Het installeren van Linux op een ZIP disk met behulp van de ppa ZIP Drive Mini-Howto

John Wiggins, *jwiggins@comp.uark.edu*
Vertaald door Ellen Bokhorst, bokkie@nl.linux.org

v0.7, 26 januari 1998

Dit document is alleen van nut voor degenen met de printer poort versie van een ZIP-drive, en die een portable of backup Linux systeem op een ZIP-disk willen.

Inhoudsopgave

1 Disclaimer

OPMERKING: Ik heb geen idee of de IDE drive op dezelfde manier werkt als de printer poort versie aangezien ik er geen heb, dus vraag me er alsjeblieft niet om.

Het Debian installatie onderdeel was eigenlijk gewoon gekopieerd met weinig of geen wijziging door deze auteur. Daarom kan het zijn dat er in de instructies, duplicaten zijn.

Dit document gaat uit van het volgende:

- Je hebt een printer poort ZIP-drive (aangezien de ZIP Plus beide heeft, veronderstel ik dat dat net zo goed zal werken.)
- Je hebt Linux reeds geïnstalleerd en draaiend; dit document is niet voor een eerste installatie van Linux.
- Je hebt ppa ondersteuning in je huidige kernel of als een module, de ppa module is geladen.
- Het mountpoint voor de ZIP-disk is de /iomega directory.

2 Introductie

Dit document is onderverdeeld in vier basissecties waarvan ieder de belangrijkste elementen beschrijft hoe je een Linux systeem op een 100 MB ZIP disk kunt installeren met gebruik van een printer poort ZIP-drive. De eerste sectie beschrijft hoe de ZIP-disk in te stellen en is algemeen voor zowel Red Hat als Slackware distributie installaties. De tweede, derde, en vierde sectie beschrijven respectievelijk hoe RedHat 4.2, Slackware 2.2, en Debian 1.2 distributies op de ZIP-disk te installeren.

OPMERKING: Ik realiseer me dat Red Hat 5.0 nu op de markt is gebracht, maar tussen lesuren en werk, laten we goed en wel zeggen dat het ergens eind mei zal zijn eer ik er toe kom er mee te werken. Hopelijk zal ik de andere distributies aan het uittesten zijn.

2.1 Wat is nieuw

Ik heb eindelijk genoeg tijd (alhoewel ik eigenlijk zou moeten studeren voor een sociologietest...) om dit document bij te werken. Met dank aan iedereen die me opmerkingen mailde.

Nieuw aan dit document:

- De Debian distributie
- LILO op een diskette
- Nieuwe versie van Red Hat (4.2 Biltmore)
- Netwerk configuratie (Voor Red Hat; Niet getest)

2.2 Conventies

Geeft aan dat de volgende tekst commentaar is: ==>

Geeft iets aan dat het benoemen waard is:

```
    OPMERKING:
```

Duidt een scherm opname/capture aan:

```
            Tekst hier.
```

2.3 Updates

Kijk alsjeblieft voor enige updates, hoe zeldzaam ze ook mogen zijn, bij: *http://comp.uark.edu/˜jwiggins/linuxZIP/*

2.4 Referenties

- Installation-HOWTO
- SCSI-HOWTO
- NET-3-HOWTO (voor sectie 4.3.3)
- ZIP-Drive (mini-HOWTO)
- ParPort kernel patch (geeft toegang tot printer poort) *http://www.cyberelk.demon.co.uk/parport/*

2.4.1 Erkenningen/Medewerkers

Slackware 2.2 sectie, Michael Littlejohn *mike@mesa7.mesa.colorado.edu* was zo vriendelijk.

Debian 1.2 sectie, John D. Blair *jdblair@uab.edu* was zo vriendelijk.

LILO informatie en veel andere nuttige inzichten, hoffelijkheid van Darcy Boese *possum@niagara.com* en Javier Rodriguez *jrodrigu@nextgeninter.net.mx*

3 ZIP disk instellen

(Algemeen voor zowel Red Hat als Slackware distributies) Verzeker jezelf er van dat je toegang hebt tot de ZIP drive, vóórdat je begint; óf je hebt ppa in de kernel óf je hebt de ppa module geladen. Een gemakkelijke manier om dit uit te zoeken is door het bekijken van de uitvoer van dmesg:

==> **dmesg**

Mogelijk moet je dit middels een pipe doorgeven aan more omdat de uitvoer van dmesg tamelijk lang kan zijn. Hier is een deel van het mijne:

```
scsi0 : PPA driver version 0.26 using 4-bit mode on port 0x3bc.
scsi : 1 host.
  Vendor: IOMEGA    Model: ZIP 100          Rev: D.08
  Type:   Direct-Access                     ANSI SCSI revision: 02
Detected scsi removable disk sda at scsi0, channel 0, id 6, lun 0
SCSI device sda: hdwr sector= 512 bytes. Sectors= 196608 [96 MB] [0.1 GB]
 sda: Write Protect is off
 sda: sda1 sda2
```

Als je slechts iets ziet als:

```
scsi : 0 hosts.
scsi : detected total.
```

dan heb je SCSI-ondersteuning maar de ZIP werd niet gevonden.

