The Linux Tips HOWTO

Paul Anderson, paul@geeky1.ebtech.net diterjemahkan oleh Ronny Haryanto, giant@technologist.com
v3.0, 7 January 1997, diterjemahkan 23 November 1997

HOWTO ini berisi petunjuk-petunjuk dan tips-tips yang membuat Linux sedikit lebih baik.

Contents

1 Pendahuluan

Selamat datang di **Linux Tips HOWTO**, sebuah daftar dari trik-trik dan optimasi-optimasi yang membuat Linux menjadi lebih menyenangkan. Semua yang ada di sini sekarang adalah dari pengetahuan penulis, dan juga dari Tips-HOWTO yang lama (trik yang bagus tidak harus dihilangkan, bukan?). Jadi kirimkan semua petunjuk dan trik favorit Anda kepada penulis sehingga penulis bisa mengikutsertakannya di Tips-HOWTO yang berikut.

Paul Anderson *Maintainer–Linux TIPS HOWTO*

`panderso@ebtech.net`

2 Petunjuk Singkat

2.1 Trik Syslog yang berguna *Paul Anderson, Tips-HOWTO maintainer*

Suntinglah /etc/syslog.conf, dan letakkan baris berikut:

```
# Dump everything on tty8
*.*                                     /dev/tty8
```

Harap diingat: *JANGAN LUPA UNTUK MENGGUNAKAN TAB!* syslog tidak menyukai spasi...

2.2 Script untuk melihat isi HOWTO yang dipadatkan (compressed). *Didier Juges,* `dj@destin.nfds.net`

Dari pendatang baru dan seterusnya, ini adalah script singkat yang memudahkan mencari dan melihat dokumen-dokumen HOWTO. Berkas-berkas HOWTO terletak di /usr/doc/faq/howto/ dan dipadatkan dengan gzip. Nama-nama berkas ialah XXX-HOWTO.gz, di mana XXX adalah subjek dari HOWTO. Penulis membuat script berikut yang dinamakan "howto" di direktori /usr/local/sbin

```
#!/bin/sh
if [ "$1" = "" ]; then
```

```
    ls /usr/doc/faq/howto | less
else
    gunzip -c /usr/doc/faq/howto/$1-HOWTO.gz | less
fi
```

Ketika dijalankan tanpa menyertakan argumen, script ini akan menampilkan sebuah direktori berisi HOWTO yang ada. Kemudian ketika disertakan subjek dari HOWTO (bagian depan dari nama berkas sebelum tanda hubung) sebagai argumen, maka akan meng-unzip (tanpa mengubah berkas asli) dan menampilkan isi dokumen tersebut.

Sebagai contoh, untuk melihat dokumen Serial-HOWTO.gz, masukkan:

$ howto Serial

2.3 Apakah ada cukup tempat di hard disk??? *Hans Zoebelein,* `zocki@goldfish.cube.net`

Berikut ini adalah sebuah script singkat yang akan mengecek secara berkala apakah ada cukup tempat kosong yang tersedia di semua yang ditampilkan oleh mount (disks, cdrom, floppy...)

