

<ahmad.ramadhana(at)gmail.com>

### Abstrak:

Seringkali muncul masalah saat anda harus melakukan replikasi data dari satu komputer ke komputer lainnya.. Howto berikut akan menjelaskan penggunaan metoda yang aman, mudah dilakukan dan efektif dalam kloning berbasis jaringan.

## Skenario

Ketika riset-riset mengenai kloning binatang (Domba Dolly) dan bahkan embrio manusia banyak diperdebatkan dan diragukan, tidak diragukan lagi bahwa pengetahuan seputar mengkloning mesin komputer tidaklah terlalu berbahaya (bila dilakukan dengan tepat) tetapi juga berguna untuk evolusi penulisan konfigurasi skrip dan setting seseorang. Menurut hukum Moore dan perkembangan yang cepat dalam fabrikasi komputer, seseorang akan menghadapi setidaknya satu skenario kloning selama masa penggunaan distribusi – bisa jadi sebuah PC rumahan yang akan di ganti oleh laptop atau sekedar prosesor yang lebih cepat yang telah tersedia di pasar karena dana untuk membeli komputer baru tidak ada. Gugus tugasnya meliputi *mengambil seluruh filesystem* komputer A dan meletakkannya untuk bekerja pada komputer B. Ada beberapa cara untuk melakukannya, yang pertama adalah yang tidak dijelaskan dalam howto ini yaitu dengan membuka casing, menukar hardisk dan melakukan kopi secara lokal pada komputer satunya. Seringkali hal ini tidaklah mungkin – membuka casing pelindung acapkali menyebabkan hilangnya garansi – dan juga sangat berbahaya, terutama bila dilakukan oleh orang yang tidak berpengalaman sehingga dapat menyebabkan kerusakan hardware secara elektrik maupun fisik. Saya pernah mengalami kehilangan data dengan cara seperti ini, disebabkan oleh program low level yang mengandung bug. Pendekatan lain membutuhkan network card pada kedua PC (dimana saat ini telah banyak di jumpai di PC-PC rumahan), yang lebih aman dan akan dijelaskan di sini.

Ide pokok umum pada semua metoda yang dijelaskan berikut ialah untuk menyelenggarakan koneksi network antara komputer "*sumber*" (yang akan di kloning) dan komputer "*target*" (kloningnya). Hal ni mudah tercipta bila keduanya terhubung ke sebuah hub, bila tidak anda dapat menghubungkan network card-network card tersebut lewat kabel "*crossover*" (karena kabel straight yang normal tidak bisa digunakan). Untuk PC target,

dibutuhkan sebuah live CD (seperti Knoppix atau LNX-BBC) atau instalasi minimal yang dapat mengoperasikan network card dan/atau `ssh/netcat`. Ada pula beberapa floppy yang dapat melakukan hal ini (meskipun dalam kasus saya `tomsrbt` hang saat inisialisasi network card). Bagaimanapun juga bila anda bermaksud untuk melakukan instalasi fresh sebuah distro, ini adalah alternatif lain mudah. Kedua komputer harus di konfigurasi dengan alamat IP pada network yang sama sehingga mereka dapat berbicara satu dengan yang lain, seperti gambar diatas.

## Metoda yang memungkinkan

Dengan setup dasar yang diberikan, ada beberapa cara untuk menyelenggarakan kegiatan kloning:

- pengkopian binary lewat `dd`
- piping `tar / cpio`
- `rsync`
- `dump` dan `restore`

Yang pertama sangatlah komplek dan tidak memungkinkan bila hard drive-hard drive tidak benar-benar berada dalam tipe dan geometri yang sama. Sangatlah luar biasa sekali mengkopi iso images (`dd if=/dev/cdrom of=the.iso`), atau floppy – [disini \(diskcopy shell script\)](#) sebagai contoh `diskcopy` dengan `dd`. Kerugian lainnya dari pendekatan berbasis `dd` ialah ikut terkopinya space kosong, sehingga mengambil waktu yg dibutuhkan. Piping `tar` dan `cpio` akan sangat memakan waktu (sekitar beberapa jam) dan memiliki beberapa kendala. Sebagai contoh, ada pembatasan pada filename dan penggunaa `symlink`, pencekikan file-file di `/dev` dan lain sebagainya. Saya pribadi tak akan merekomendasikan hal ini. Bila anda memiliki filesystem yang berbeda pada source dan target, **`rsync (1)`** mungkin adalah pilihan terbaik. Hanya membutuhkan `ssh` yang sedang bekerja dan transmit file yang efektif dikarenakan builtin protokol yang efisien. Ia juga memiliki swicth `-D` untuk file-file divais, dan banyak pilihan-pilihan yang dapat dipenuhi dalam beragam skenario. Perangkat yang sangat berguna untuk backup harian, mirroring dan sejenisnya, man page yang memiliki banyak contoh untuk dipelajari dengan baik. Contoh dari kloning lewat `rsync` dijelaskan pada [1].