3.1 ZIP-disk Partitioneren

Start fdisk op, om de ZIP-disk te partitioneren:

==> **fdisk /dev/sda/**

Hier is een deel van de partitietabel zoals ik 'm ingesteld heb:

```
Disk /dev/sda: 64 heads, 32 sectors, 96 cylinders
Units = cylinders of 2048 * 512 bytes

   Device Boot   Begin    Start      End   Blocks   Id  System
/dev/sda1            1        1       81    82928   83  Linux native
/dev/sda2           82       82       96    15360   82  Linux swap
```

Ik besloot een swap partitie aan te maken, aangezien ik in staat wilde zijn om dit met elke machine te gebruiken.

3.2 De ZIP-disk formatteren en mounten

Na het uitvoeren van fdisk, de nieuwe partitie formatteren:

==> **mke2fs -c /dev/sda1**

Maak vervolgens de swappartitie aan: (15360 blokken zoals fdisk aangaf)

==> **mkswap -c /dev/sda2 15360**

Als laatste, mount je de ZIP-disk:

==> **mount /dev/sda1 /iomega -t ext2**

3.3 Bootdisk aanmaken

Aangezien de ppa versie van de ZIP-drive geen echt SCSI-apparaat is, is het geen opstartbaar apparaat, en heeft daarom een bootdisk nodig met ppa in de kernel en niet als een module.

3.3.1 De kernel configureren en aanmaken

Als eerste is het nodig dat je een kernel met ppa ondersteuning hebt en niet als een laadbare module. Om de ppa optie te verkrijgen, selecteer je SCSI-support:

SCSI support (CONFIG_SCSI) [Y/m/n/?]

Plus, SCSI disk support:

SCSI disk support (CONFIG_BLK_DEV_SD) [Y/m/n/?]

En als laatste, onder de SCSI low-level drivers, staat de ppa ondersteuning:

IOMEGA Parallel Port ZIP drive SCSI support (CONFIG_SCSI_PPA) [Y/m/n/?]

Nogmaals, zorg dat je ppa niet insluit als een module, maar juist in de kernel. Tot zo ver, zonder het gebruik van de parport kernel patch (zie 1.4), zal de ppa driver niet toestaan dat de passieve poort van de ZIP drive wordt gebruikt voor een printer, dus mogelijk wil je de parallelle printer ondersteuning niet:

Parallel printer support (CONFIG_PRINTER) [N/y/m/?]

```
OPMERKING: Raadpleeg alsjeblieft de ZIP-Drive mini-HOWTO voor meer
informatie aangaande de ppa driver.
```

Als de kernel ééńmaal is geconfigureerd, maak je de kernel aan:

==> **make dep; make clean; make zImage**

De nieuwe kernel kan mogelijk worden gevonden in arch/i386/boot/zImage.

3.3.2 De kernel naar een diskette overbrengen

Nadat ik 4 aparte diskettes had, te wijten aan verschillende kernels en omdat ik verschillende parameters nodig had, (plus de belangrijke email die ik kreeg waarin stond hoe ik dit kon doen) heb ik een sectie over LILO opgenomen bestemd voor het aanmaken van een opstartdiskette.

LILO installatie Voor degenen die diverse kernels op e´´

endiskette(ophetmoment, zijndemijnetegroot)moetenhebbenofomwatvoorredendanookditwillen, ofgewooninstaatwillenzijnomargumentendoortegeven(zoalssingleusermode)ontvingikeenemailoverhoeLILOopeendiskettekanwordengenstalleerd.

Het ext2 bestandssysteem aanmaken Gebruik gewoon dezelfde opdracht als voor de ZIP-disk om een ext2 bestandssysteem op een diskette aan te maken:

==> **mke2fs -t /dev/fd0**

De essentiële bestanden kopiëren Zorg er vervolgens voor dat er een directory is voor een mountpoint, en mount de diskette (Ik gebruikte /mnt/floppy):

==> **mount /dev/fd0 /mnt/floppy -t ext2**

om op de juiste wijze te booten, heb je dezelfde bestanden die LILO op je huidige Linux installatie gebruikt, nodig.

```
         OPMERKING:  De bestandslokaties hier zijn van mijn computer en
                     dit hoeft niet voor iedereen gelijk te zijn.
```