Jika kehabisan tempat kosong, sebuah pesan akan ditampilkan di layar setiap X detik dan juga akan dikirim 1 e-mail untuk masing-masing device yang penuh.

```
#!/bin/sh

#
# $Id: check_hdspace,v 1.18 1996/12/11 22:33:29 root Exp root $
#

#
# Since I got mysterious error messages during compile when
# tmp files filled up my disks, I wrote this to get a warning
# before disks are full.
#
# If this stuff saved your servers from exploding,
# send praising email to zocki@goldfish.cube.net.
# If your site burns down because of this, sorry but I
# warned you: no comps.
# If you really know how to handle sed, please forgive me :)
#

#
# Shoot and forget: Put 'check_hdspace &' in rc.local.
# Checks for free space on devices every $SLEEPTIME sec.
# You even might check your floppies or tape drives. :)
# If free space is below $MINFREE (kb), it will echo a warning
# and send one mail for each triggering device to $MAIL_TO_ME.
# If there is more free space than trigger limit again,
# mail action is also armed again.
#

# TODO: Different $MINFREE for each device.
```

```
# Free /*tmp dirs securely from old junk stuff if no more free space.


DEVICES='/dev/sda2 /dev/sda8 /dev/sda9'         # device; your put disks here
MINFREE=20480                                   # kb; below this do warning
SLEEPTIME=10                                    # sec; sleep between checks
MAIL_TO_ME='root@localhost'                     # fool; to whom mail warning


# ------- no changes needed below this line (hopefully :) -------

MINMB=0
ISFREE=0
MAILED=""
let MINMB=$MINFREE/1024         # yep, we are strict :)

while [ 1 ]; do
        DF="`/bin/df`"
                for DEVICE in $DEVICES ; do
                ISFREE=`echo $DF | sed s#.\*$DEVICE" "\*[0-9]\*""\*[0-9]\*" "\*## | sed s#" ".\*##`

                if [ $ISFREE -le $MINFREE ] ; then
                        let ISMB=$ISFREE/1024
                        echo  "WARNING: $DEVICE only $ISMB mb free." >&2
                        #echo "more stuff here" >&2
                        echo -e "\a\a\a\a"

                        if [ -z  "`echo $MAILED | grep -w $DEVICE`" ] ; then
                                echo "WARNING: $DEVICE only $ISMB mb free.      (Trigger is set to $MINMB mb)"
                                | mail -s "WARNING: $DEVICE only $ISMB mb free!" $MAIL_TO_ME
                                MAILEDH="$MAILED $DEVICE"
                                MAILED=$MAILEDH
                                # put further action here like cleaning
                                # up */tmp dirs...
                        fi
                        elif [ -n  "`echo $MAILED | grep -w $DEVICE`" ] ; then
                                # Remove mailed marker if enough disk space
                                # again. So we are ready for new mailing action.
                                MAILEDH="`echo $MAILED  | sed s#$DEVICE##`"
                                MAILED=$MAILEDH
                        fi

                done
                sleep $SLEEPTIME

done
```

2.4 Utilitas untuk membersihkan berkas-berkas log. *Paul Anderson, Tips-HOWTO Maintainer>*

Jika Anda seperti penulis, Anda punya list dengan 250 pelanggan, ditambah lagi 100+ e-mail per hari datang lewat UUCP. Jadi, apa yang akan dilakukan oleh hacker dengan log-log yang besar ini? Pasang chklogs, itulah yang seharusnya dilakukan. Chklogs ditulis oleh Emilio Grimaldo, `grimaldo@panama.iaehv.nl`, dan versi terbaru ialah 1.8 yang bisa didapat dari `<ftp.iaehv.nl:/pub/users/grimaldo/chklogs-1.8.tar.gz>`. Tidak perlu banyak penjelasan untuk memasang chklogs (tentu saja Anda akan membaca info di subdirektori doc). Jika chklogs sudah terpasang dengan benar, tambahkan yang berikut ini di crontab:

```
# Run chklogs at 9:00PM daily.
00 21 * * *       /usr/local/sbin/chklogs -m
```

Sambil lalu, sampaikan kepada penulis chklogs betapa bagusnya program ini :)

2.5 Script untuk membersihkan berkas-berkas core. *Otto Hammersmith,* `ohammers@cu-online.com`

Buat berkas bernama rmcores (penulisnya menyebutnya handle-cores) yang berisi:

```
#!/bin/sh
USAGE="$0 <directory> <message-file>"

if [ $# != 2 ] ; then
        echo $USAGE
        exit
fi

 echo Deleting...
find $1 -name core -atime 7 -print -exec rm {} \;

echo e-mailing
for name in `find $1 -name core -exec ls -l {} \; | cut -c16-24`
do
        echo $name
        cat $2 | mail $name
done
```

Dan buatlah sebuah cron job yang menjalankannya setiap beberapa saat tertentu.

2.6 Memindahkan direktori-direktori antar filesystem. *Alan Cox,* `A.Cox@swansea.ac.uk`

Sebuah cara cepat untuk memindahkan seluruh direktori beserta isinya dari satu disk ke disk yang lain:

```
(cd /source/directory && tar cf - . ) | (cd /dest/directory && tar xvfp -)
```

[Gantilah dari cd /source/directory; tar....dst. untuk menghindari kemungkinan terjadinya penghapusan direktori secara tidak sengaja atau terjadinya bencana. Terima kasih kepada Jim Dennis, jadestar@rahul.net, yang memberitahu penulis.]