Disini, kita akan menggunakan metoda melalui `dump` dan `restore` yang berarti menciptakan kembali segenap filesystem. Metoda ini cepat, efektif dan memberikan hasil yang diharapkan, dengan upaya seminimal mungkin. Sebenarnya saya harus melakukan seluruh prosedur kloning dua kali, karena target PC akan diganti. Dalam kedua kasus, mutlak tidak ada kendala yang muncul dan memberikan kloning yang berfungsi, bootable dalam waktu kurun waktu sekitar 1 jam mengkopi bergiga-gigabytes. Pendakatan ini membutuhkan baik PC target maupun sumber ciri-ciri filesystem yang sama. Kita berasumsi bahwa ini adalah `ext2` or `ext3`, karena saat ini keduanya adalah filesystem yang paling luas digunakan (lihat catatan di [bawah](#)).

## Setting up ssh

Setelah sistem terkonfigurasi dengan instalasi minimal atau running dengan live CD, langkah berikutnya ialah mengeseet `ssh` (bila tidak menggunakan `netcat` untuk transfer seperti yang dijelaskan lebih jauh di bawah ini). Ini membutuhkan PC sumber memiliki `sshd` (secure shell daemon) yang telah bekerja. Cek `/etc/init.d/` bila tidak yakin. Pada PC target, ketik (sebagai root)

```
ssh-keygen -t rsa
```

Untuk lebih mudahnya, jangan masukkan passphrase. Public key yang kini diperoleh berada di `/root/.ssh/id_rsa.pub`. Kopi file ini ke PC sumber anda lewat :

```
scp /root/.ssh/id_rsa SourcePC:/tmp
```

dimana `SourcePC` adalah alamat IP dari PC sumber. Saat ditanya apakah anda yakin, masukkan "yes" secara lengkap ("y" saja sendirian seringkali tidak bekerja). Anda masih akan diberi prompt password root pada PC sumber. Sekarang tambahkan PC target sebagai node yang terpercayai dengan memberikan :

```
cat /tmp/id_rsa.pub >> /root/.ssh/authorized_keys
```

pada PC sumber. Untuk mengecek apakah segalanya OK, ulangi perintah kopi pada PC target. Anda seharusnya tak akan di beri prompt untuk password.

## Membuat file system pada PC target

Langkah pertama dapat dipastikan adalah mempartisi hard drive pada sistem target dan menciptakan filesystem `ext2/ext3`. Yang terakhir adalah varian journalling yang lebih baik dan di enable secara sederhana dengan mengeset pilihan `-j` (journalling) pada `mke2fs` (membutuhkan dukungan `ext3` di kernel). Anda pun dapat mengkonversikan sistem `ext2` ke `ext3`, lihat **tune2fs (8)**. Jadi, bila PC sumber memiliki konfigurasi sebagai berikut :

Filesystem	Ukuran	Penggunaan	Penggunaan%	Di Mount pada
/dev/hda3	2.7G	552M	22%	/
/dev/hda5	7.8G	1.6G	22%	/usr
/dev/hda7	6.3G	1.7G	28%	/usr/share
/dev/hda8	3.4G	601M	19%	/home
/dev/hda12	5.3G	1.9G	37%	/opt
/dev/hda1	587M	70M	13%	/var/backup

Saya rekomendasikan untuk selalu melakukan proses partisi. Bila tidak, maka kesalahan apapun dalam penggunaan filesystem atau hard-drive knock-out walaupun beberapa sektor saja dapat menghancurkan seluruh data anda. Dan menurut hukum Murphy, hal ini pasti terjadi bila tidak ada tindakan pencegahan yang diambil dengan penggunaan partisi yang berbeda daripada sebuah monolithic. Saya memiliki sebuah kasus baru-baru ini dengan sebuah kernel yang lucu dan belum mempartisi drive, selama penguyahan filesystem root. Diatas menunjukkan bahwa `/usr` telah terisi begitu besar, oleh karena itu `/usr/share` harus di tambahkan. Waktunya mengganti dengan hardisk yang lebih besar.