==> **cp /boot/boot.b /mnt/floppy**

==> **cp /boot/map /mnt/floppy**

==> **cp /usr/src/linux/arch/i386/boot/zImage /mnt/floppy/vmlinuzDESK**

Om nu het configuratiebestand voor LILO aan te maken, mis ik nu het liloconfig programma... (Met dank aan Javier Rodriguez voor deze informatie) Maak als eerste het LILO configuratiebestand aan, /mnt/floppy/lilo.conf, voor de kernel(s) voor de ZIP-disk. Dit gebruikte ik zodat ik verschillende kernels had om mee te testen:

```
boot=/dev/fd0
map=/mnt/floppy/map
install=/mnt/floppy/boot.b
prompt
compact
timeout=50
image=/mnt/floppy/vmlinuzLAP
   label=Laptop
   root=/dev/sda1
   read-only
image=/mnt/floppy/vmlinuzDESK
   label=Desktop
   root=/dev/sda1
   read-only
image=/mnt/floppy/vmlinuzDESK
   label=rescue
   root=/dev/hdc1
   read-only
```

Ik heb twee kernels, één voor mijn 486 laptop die de math-co emulatie in de kernel vereiste en de andere voor mijn desktop. De rescue geeft me de mogelijkheid om een noodboot naar de harde schijf te kunnen maken.

Als laatste maar niet het minste, met de diskette nog steeds gemount, start je LILO op om het op een diskette te installeren met de opdracht:

==> **lilo -C /mnt/floppy/lilo.conf**

Als LILO éénmaal is geïnstalleerd op de diskette, sla dan de volgende twee stappen over, tenzij je er plezier in hebt om dit nog een keer over te doen :)

Slechts de Kernel installatie

```
        OPMERKING: Dit heeft geen betrekking op de LILO installatie.
```

Kopieer de nieuw aangemaakte kernel naar een diskette:

==> **cp arch/i386/boot/zImage /dev/fd0**

of

==> **cat arch/i386/boot/zImage > /dev/fd0**

Ja, er zijn veel manieren om de kernel naar een diskette te kopiëren, maar de laatste manier, mijn favoriet, is een beetje cryptischer. Probeer de > niet te vergeten tenzij je graag binaire bestanden bekijkt :)

De root en de swap op de diskette plaatsen

```
        OPMERKING: Dit heeft geen betrekking op de LILO installatie.
```

Zodra de kernel op de diskette staat, is het nodig om het root device op de ZIP-disk in te stellen: ==> **rdev /dev/fd0 /dev/sda1** Ik weet niet zeker of de volgende optie nodig is, maar toch deed ik het.

Instellen van de swap:

==> **rdev -s /dev/fd0 /dev/sda2**

4 Red Hat 4.2 installatie

Met alles in relatie tot een computer, wordt iets dat 3 maanden oud is als verouderd aangemerkt en heeft een upgrade nodig. Aangezien ik niet altijd de tijd zal hebben om dit document bij te werken met iedere update, zal ik proberen om op z'n minst 't bij te werken bij iedere andere versie. Voor de andere distributies geldt dat ze zo zullen blijven zoals 't is, tenzij de auteurs me updates willen sturen.

4.1 Persoonlijke setup

Voor mijn installatie heb ik gebruik gemaakt van:

- Kernel 2.0.30
- Iomega ppa disk drive
- Red Hat 4.2

4.2 Package installatie

Toen ik in het begin besloot om RedHat op een ZIP disk te installeren, bedacht ik dat het veel gemakkelijker zou zijn om gewoon een Red Hat opstartdiskette te gebruiken. Toen werd ik wakker geschud. Ik had het bijna voor elkaar om een echte bootdisk aan te maken, kreeg zelfs hulp van diverse mensen bij RedHat via e-mail maar begon het gehele project tenslotte op te geven totdat ik de –root optie van rpm ontdekte.

4.2.1 Welke pakketten

Ik vond welke pakketten ik zou moeten installeren door, door een bestand te bladeren dat ik vond bij één van de mirrors van RedHat. Dit bestand kan op iedere mirror worden gevonden bij:

```
redhat/redhat-4.2/i386/RedHat/base/comps
```

Voor deze installatie wilde ik netwerkondersteuning insluiten, maar als gevolg van RedHat's X netwerk-configuratie, zal ik het handmatig moeten gaan configureren, of liever gezegd, de netwerk-setupscripts in /etc/sysconfig/ proberen handmatig te configureren (zie sectie 4.3.3.)