2.7 Mengetahui direktori yang berukuran terbesar. *Mick Ghazey,* `mghazey@miso.lowdown.com`

Pernahkan terpikir untuk mengetahui direktori mana yang terbesar ukurannya? Berikut ini adalah sebuah caranya.

```
du -S | sort -n
```

2.8 The Linux Gazette

Pujian untuk John Fisk, pembuat The Linux Gazette. Ini adalah sebuah majalah elektornik (e-zine) plus, dan **GRATIS!!!** Nah, apa lagi yang bisa anda harapkan? Silakan dilihat di:

```
http://www.ssc.com/lg
```

Untuk informasi Anda, (1) LG sekarang terbit bulanan, dan (2) LG bukan lagi diurus oleh John Fisk, melainkan orang-orang dari SSC.

2.9 Petunjuk untuk memperbaiki GNU Make 3.70: mengubah cara VPATH berlaku. *Ted Stern,* `stern@amath.washington.edu`

Penulis tidak tahu apakah banyak orang yang mengalami hal yang sama, tetapi ada "feature" dari GNU Make version 3.70 yang tidak disukai penulis, ialah VPATH berlaku aneh jika diberikan pathname yang absolut. Ada patch yang sangat baik yang bisa memperbaiki hal ini, Anda bisa mendapatkannya dari Paul D. Smith <`psmith@wellfleet.com`>. Beliau juga mengirimkan dokumentasi dan patch setiap setelah revisi dari GNU Make di newsgroup 'gnu.utils.bug'. Pada umumnya, penulis menggunakan patch ini dan meng-compile ulang gmake di setiap sistem di mana penulis mempunyai akses.

2.10 Bagaimana cara memberhentikan fsck pada setiap kali reboot? *Dale Lutz,* `dal@wimsey.com`

Q: Bagaimana cara memberhentikan e2fsck mengecek disk saya setiap kali saya boot?

A: Ketika anda membangun ulang kernel, filesystem diberi tanda 'kotor' dan disk Anda akan dicek pada setiap boot. Untuk memperbaikinya, jalankan:

rdev -R /zImage 1

Ini akan memperbaiki kernel supaya kernel percaya bahwa filesystem tidak 'kotor' lagi.

Catatan: jika menggunakan lilo, maka tambahkan `read-only` *pada linux setup di berkas konfigurasi lilo Anda (biasanya /etc/lilo.conf)*

2.11 Bagaimana menghindari fsck yang disebabkan oleh "device busy" pada saat reboot. *Jon Tombs,* `jon@gtex02.us.es`

Jika Anda sering mengalami kesalahan "device busy" pada saat shutdown yang menyebabkan filesystem membutuhkan fsck pada saat reboot, berikut ini ada cara mudah untuk memperbaikinya:

Tambahkan baris berikut pada /etc/rc.d/init.d/halt atau /etc/rc.d/rc.0

```
mount -o remount,ro /mount.dir
```

untuk semua filesystem yang di-mount kecuali /, sebelum menjalankan umount -a. Ini berarti, jika untuk suatu alasan, shutdown tidak bisa membunuh semua proses dan umount semua disk, semuanya akan masih bersih sewaktu reboot. Bagi penulis hal ini dapat menghemat waktu banyak.

2.12 Cara untuk menemukan berkas berukuran terbesar di hard-drive Anda.

Simon Amor, simon@foobar.co.uk

```
ls -l | sort +4n
```

Atau, bagi Anda sekalian yang kekurangan tempat kosong, ini agak memakan waktu tetapi bisa berjalan dengan baik:

```
cd /
ls -lR | sort +4n
```

2.13 Cara mencetak supaya kertas bisa dilubangi di bagian tepi. *Mike Dickey,* mdickey@thorplus.lib.purdue.edu

```
#!/bin/sh
# /usr/local/bin/print
# a simple formatted printout, to enable someone to
# 3-hole punch the output and put it in a binder

cat $1 | pr -t -o 5 -w 85 | lpr
```