Pada PC target, jalankan `parted` (direkomendasikan) atau program partisi favorit (Qtptered adalah variant GUI yang baik, nyaris bisa dibilang sebagai kloning dari Partition Magic). Jangan lupa partisi swap. Setelah menyimpan table partisi, masukkan filesystem pada seluruh partisi yang baru dibuat, dengan :

```
mke2fs -j -L <label> /dev/xxx
```

dimana `xxx` adalah nama partisi dan `<label>` adalah sebuah string label. Biasanya saya menggunakan `"/usr"` sebagai label (akan anda lihat saat boot). Anda dapat mengeset bermacam hal melalui **tune2fs (8)**, seperti rentang waktu tetap untuk pengecekan sistem internal .

# Mentransfer filesystem

Pertama, seluruh partisi yang baru tersebut harus di mount. Kita mulai dari filesystem root ("/") dan mount direktori yang tersisa selanjutnya. Sangat mungkin untuk menyingkat dua partisi dari PC sumber ke dalam sebuah target singel, ini lah yang akan dilakukan pada /usr/ dan /usr/share dalam contoh diatas. Jadi mount filesystem root "masa depan" dengan menggunakan :

```
mount /dev/xxx /mnt
```

Saat kloning, penting untuk melakukan chdir ke target direktori

```
cd /mnt
```

Kemudian bit network, pada target PC ketik

```
ssh targetPC 'dump -0 -f - /' | restore -r -f -
```

dimana targetPC adalah alamat IP PC target . Option yang digunakan, "-0" berarti full backup, "-f -" memerintahkan penggunaan stdin/stdout sebagai file descriptors dan "-r" memerintahkan untuk program restore untuk menciptakan ulang filesystem yang sedang di piping lewat network pada PC target . Untuk option lebih lanjut, baca **dump (8)** dan **restore (8)** . Berikut adalah output untuk mentransfer filesystem root :

```
$ ssh 10.42.3.42 'dump -0 -f - /' | restore -r -f -
DUMP: Date of this level 0 dump: Tue Feb 22 15:50:12 2005
DUMP: Dumping /dev/hda3 (/) to standard output
DUMP: Label: debian
DUMP: Writing 10 Kilobyte records
DUMP: mapping (Pass I) [regular files]
DUMP: mapping (Pass II) [directories]
DUMP: estimated 547312 blocks.
DUMP: Volume 1 started with block 1 at: Tue Feb 22 15:50:14 2005
DUMP: dumping (Pass III) [directories]
DUMP: dumping (Pass IV) [regular files]
DUMP: Volume 1 completed at: Tue Feb 22 15:51:43 2005
DUMP: Volume 1 546590 blocks (533.78MB)
DUMP: Volume 1 took 0:01:29
DUMP: Volume 1 transfer rate: 6141 kB/s
DUMP: 546590 blocks (533.78MB)
DUMP: finished in 89 seconds, throughput 6141 kBytes/sec
DUMP: Date of this level 0 dump: Tue Feb 22 15:50:12 2005
DUMP: Date this dump completed: Tue Feb 22 15:51:43 2005
DUMP: Average transfer rate: 6141 kB/s
DUMP: DUMP IS DONE
```

Restore selalu membuat sebuah file `restoresymtable` yang dapat dihapus setelah anda yakin tak ada error yang muncul selama restorasi filesystem. Saat selesai dengan filesystem root, sekarang kita akan memproses masing-masing sub-filesystem yang di mount, dimulai dengan /usr (asumsi, direktori tempat anda bekerja saat ini adalah root dari filesystem "masa depan").

```
mount /dev/xxx ./usr
```

```
cd ./usr
```

```
ssh targetPC 'dump -0 -f - /usr' | restore -r -f -
```

Siklus mount–cd–dump/restore sekarang berulang lagi untuk semua direktori yang mungkin anda miliki. Dengan menganggap /usr/share (dimana PC sumber memiliki partisinya), setelah langkah diatas, anda dapat dengan mudah chdir ke ./usr/share (perhatikan ".") dan mengulangi

```
ssh targetPC 'dump -0 -f - /usr/share' | restore -r -f -
```

Restore hanya akan mengkomplain bila file telah ada pada filesystem yang tengah direstore, sehingga tidak masalah saat meletakkan dua partisi dari PC source ke dalam satu. Mengkloning seluruh PC akan menghabiskan waktu sekitar satu jam dengan ssh dan network card 100 MB (crossover cable)

**Catatan:** Untuk dump sebuah filesystem, tidak perlu di mount. Anda dapat pula mem–pass sebuah *nama partition*, seperti /dev/hda6, dibanding nama direktori dari partisi yang di mount.