Ik besloot om geen ontwikkelaarspakketten te installeren zoals de ZIP drive, op z'n minst is een niet bijgewerkte kernel versie nogal traag om iets gecompileerd te krijgen. Ik koos er ook voor om X niet te installeren, hoofdzakelijk vanwege diskruimte. Op een later tijdstip kan ik altijd nog proberen om mijn harddisk te mounten en een symbolische koppeling aan te maken met /usr om te bekijken of ik X werkend kan krijgen.

Hierna volgt een lijst met de pakketten die ik installeerde, in volgorde van installatie weergegeven. De met een * gemarkeerden zijn bijgewerkt vanaf Red Hat's errata. Tussen haakjes staat het bijgewerkte pakketnummer;

```
b.v. NetKit-B-0.09-6 was bijgewerkt tot NetKit-B-0.09-8 dus de naam op de
lijst zou zijn: *91) NetKit-B-0.09-6 (-8)
```

```
      (Lijst aangemaakt met de opdracht rpm --root /iomega -qa)

       1) setup-1.7-2                  2) pamconfig-0.51-2
       3) filesystem-1.3-1             4) MAKEDEV-2.2-9
       5) adduser-1.7-1                6) libc-5.3.12-18
       7) SysVinit-2.64-8              8) ash-0.2-8
       9) at-2.9b-2                   10) libtermcap-2.0.8-4
      11) bash-1.14.7-1               12) bc-1.03-6
      13) bdflush-1.5-5               14) cpio-2.4.2-4
      15) cracklib-dicts-2.5-1        16) tmpwatch-1.2-1
      17) crontabs-1.5-1             *18) db-1.85-10 (-11)
      19) dev-2.5.1-1                 20) diffutils-2.7-5
      21) etcskel-1.3-1               22) file-3.22-5
      23) fileutils-3.16-1            24) findutils-4.1-11
      25) grep-2.0-5                  26) groff-1.10-8
     *27) ld.so-1.7.14-4 (-5)         28) getty_ps-2.0.7h-4
      29) gzip-1.2.4-7                30) mingetty-0.9.4-3
     *31) initscripts-2.92-1 (93-1)   32) ed-0.2-5
      33) info-3.9-1                  34) ncurses-1.9.9e-4
      35) libg++-2.7.1.4-5           *36) pwdb-0.54-3 (-4)
      37) rootfiles-1.5-1            *38) pam-0.57-2 (-4)
      39) redhat-release-4.2-1        40) less-321-3
      41) mount-2.5l-2                42) zlib-1.0.4-1
      43) rpm-2.3.11-1                44) e2fsprogs-1.10-0
      45) sysklogd-1.3-15             46) tar-1.11.8-11
      47) passwd-0.50-7               48) gawk-3.0.2-1
      49) gdbm-1.7.3-8                50) gpm-1.10-8
      51) hdparm-3.1-2                52) kbd-0.91-9
      53) slang-0.99.37-2             54) newt-0.8-1
      55) kbdconfig-1.4-1             56) ncompress-4.2.4-7
     *57) sh-utils-1.16-4 (-5)        58) procinfo-0.9-1
     *59) logrotate-2.3-3 (4-1)       60) lilo-0.19-1
      61) losetup-2.5l-2              62) linuxthreads-0.5-1
     *63) mkinitrd-1.6-1 (7-1)        64) mailcap-1.0-3
     *65) man-1.4h-5 (j-1)            66) mt-st-0.4-2
      67) modules-2.0.0-5             68) mailx-5.5.kw-6
      69) net-tools-1.32.alpha-2      70) procmail-3.10-10
      71) procps-1.01-11              72) psmisc-11-4
```

```
     73) quota-1.55-4                 74) readline-2.0-10
     75) sed-2.05-6                   76) setconsole-1.0-1
     77) sendmail-8.8.5-4             78) shadow-utils-960530-6
     79) stat-1.5-5                   80) tcsh-6.06-10
     81) termcap-9.12.6-5             82) textutils-1.22-1
     83) time-1.7-1                   84) timeconfig-1.8-1
     85) util-linux-2.5-38            86) vim-4.5-2
     87) vixie-cron-3.0.1-14          88) which-1.0-5
     89) zoneinfo-96i-4               90) tcp_wrappers-7.5-1
    *91) NetKit-B-0.09-6 (-8)        *92) lpr-0.18-1 (19-1)
    *93) bind-4.9.5p1-2 (9.6-1)      *94) bind-utils-4.9.5p1-2 (9.6-1)
    *95) wu-ftpd-2.4.2b12-6 (b15-1)   96) anonftp-2.3-3
     97) zip-2.1-1                    98) unzip-5.12-5
     99) statserial-1.1-7            100) minicom-1.75-2
    101) lrzsz-0.12.14-1             102) dip-3.3.7o-9
    103) ppp-2.2.0f-3                104) portmap-4.0-3
    105) perl-5.003-8               *106) traceroute-1.0.4.4bsd-2 (1.4a5-1)
   *107) elm-2.4.25-7 (-8)           108) lynx-2.6-2
    109) ncftp-2.3.0-5               110) pine-3.95-2
    111) rdate-0.960923-1            112) apache-1.1.3-3
   *113) nfs-server-2.2beta16-7     *114) nfs-server-clients-2.2beta16-7
         (2.2beta16-8)                    (2.2beta16-8)
```

En met al het bovenstaande geïnstalleerd, heb ik nog 32MB over!