2.14 Sebuah cara untuk mencari dengan regular expression di antara berkas-berkas. *Raul Deluth Miller,* rockwell@nova.umd.edu

Penulis memberi nama script ini 'forall'. Cara penggunaannya sebagai berikut:

```
forall /usr/include grep -i ioctl
forall /usr/man grep ioctl
```

Berikut ini adalah script forall:

```
#!/bin/sh
if [ 1 = `expr 2 \> $#` ]
then
        echo Usage: $0 dir cmd [optargs]
        exit 1
fi
dir=$1
shift
find $dir -type f -print | xargs "$@"
```

2.15 Sebuah script untuk membersihkan berkas autosave dan backup yang dibuat oleh program.*Barry Tolnas,* `tolnas@nestor.engr.utk.edu`

Berikut adalah script dua baris yang secara rekursif menjelajahi hirarki dari sebuah direktori dan menghapus berkas-berkas autosave (#) dan backup (~) dari emacs, berkas-berkas berakhiran .o, dan berkas-berkas berakhiran .log dari TeX. Script ini juga memadatkan (compress) berkas-berkas berakhiran .tex dan berkas-berkas README. Penulis memberi nama 'squeeze' untuk script ini pada sistemnya.

```
#!/bin/sh
#SQUEEZE removes unnecessary files and compresses .tex and README files
#By Barry tolnas, tolnas@sun1.engr.utk.edu
#
echo squeezing $PWD
find  $PWD \( -name \*~ -or -name \*.o -or -name \*.log -or -name \*\#\) -exec
rm -f {} \;
find $PWD \( -name \*.tex -or -name \*README\* -or -name \*readme\* \) -exec gzip -9 {} \;
```

2.16 Cara mengetahui proses yang paling banyak memakan memori. *Simon Amor,* `simon@foobar.co.uk`

```
ps -aux | sort +4n
```

-ATAU-

```
ps -aux | sort +5n
```

3 Petunjuk-petunjuk lebih mendetil.

3.1 Menggunakan partisi swap bersama antara Linux dan Windows. *Tony Acero,* `ace3@midway.uchicago.edu`

1. Formatlah sebuah partisi sebagai partisi DOS, dan buatlah swap untuk Windows pada partisi tersebut, tetapi jangan menjalankan Windows dahulu. (Kita menginginkan supaya swap benar-benar kosong untuk saat ini, sehingga dapat dipadatkan (di-compress) dengan baik).

2. Boot linux dan simpanlah partisi tersebut ke sebuah berkas (file). Sebagai contoh, jika partisinya adalah /dev/hda8:

   ```
   dd if=/dev/hda8 of=/etc/dosswap
   ```

3. Padatkan berkas dosswap; karena semua isinya adalah 0 maka akan dipadatkan dengan sangat baik

   ```
   gzip -9 /etc/dosswap
   ```

4. Tambahkan yang berikut ini pada berkas /etc/rc untuk mempersiapkan dan memasang tempat swap di Linux:

   XXXXX adalah jumlah block di partisi swap

   ```
   mkswap /dev/hda8 XXXXX
   swapon -av
   ```

Jangan lupa menambahkan masukan untuk partisi swap di berkas /etc/fstab Anda.

5. Jika paket init/reboot Anda mendukung /etc/brc atau /sbin/brc tambahkan yang berikut ini pada /etc/brc. Jika tidak, jalankan ini secara manual ketika Anda menge-boot ke DOS atau OS/2 dan Anda ingin mengubah partisi swap kembali ke versi dos/windows:

```
swapoff -av
zcat /etc/dosswap.gz | dd of=/dev/hda8 bs=1k count=100
```

Harap diingat bahwa ini hanya akan menulis 100 block pertama kembali ke partisi. Dari pengalaman, penulis berpendapat bahwa ini sudah cukup.

>> Apa saja pro dan kontra dari hal ini?

Pro: Anda dapat menghemat sejumlah tempat di disk.