## Alternatif: netcat

Sebuah alternatif lain dari ssh adalah penggunaan netcat (1), yang disingkat sebagai nc. Netcat adalah client–server sederhana yang siap digunakan yang memungkinkan pembuatan sebuah "pipa" melewati network. Contoh diatas dapat dimodifikasi sebagai berikut. Kami berasumsi bahwa partisi yang di mount dalam /var/backup akan ditransfer dengan dump/restore dari PC sumber ke PC target.

Pada sisi penerima (*PC target*) membuat sebuah instance yang mendengarkan dari netcat dengan -l dimana akan memiping outputnya ke program restore.

```
nc -l -p 2000 -q 1 | restore -r -f -
```

Pada PC sumber, buat sebuah instance dari netcat yang akan mengambil input dari sebuah pipa dimana target-IP adalah alamat IP PC target.

```
dump -0 -f - /var/backup | nc <target-ip> 2000
```

Option -q option digunakan untuk menghentikan nc setelah menerima end–of–file, namun dalam kasus saya harus menghentikannya secara manual. Bagaimanapun juga lebih direkomendasikan menggunakan ssh.

## Pembersihan

Selamat, bila sejauh ini sukses, pada dasarnya anda memiliki sebuah sistem ter–kloning. Sekarang tiba saatnya mengubah klon ini menjadi sebuah PC yang bisa bekerja. Hal yang pertama perlu dilakukan adalah mengupdate /etc/fstab dengan setting baru, bila tidak anda tak akan bisa menggunakannya. Bila alamat IP berubah, ini juga perlu diupdate (/etc/hosts, /etc/network/interfaces under debian). Hal penting berikutnya ialah *konfigurasi boot* yang dalam banyak kasus perlu diupdate. Dengan lilo, edit /etc/lilo.conf (khususnya option root=...) dan kemudian jalankan lilo -v. Dengan grub, edit /boot/grub/menu.lst (atau /boot/grub/grub.conf tergantung mana yang symlink) dan kemudian masuk ke dalam grub,

```
grub> root (hd0,xxx)
... filesystem is ...
grub> setup (hd0)
... lots of output here
```

```
grub> quit
```

atau jalankan `grub-install /dev/xxx` dimana `xxx` adalah hard drive anda. Disini, cek pada `root` (`hdn, xx`) dan tambahkan setting `root=/dev/xxx`.

Dalam kasus sejenis dengan hardware yang lebih baik pada PC yang baru di clone, anda mungkin perlu mengupdate *settings untuk kernel custom anda*. Bila anda menggunakan sistem dengan banyak modul yang sudah pre-configured (seperti RedHat, SuSe, Mandrake, Fedora ...) jangan takut, sepertinya akan ada modul yang cocok. Bila tidak, `lspci -vv` dan kompil kernel seperti biasanya. Bila video card anda berbeda, update `/etc/X11/XF86Config-4` (or `xorg.conf` pada RH/Fedora) untuk menyesuainya, atau anda tidak akan memperoleh output. Bila mungkin, gunakan perangkat grafis untuk mensetting X dengan membooting pada runlevel 3 bila anda memiliki toolsnya. Pada debian, beberapa penyelidikan sangat diperlukan, saya cukup beruntung menemukan perubahan driver dari `r128` ke `radeon` dan itu bekerja.

## Sistem lain

Howto ini menjelaskan prosedur kloning untuk filesystem `ext2/ext3`. Banyak perintah sejenis yang dapat ditemukan pada banyak sistem Linux lain. Sebagai contoh, beberapa jenis unix seperti FreeBSD, HP-UX, IRIX juga menyediakan perintah `dump/restore`; pada Solaris ada `ufsdump/ufsrestore`. Ada beberapa filesystem yang tidak menawarkan fungsi dump filesystem, seperti. ReiserFS. Disini, gunakan `rsync`. Lihat [1] untuk laporan kesuksesan penggunaan `rsync` untuk mengkloning sebuah sistem Linux.

## Referensi

[1] "*Replicating a Linux System – Yet Another Method.*" Ben Okopnik, Linux Gazette Issue 83, October 2002.

---

<p><u>Halaman Web Dirawat oleh Team Editor</u> <u>LinuxFocus</u> © Gerrit Renker "some rights reserved" see <a href="http://linuxfocus.org/license/">linuxfocus.org/license/</a> <a href="http://www.LinuxFocus.org">http://www.LinuxFocus.org</a></p>	<p>Informasi Terjemahan: en --&gt; -- : Gerrit Renker &lt;gerrit.renker(at)gmx.de&gt; en --&gt; id: Ahmad SS Ramadhana &lt;ahmad.ramadhana(at)gmail.com&gt;</p>
--	---

2005-06-30, generated by lfparsr\_pdf version 2.51