Updates; errata Als zovelen, hoop ik dat RedHat gebruikers weten, dat van sommige pakketten kan worden geconstateerd dat er zich onvolkomenheden met betrekking tot de beveiliging of andere problemen voor kunnen doen. Om deze reden geeft RedHat voor dergelijke pakketten updates uit. Ik heb de pakketten die ik had en die in bovenstaande lijst zijn gemarkeerd, waarvoor updates waren, bijgewerkt. Raadpleeg alsjeblieft de webpage van RedHat die betrekking hebben op de bijgewerkte pakketten bij:

http://www.redhat.com/support/docs/rhl/rh42-errata-general.html

```
     OPMERKING:Voordat je de pakketten kunt updaten, moet je
               ldconfig opstarten zoals in sectie 3.3.1.1 staat beschreven.
```

4.2.2 Hoe pakketten met rpm te installeren, zonder glint

Gebruik met rpm de –root optie om de gemounte directory als de root voor installatie aan te geven. Ik kwam er achter dat bij veel van de pakketten de installatie mislukte vanwege het preinstall of postinstall script dat niet juist werd uitgevoerd te wijten aan de afwijkende root directory, gebruik dus de –noscripts optie:

==> **rpm –root /iomega -i –noscripts PACKAGE.i386.rpm**

Ik ben er zeker van dat velen zullen opmerken, dat je een foutmelding zult krijgen zoals:

```
       failed to open /iomega/var/lib/rpm/pakket.rpm

       error: cannot open /iomega/var/lib/rpm/pakket.rpm
```

Dus maak de directory var/lib/rpm directory gewoon aan:)

==> **mkdir /iomega/var; mkdir /iomega/var/lib; mkdir /iomega/var/lib/rpm**

4.2.3 Welke kwam eerst, pamconfig of pam?

Als iemand ooit heeft geprobeerd om pamconfig te installeren, het zal klagen over het ontbreken van pam; en als je dan probeert om pam te installeren, dan zal pam klagen over het ontbreken van pamconfig! Hierover, als het kip en het ei probleem, piekerde ik een tijdje, maar dankzij de –nodeps flag, kunnen we pamconfig geforceerd installeren; bovendien ontbreken er bij pam meer afhankelijkheden dan bij pamconfig.

==> **rpm –root /iomega -i –nodeps –noscripts pamconfig-0.51-2**

4.3 Problemen na installatie van pakketten

Helaas is de disk nog niet geheel functioneel als alles eenmaal uitstekend is geïnstalleerd. Wat ik bedoel te zeggen is, dat als je nu probeert te booten met de diskette, je niet ver zult komen. Zodra init probeert op te starten, zul je twee prachtige foutmeldingen krijgen; beide klagen over een paar bestanden die niet konden worden gevonden die zouden worden aangemaakt als de scripts waren uitgevoerd.

4.3.1 Het geval van het ontbreken van /etc/ld.so.cache en libc.so.5

Als je probeerde te booten, kreeg je twee foutmeldingen, de eerste zal de afwezigheid van het /etc/ld.so.cache bestand zijn. De tweede klaagt over het ontbreken van libc.so.5.

/etc/ld.so.cache Zoals opgemerkt door veel lezers, werkten mijn vorige instructies niet helemaal zoals aangegeven. Om dit bestand aangemaakt te krijgen, zul je ldconfig uit moeten voeren terwijl de ZIP-disk nog gemount is:

==> **chroot /iomega /sbin/ldconfig**

Met dank aan Javier Rodriguez voor deze oplossing.

libc.so.5 Om het ontbreken van de lib op te lossen, zul je een symbolische link aan moeten maken die zou zijn aangemaakt door de installatie-scripts.

==> **cd /iomega/lib; ln -s libc.so.5.3.12 libc.so.5**

Met dank aan Darcy Boese voor deze oplossing.