Cons: Jika langkah ke-5 tidak berlangsung otomatis, Anda harus ingat untuk melakukannya secara manual, dan ini akan memperlambat proses boot sebanyak beberapa nano-detik :-)

3.2 Undelete dalam keadaan putus asa. *Michael Hamilton,* `michael@actrix.gen.nz`

Ini adalah trik yang sudah penulis gunakan beberapa kali.

Undelete berkas teks untuk orang yang sudah putus asa.

Jika Anda secara tidak sengaja menghapus sebuah berkas teks, misalnya email, atau hasil pemrograman di larut malam, masih ada kemungkinan bahwa semuanya tidak hilang. Jika berkas pernah disimpan di disk selama lebih dari 30 detik, isinya mungkin masih ada di partisi.

Anda bisa menggunakan utilitas grep untuk mencari isi berkas di partisi disk secara langsung (raw).

Sebagai contoh, baru-baru ini penulis menghapus sebuah email secara tidak sengaja. Jadi penulis berhenti dari semua aktivitas yang bisa mengakibatkan berubahnya partisi: dalam hal ini penulis hanya menunda penyimpanan berkas atau menunda melakukan compile, dsb. Pada kejadian lain, penulis pernah terbawa kesulitan hingga harus berpindah ke mode single user, dan meng-unmount filesystem.

Kemudian penulis menggunakan perintah egrep terhadap partisi disk tersebut: dalam hal ini, emailnya terletak di /usr/local/home/michael/, jadi dari keluaran df, penulis bisa melihat bahwa ini sebenarnya di /dev/hdb5

```
sputnik3:~ % df
  Filesystem         1024-blocks  Used Available Capacity Mounted on
  /dev/hda3              18621    9759      7901     55%   /
  /dev/hdb3             308852  258443     34458     88%   /usr
  /dev/hdb5             466896  407062     35720     92%   /usr/local

  sputnik3:~ % su
  Password:
  [michael@sputnik3 michael]# egrep -50 'ftp.+COL' /dev/hdb5 > /tmp/x
```

Sekarang, penulis sangat berhati-hati ketika bermain-main dengan partisi disk, sehingga penulis berhenti sejenak supaya yakin bahwa penulis mengerti arti sintaksis dari perintah yang akan dimasukkan SEBELUM menekan return/enter. Dalam hal ini email tersebut mengandung kata 'ftp' yang diikuti dengan teks tertentu yang diikuti dengan kata 'COL'. Email tersebut kira-kira panjangnya 20 baris, jadi penulis menggunakan -50 untuk mencakup frase tersebut. Di masa lampau penulis menggunakan -3000 untuk mencakup sebuah frase dari sebuah kode (source code). Penulis mengarahkan keluaran dari egrep ke partisi disk yang lain, hal ini mencegah terjadinya penimpaan atas email yang dicari tersebut.

Penulis kemudian menggunakan strings untuk membantu mengamati keluaran tersebut:

```
strings /tmp/x | less
```

Dan ternyata email tersebut ada di sana.

Metoda ini tidak bisa diandalkan sepenuhnya, karena ada kemungkinan semua atau sebagian dari disk space tersebut telah digunakan kembali.

Trik ini mungkin hanya berguna untuk sistem single user. Pada sistem multi-user dengan aktivitas disk yang sangat tinggi, space yang dibebaskan (karena proses penghapusan) mungkin telah digunakan kembali untuk yang lain. Dan kita tidak bisa seenaknya mengambil alih komputer dari pengguna-pengguna yang lain ketika kita butuh untuk mengambil kembali berkas yang terhapus.

Di sistem di rumah penulis trik ini sudah digunakan kira-kira tiga kali dalam beberapa tahun terakhir - biasanya ketika penulis menghapus hasil kerja hari tersebut. Jika apa yang penulis kerjakan sampai pada suatu titik di mana penulis merasa bahwa sudah ada cukup kemajuan, pekerjaan itu akan di-backup ke floppy, jadi penulis jarang menggunakan trik ini.