4.3.2 Wachtwoord van root instellen

Net toen ldconfig werd uitgevoerd in 4.3.1.1, kon je net zo goed een root wachtwoord voor dit nieuwe systeem wijzigen/aanmaken:

==> **chroot /iomega passwd root**

4.3.3 Wat het installatie programma aanmaakte

```
OPMERKING: Dit is slechts een beknopte en bondige setup, waarvan ik niet
in de gelegenheid was om ze te testen, al was het maar om te zien
of het werkt. Theoretisch gezien zou het moeten werken, maar stuur
me alsjeblieft geen klachten mocht het niet werken.
```

Tijdens het onderzoeken van mijn 4.2 CD-ROM, kwam ik nogal iets interessants tegen; de source code voor het installatie programma. Ik vond het onder /misc/src/install en één van de dingen die ik van enig nut vond, was het bestand net.c. Hierin vond ik welke andere bestanden zouden zijn aangmaakt als het installatieprogramma zou zijn uitgevoerd. Het meeste ervan geeft je netwerkondersteuning (vandaar de naam net.c) maar zelfs als je geen netwerkkaart hebt, kun je altijd nog localhost gebruiken voor een netwerk (en apache zal melden dat het niet in staat is om een hostnaam vast te stellen). Het gaat om de volgende bestanden:

```
/etc/hosts
/etc/HOSTNAME
/etc/resolv.conf
/etc/sysconfig/network
/etc/sysconfig/network-scripts/ifcfg-eth0
(of ieder ander netwerk-apparaat dat je zou kunnen hebben).
```

4.3.4 Netwerk setup

Voor dit document wilde ik netwerkondersteuning voor mijn 3Com 3c595 snelle-ethernet kaart (die voor de laatste verscheidene maanden in een andere computer is gebruikt).

Als eerste had ik een naam nodig, en aangezien ik mijn eigen name server draai (wat een ander lang verhaal is) gaf ik mezelf de naam: *dash-dot.wig.org* (Ik vond het wel aardig klinken). Een naam zonder een IP is nogal zinloos, dus ik gebruikte een gereserveerde niet-internet geschikt netwerkadres 192.168.10.0 dat ook door mijn name server wordt gebruikt. Ook al wordt de HOSTNAME gewoonlijk opgeslagen in /etc/HOSTNAME, Red Hat controleert /etc/sysconfig/network op deze naam; dus laten we hier beginnen.

Voorbeeld van mijn /etc/sysconfig/network:

```
NETWORKING=yes
HOSTNAME=dash-dot
DOMAINNAME=wig.org
GATEWAY=
GATEWAYDEV=eth0
NS1=192.168.10.7
```

Dupliceer praktisch dezelfde informatie naar /etc/HOSTNAME, /etc/resolv.conf, en /etc/hosts:

/etc/HOSTNAME:

```
dash-dot.wig.org
```

/etc/resolv.conf:

```
search wig.org
nameserver ns.wig.org
```

/etc/hosts:

```
127.0.0.1 localhost
192.168.10.99 dash-dot.wig.org dash-dot
192.168.10.7  ns.wig.org ns
```

Red Hat configureert alle netwerk apparaten vanuit de scripts die in de directory /etc/sysconfig/network-scripts staan. De configuratie van enig netwerk apparaat wordt meestal eerst aangemaakt via het installatie programma dus ik moest deze configuratie bestanden handmatig aanmaken. Ze beginnen allemaal met ifcfg-XXX waar XXX het netwerk apparaat voorstelt dat ifconfig naar voren brengt; b.v. ppp0, eth0, enz.

Voor dit voorbeeld moest ik een bestand met de naam ifcfg-eth0 aanmaken met de volgende inhoud:

```
DEVICE=eth0
ONBOOT=yes
BOOTPROTO=none
BROADCAST=192.168.10.255
NETWORK=192.168.10.0
NETMASK=255.255.255.0
IPADDR=192.168.10.99
```

En als laatste maar niet het minste, om dit nu direct plaats te laten vinden start je, terwijl je je nog in de directory /etc/sysconfig/network-scripts bevindt, op:

==> **./ifup ifcfg-eth0 boot**

Hiermee wordt het script opgestart waarmee de netwerk interface wordt geconfigureerd als die interface zodanig is ingesteld dat ze tijdens 'boottime' opstart.

Raadpleeg voor verdere informatie alsjeblieft het NET-3-HOWTO document.

5 Slackware 2.2 installatie

```
OPMERKING: Dit aandeel is niet bijgewerkt.
```

5.1 Benodigdheden

- ZIP-Disk en Drive (uiteraard)
- Kernel met ZIP-ondersteuning
- ZIP-Howto (aanbevolen)
- 1.44"HD geformatteerde diskette
- 1 tot 2 uur tijd

5.2 Installatie

Ok, nu komt het leuke gedeelte: Uitzoeken welke bestanden op de systeemdisk nodig zijn, en welke pakketten je wilt (en passen) op je ZIP-drive.

Ik besloot dat het 't eenvoudigst was te beginnen met Slackware direct op de ZIP drive te installeren. Ik besloot tot deze aanpak voornamelijk omdat Slackware een kleinere distributie is dan Red Hat, en het zou makkelijker zijn om weg te laten wat ik niet wilde hebben. Dat en het feit dat ik de Slackware distributie toch al gebruikte, maakte het een vanzelfsprekende keuze.

Het installeren van Slackware op de ZIP-disk is makkelijk, voer als root het setup programma uit, en kies /iomega als de installatie partitie, stel de installatie partitie in, van waar de Slackware sources moeten worden geïnstalleerd (cdrom, hardrive, etc), selecteer install en volg de aanwijzingen.

5.3 Wat te installeren?

Het moeilijkste gedeelte is te beslissen wat toe te voegen en wat niet. Vanzelfsprekend heb je de 'A' series nodig (dit is ongeveer 8 meg), de rest is aan jou.

Ik kreeg het voor elkaar om de Slackware release te verminderen tot een respectabele installatie van 70 meg, dat bestond uit gcc/g++, perl, X11R6 (NIET ALLES ERVAN!), sendmail, online docs (Minus alle ontwikkelaars man pages, maar inclusief alle howto's), en een assortiment met andere nuttige hulpmiddelen, terwijl er ongeveer 10 meg vrij bleef voor gebruikersbestanden.

6 /etc/fstab aanmaken

(Algemeen voor zowel Red Hat als Slackware distributies)

Het laatste wat nog moet worden gedaan voor het herstarten is het aanmaken van het fstab bestand op wat spoedig de root partitie zal zijn. Het volgende zal je op z'n minst nodig hebben voor /iomega/etc/fstab:

```
/dev/sda1     /       ext2      defaults  1  1
/dev/sda2     none    swap      sw
none          /proc   proc      defaults  1  1
```

Sla het bestand op, en reboot met de zojuist aangemaakte bootdiskette en vermaak je!

(Mijn speciale dank aan Mike voor de waarschuwing over deze zeer belangrijke en cruciale stap. - John)

7 Debian 1.2 Installatie

```
OPMERKING: De auteur van deze sectie heeft me dit 11 Juni 1997 opgestuurd.
```

7.1 Benodigdheden

- Ppa ZIP drive en disk.
- 2 lege 1.44 diskettes
- Een complete set Debian installatie diskettes (bekijk de Debian installatie docs nogeens, als je niet meer weet hoe je deze aan kunt maken).
- Een aantal uren de tijd

7.2 Overzicht

Na een aantal uren worstelend gespendeerd te hebben aan dpkg, besloot ik dat het eenvoudiger zou zijn om de Debian "Rescue"disk zodanig aan te passen dat het de ppa ZIP drive zou herkennen. Dit bleek erg makkelijk te zijn. Je kunt deze gewijzigde disk dan gebruiken om de normale Debian basis systeeminstallatie te doorlopen. Zodra je het basissysteem hebt geïnstalleerd, kun je een opstartdiskette gebruiken om het nieuwe basissysteem mee op te starten en de installatie te voltooien met gebruik van dselect. Om deze techniek te kunnen gebruiken moet je twee kernels aanmaken - een met ppa en initiële RAM-disk ondersteuning, - en een andere zonder de RAM-disk ondersteuning.

Als je dat wilt, kun je alle stappen in sectie 2 overslaan en de Debian installatie procedure het formatteren van de ZIP-disk voor je af laten handelen.

7.3 De gewijzigde Rescue-disk aanmaken

De Debian rescue disk is een SYSLINUX stijl boot disk, die gebruik maakt van een DOS geformatteerde diskette en een speciale bootloader om te voorkomen dat MS-DOS wordt geladen. Deze disks zijn erg makkelijk aan te passen zodanig dat je je eigen aangepaste bootconfiguratie kunt starten. Het Debian "boot-floppies" pakket bestaat uit een set scripts om het proces bootdisks aanmaken, te automatiseren. Het is echter zo eenvoudig, dat ik het makkelijker vond om dit proces met de hand te wijzigen. Dit wijkt een beetje af van de Debian filosofie, maar ik ben er over heen :). Er staan korte en bondige instructies in het 'readme.txt' bestand op de Rescue diskette.

7.3.1 Gebruik dd (of RAWRITE onder DOS) om een nieuwe Rescue-disk aan te maken.

Kijk nog eens naar de Debian installatie docs als je niet meer weet hoe je dit doet.