3.3 Cara menggunakan flag yang tidak berubah (immutable flag). *Jim Dennis,* `jadestar@rahul.net`

Penggunaan 'Immutable Flag'

Setelah Anda memasang dan mengkonfigurasi sistem Anda, jelajahilah /bin, /sbin/, /usr/bin, /usr/sbin dan /usr/lib (dan juga beberapa lagi yang bisa dicurigai) kemudian pergunakan 'chattr +i command' seperlunya. Juga tambahkan ke berkas-berkas kernel di root. Sekarang 'mkdir /etc/.dist/' salinlah semua dari /etc/, termasuk sub-subdirektorinya (Penulis menggunakan 2 langkah, menggunakan /tmp/etcdist.tar untuk menghindari recursion). (Kalau Anda mau, bisa juga membuat /etc/.dist.tar.gz) – dan tandailah sebagai immutable.

Latar belakang dari semua ini ialah untuk membatasi kerusakan yang bisa terjadi ketika login sebagai root. Anda tidak akan menimpa berkas-berkas dengan operator redirection yang salah, dan Anda tidak akan membuat sistem tidak bisa terpakai gara-gara penggunaan sebuah spasi yang salah dalam perintah 'rm -fr' (Masih banyak kemungkinan Anda bisa melakukan kerusakan pada data Anda – tetapi berkas-berkas lib dan bin akan menjadi lebih aman).

Hal ini juga akan membuat sejumlah eksploitasi atas security dan 'denial of service' menjadi tidak mungkin atau lebih sukar (dikarenakan kebanyakan dari eksploitasi tersebut mengandalkan penimpaan berkas melalui tindakan-tindakan dari program-program SUID yang *tidak mendukung perintah shell sembarang*).

Satu-satunya ketidakpraktisan dari hal ini ialah ketika membangun dan melakukan 'make install' terhadap bermacam jenis berkas biner sistem. Di lain pihak, hal ini juga mencegah 'make install' dari penimpaan berkas-berkas. Jika Anda lupa membaca Makefile dan 'chattr -i' terhadap berkas-berkas yang akan ditimpa (dan direktori di mana Anda akan menambahkan berkas-berkas) – program 'make' akan gagal, Anda tinggal menggunakan perintah chattr dan menjalankan ulang. Anda juga bisa menggunakan kesempatan itu untuk memindahkan berkas-berkas bin atau lib yg lama atau apa saja yg lain ke sebuah direktori .old/ atau mengubah nama direktorinya, melakukan 'tar' atau terserah Anda.

3.4 Saran: tempat untuk meletakkan barang-barang baru. *Jim Dennis,* `jadestar@rahul.net`

Semua barang baru dimulai di bawah /usr/local! atau /usr/local/'hostname'

Jika distribusi Anda adalah yang /usr/local -nya kosong, maka Anda tinggal membuat /usr/local/src, /usr/local/bin dst dan menggunakannya. Jika distribusi Anda meletakkan berkas-berkas di /usr/local maka Anda mungkin bisa melakukan 'mkdir /usr/local/'hostname'' dan memberikan group 'wheel' +w (Penulis juga membuatnya SUID dan SGID untuk menjamin bahwa masing-masing anggota dari group 'wheel' hanya bisa mengacak-acak berkas mereka sendiri di sana, dan bahwa semua berkas yang dibuat akan menjadi milik group 'wheel').

Sekarang, biasakan diri Anda sendiri untuk *SELALU! SELALU! SELALU!* meletakkan paket-paket baru di bawah /usr/local/src/.from/$DARI_MANA_SAYA_DAPAT/ (untuk berkas .tar atau apa pun akhirannya) dan compile-lah di bawah /usr/local/src (atau .../$HOSTNAME/src). Yakinkan bahwa itu bisa di-install di bawah hirarki lokal. Jika paket tsb. *benar-benar harus* di-install kembali ke /bin atau /usr/bin atau di tempat lainnya – berikan symbolic link dari hirarki lokal ke masing-masing elemen di tempat lainnya.

Alasan dari hal ini ialah – walaupun berarti menambah pekerjaan – hal ini bisa membantu mengisolir apa yang harus di backup dan di- restore atau di-reinstall dalam rangka re-install secara penuh dari media distribusi (sekarang ini biasanya CD). Dengan menggunakan direktori /usr/local/.from Anda juga menyimpan sebuah log informal mengenai asal sumber – yang akan membantu jika Anda mencari update – dan mungkin masalah ini menjadi kritis jika memonitor mailing list security announcement.