7.3.2 Een nieuwe kernel aanmaken met ZIP ppa ondersteuning

Maak een nieuwe kernel aan met ZIP ppa ondersteuning (als in stap 3.3.1), maar dan ook met de configuratie van een RAM disk en initiële RAM-disk ondersteuning. Het is ook nodig om de msdos, fat, minix, ext2fs, en procfs bestandssystemen te configureren.

Configureer ook elke module die je graag op je uiteindelijke installatie op de ZIP disk wilt.

Zodra de kernel is geconfigureerd, aangemaakt met:

==> **make dep; make clean**

==> ***make bzImage***

Maak dan de modules aan met:

==> **make modules**

Je zult deze later installeren.

```
        OPMERKING: Zorg dat je 'make bzImage' gebruikt,
                   en niet 'make zImage'.
```

7.3.3 Mount de nieuwe Rescue disk.

==> **fdmount fd0**

of

==> **mount /dev/fd0 /mnt**

of

==> **wat dan ook :)**

7.3.4 Kopieer de kernel image

Kopieer de kernel image (op het i386 platform zal het te vinden zijn op arch/i386/boot/bzImage) naar 'linux' op de diskette.

7.3.5 Het 'rdev.sh' script wijzigen

Open het 'rdev.sh' script dat is geplaatst op de Rescue diskette met je favoriete editor. Verander de laatste regel:

van–:

```
'rdev /mnt/linux /dev/ram0'
```

naar–:

```
'rdev /mnt/linux /dev/sda1'
```

Je zult ook alle voorkomende /mnt/linux paden moeten wijzigen naar het gepaste path. Aangezien ik mijn diskettes mount onder /fd0, moest ik /mnt/linux wijzigen in /fd0/linux.

7.3.6 Start dit gewijzigde 'rdev.sh' script op

==> **./rdev.sh**

7.4 Het basissysteem installeren op de ZIP-drive

Boot je gewijzigde Rescue-disk. Als alles goed gaat zal je worden gepresenteerd op het bekende Debian menugeöriënteerde installatieproces, behalve dat het nu gewaar is van je ppa ZIP drive. Vervolg dit proces alsof je het systeem zou installeren op een gewone harddisk, maar mount /dev/sda1 als root en initialiseer /dev/sda2 als swap.

Vergeleken met het standaard installatie proces is er een uitzondering – installeer en/of configureer geen laadbare modules. Je zult de modules die je in stap 7.3.2 hebt aangemaakt, later installeren.

7.5 De opstartdiskette aanmaken

Je kunt een opstartdiskette aanmaken zoals staat beschreven in de stappen 3.3.1 tot 3.3.2, of, als je dat wilt, gewoonweg de "Create Boot Disköptie gebruiken gedurende de Debian installatie. Ik ben blij met deze tweede optie omdat ik een andere SYSLINUX opstartdiskette kreeg, die me toestond de begroetingsmelding te wijzigen zodat ik een soort aangepaste opstartdiskette kon beschrijven en het me in de gelegenheid stelde om aanvullende kernelargumenten in te vullen. Je kunt zelfs helpbestanden, die dan toegankelijk zijn via de functietoetsen, opnemen. Het is aan jou.

7.6 Reboot het systeem

Doe je opstartdisk in de drive en kies de 'Reboot' optie vanuit het installatie-menu

7.7 Configureer het basissysteem en maak het installatieproces af

Als het systeem herstart zal je een langzame maar volledig werkende Debian basisinstallatie hebben lopen vanaf je ppa ZIP drive. Ga vanaf dit punt gewoon verder met de installatie. Ik installeerde alle gewone UNIX utility's, samen met documentatie sets, make, gcc, libraries, en diverse nuttige bestandsmanipulatie utility's. Het resultaat is een zeer krachtig nood-opstartsysteem dat ik in nood voor ieder systeem op onze afdeling kan gebruiken.

```
OPMERKING: Je moet op z'n minst 'make' installeren om
de volgende stap te voltooien.
```

7.8 De modules installeren die je hebt aangemaakt in stap 7.3.2.

Als je make in de laatste stap hebt geïnstalleerd, zou je in staat moeten zijn om de harddisk partitie met je aangemaakte kernel te mounten, cd naar de juiste directory en start 'make modules_install' op.

Zo deed ik het:

==> **mount /dev/hda2 /mnt**

==> **cd /mnt/usr/src/linux**

==> **make modules_install**

```
OPMERKING: Het is niet nodig om het bestand /etc/fstab
te wijzigen, zoals in stap 5 werd uitgelegd.
Het Debian installatie proces heeft hier al voor gezorgd.
```

8 Verdere overwegingen

Pff!, einde in zicht. Na het vergaren van 31 ZIP-disks is mijn volgende experiment om pcmcia proberen werkend te krijgen voor mijn laptop.