Salah satu dari sistem di rumah penulis dipasang sebelum penulis membuat aturan-aturan ini untuk penulis sendiri. Penulis masih belum tahu semua cara-cara yang berbeda untuk masing-masing sistem yang terpasang. Hal ini dikarenakan penulis sangat sedikit melakukan konfigurasi dengan sistem di rumahnya dan penulis satu-satunya orang yang menggunakannya.

Sistem-sistem yang penulis pasang di tempat kerja (ketika penulis menjadi system administrator di sana) semua dipasang dengan cara ini – dan sudah diadministrasi oleh beberapa kontraktor dan orang-orang MIS lain, dan telah mengalami sejumlah besar upgrade dan pemasangan paket-paket baru. Walaupun demikian, penulis masih tahu dengan tepat, elemen-elemen mana saja yang dipasang *setelah* pemasangan dan konfigurasi pertama kali.

3.5 Mengubah semua nama berkas di direktori menjadi huruf kecil. *Justin Dossey,* `dossey@ou.edu`

Penulis memperhatikan adanya beberapa prosedur yang sulit dan tidak perlu dianjurkan di bagian 2c tips dari Edisi 12. Karena ada lebih dari satu, maka penulis mengirimkannya sbb:

```
#!/bin/sh
        # lowerit
```

```
# convert all file names in the current directory to lower case
# only operates on plain files--does not change the name of directories
# will ask for verification before overwriting an existing file
for x in ‘ls‘
  do
  if [ ! -f $x ]; then
    continue
    fi
  lc=‘echo $x  | tr ’[A-Z]’ ’[a-z]’‘
  if [ $lc != $x ]; then
    mv -i $x $lc
  fi
  done
```

Wow, betapa panjangnya. Penulis tidak akan menggunakan script untuk melakukannya; sebaliknya, penulis akan menggunakan perintah:

```
for i in * ; do [ -f $i ] && mv -i $i ‘echo $i | tr ’[A-Z]’ ’[a-z]’‘;
done;
```

di command line.

Orang yang menyumbangkan script tsb. berkata bahwa dia menulis script tsb. supaya lebih mudah dimengerti (lihat di bawah).

Di petunjuk berikut, mengenai penambahan dan pneghapusan user, Geoff melakukan hal yang benar hingga langkah terakhir. Reboot? Wow, penulis berharap dia tidak melakukan reboot setiap kali menghapus user. Yang harus Anda lakukan hanyalah dua langkah pertama saja. Lagipula, proses apa yang sedang berlangsung yang dimiliki oleh si user? IRC bot? Proses tersebut bisa dibunuh secara singkat dengan:

```
kill -9 ‘ps -aux |grep ^<username> |tr -s " " |cut -d " " -f2‘
```

Contoh, jika username adalah foo:

```
kill -9 ‘ps -aux |grep ^foo |tr -s " " |cut -d " " -f2‘
```

Hal itu sudah teratasi, sekarang beralih ke root password yang terlupakan.

Solusi yang diberikan di majalah Gazette adalah yang paling universal, tetapi bukan yang paling mudah. Dengan adanya LILO dan loadlin secara bersamaan, seseorang mungkin menge-boot dengan memberi parameter "single" untuk boot langsung ke shell default tanpa login ataupun tampilan password. Dari sana, dia bisa menghilangkan atau mengubah password sebelum mengetikkan “init 3“ untuk pindah ke mode multi-user. Jumlah reboot: 1

Dengan cara yang lain, jumlah reboot: 2

Justin Dossey

3.6 Beberapa petunjuk untuk sysadmin baru. *Jim Dennis,* jadestar@rahul.net

Buatlah sebuah /README.‘hostname‘ dan/atau sebuah /etc/README.‘hostname‘, yang kemudian dipelihara. *[Atau mungkin /usr/local/etc/README.‘hostname‘]*

Kemudian, sejak *hari pertama* mengadministrasi sebuah sistem catatlah di sebuah berkas log online. Anda bisa saja membuat baris 'vi /README.$(hostname)' di /bash_logout milik root. Cara lain untuk melakukan ini ialah menulis sebuah script su atau sudo yang isinya kira-kira:

```
function exit \
        { unset exit; exit; \
          cat ~/tmp/session.$(date +%y%m%d) \
          >> /README.$(hostname) && \
          vi /README.$(hostname)
          }
script -a ~/tmp/session.$(date +%y%m%d)
/bin/su.org -
```

(gunakan perintah typescript untuk membuat log dari sebuah sessiondan buat sebuah fungsi untuk mengotomatisasi penambahan dan update dari log).

Penulis mengakui bahwa penulis belum mengimplementasikan otomatisasi dari aturan – sejauh ini penulis hanya mengandalkan dari disiplin diri sendiri. Bagaimanapun penulis telah bermain-main dengan ide tsb. (bahkan hingga membuat prototipe dari script dan fungsi-fungsi shell tsb. di atas). Satu hal yang membuat penulis urung ialah perintah 'script' itu sendiri. Penulis berpendapat bahwa beliau harus mendapatkan sumber-sumbernya dan menambahkan beberapa command line parameters (untuk memberhentikan (stop/pause) script merekam dari command line) sebelum penulis berkomitmen untuk menggunakannya.

Saran terakhir penulis (untuk bagian ini):

PATH dari root seharusnya berisi 'PATH= /bin'

Itu saja. Tidak ada lagi yang lainnya. Semua yang dilakukan oleh root: (1) disediakan melalui symbolic link dari /bin atau dengan alias atau fungsi shell, atau (2) adalah script atau berkas biner di /bin, atau (3) diketik secara langsung dengan menyertakan path secara eksplisit.

Hal ini membuat seseorang yang bertindak sebagai root menyadari (terkadang menyakitkan) tentang bagaimana dia mempercayai berkas-berkas biner. Administrator yang bijak dari sebuah host multi-user akan mengecek secara berkala berkas-berkas /bin dan /.*history untuk mencari pola-pola dan lubang-lubang.

Administrator yang benar-benar termotivasi akan menemukan daerah-daerah yang bisa diotomatisasi, tempat-tempat di mana pengecekan kesehatan sistem bisa dimasukkan, dan pekerjaan- pekerjaan dengan hak/akses 'root' seharusnya dihindarkan sementara (menjalankan penyunting (editor), MTA dan program- program interaktif besar yang lain yang memiliki kelebihan membuat script secara terperinci, yang *mungkin* diikutsertakan di berkas transparan atau berkas data – seperti ./.exrc dari vi dan ./.emacs dari emacs dan $EXINIT yang lebih membahayakan lagi dan header/footer macros yang diikutsertakan). Biasanya perintah-perintah seperti itu bisa dijalankan dengan:

```
cp $data $some_users_home/tmp
su -c $origcommand $whatever_switches
cp $some_users_home/tmp $data
```

(...di mana spesifikasinya tergantung kepada masing-masing perintah).

Kebanyakan bermacam cara pencegahan yang terakhir ini agak berlebihan untuk sistem di rumah atau single-user – tapi ini adalah aturan yang sangat baik untuk sistem yang multi-user – terutama sistem yang bisa diakses oleh publik (seperti di netcom).

3.7 Cara mengkonfigurasi chooser dari xdm untuk pilihan host. *Arrigo Triulzi,* `a.triulzi@ic.ac.uk`

1. Suntinglah berkas yang menjalankan xdm (biasanya /etc/rc/rc.6 atau /etc/rc.local) sehingga berkas tsb. mengandung baris-baris berikut di bagian xdm startup.

```
/usr/bin/X11/xdm
exec /usr/bin/X11/X -indirect hostname
```

2. Suntinglah /usr/lib/X11/xdm/Xservers dan berikan tanda komentar pada baris-baris yang memulai server pada mesin lokal $yaitumemulai0:]Reboot, danitusaja.$

Penulis menambahkan ini karena ketika penulis, sesudah putus asa, mencoba memasangnya untuk subnet penulis di sini, penulis sudah menghabiskan waktu seminggu untuk memcahkan semua persoalannya.

Harap diingat: dengan SLS lama (1.1.1), untuk suatu alasan anda bisa membubuhkan -nodaemon setelah baris xdm – hal ini **TIDAK** berlaku untuk rilis setelah itu.