

LTSP – Linux Terminal Server Project – v4.1

James McQuillan

<jam@LTSP.org>

Tom Welter

Nederlandse vertaling

ltsp@startstop.nl

Copyright © 2004 James A. McQuillan

Wijzigingen

Herziening 4.1.3	2004-06-20	Herzien door: jam
Herziening 4.1.3-1-nl	2005-03-02	Herzien door: tgw

GNU/Linux is een uitermate geschikt platform voor het gebruik van schijfloze zogenaamde 'thin clients' of werkstations. Het belangrijkste doel van dit document is duidelijk te maken hoe zulke werkstations met behulp van LTSP kunnen worden geïnstalleerd. Daarnaast behandelt dit document onderwerpen over thin clients in het algemeen.

Inhoudsopgave

<u>Inleiding</u>	1
<u>1. Aansprakelijkheid</u>	1
<u>2. Auteursrechten en licentie</u>	2
<u>Hoofdstuk 1. Hoe werkt het?</u>	3
<u>1.1. De stappen die een werkstation doorloopt</u>	3
<u>1.2. Laden van de kernel in het geheugen</u>	6
<u>Hoofdstuk 2. Installeren van LTSP op de server</u>	8
<u>2.1. Installeren van de LTSP hulpprogramma's</u>	8
<u>2.2. Installeren van de LTSP client pakketten</u>	8
<u>2.3. Configureren van de benodigde services voor LTSP</u>	12
<u>2.4. Werkstationspecifieke configuratie</u>	15
<u>2.5. Huidige configuratie weergeven</u>	18
<u>Hoofdstuk 3. Installeren van het werkstation</u>	19
<u>3.1. Opstarten met PXE</u>	19
<u>3.2. Opstarten met Etherboot</u>	20
<u>Hoofdstuk 4. Opstarten van het werkstation</u>	22
<u>Hoofdstuk 5. Afdrukken</u>	23
<u>5.1. Installeren gebruikerszijde</u>	23
<u>5.2. Installeren serverzijde</u>	23
<u>Hoofdstuk 6. Schermscripts</u>	26
<u>Hoofdstuk 7. Problemen oplossen</u>	28
<u>7.1. Problemen oplossen van het Etherboot floppy image</u>	28
<u>7.2. Problemen oplossen van DHCP</u>	28
<u>7.3. Problemen oplossen van TFTP</u>	31
<u>7.4. Problemen oplossen van het NFS root bestandssysteem</u>	32
<u>7.5. Problemen oplossen met de Xserver</u>	34
<u>7.6. Problemen oplossen met de Displaymanager</u>	34
<u>Hoofdstuk 8. Kernels</u>	38
<u>8.1. Beschikbare standaard LTSP kernels</u>	38
<u>8.2. Zelf een Kernel maken</u>	38
<u>Hoofdstuk 9. lts.conf items</u>	43
<u>9.1. lts.conf voorbeeldbestand</u>	43
<u>9.2. Beschikbare lts.conf parameters</u>	43
<u>Hoofdstuk 10. Lokale toepassingen</u>	51
<u>10.1. Voordelen van het lokaal draaien van programma's</u>	51
<u>10.2. Instellingen voor het gebruik van lokale toepassingen</u>	51
<u>10.3. Server Configuratie voor Lokale toepassingen</u>	52
<u>10.4. Configuratie van toepassingen</u>	53

Inhoudsopgave

<u>Hoofdstuk 10. Lokale toepassingen</u>	
<u>10.5. Het starten van lokale toepassingen</u>	54
<u>Hoofdstuk 11. Configuratievoorbeelden</u>	55
<u>11.1. Seriële Muis</u>	55
<u>11.2. PS/2 Wielmuis</u>	55
<u>11.3. USB printer op een ThinkNic</u>	55
<u>11.4. Geforceerd laden van de XFree86 3.3.6 Xserver op een werkstation</u>	55
<u>Hoofdstuk 12. Andere informatiebronnen</u>	56
<u>12.1. Online referenties</u>	56
<u>12.2. Gedrukte publicaties</u>	56

Inleiding

LTSP biedt een simpele manier voor het gebruik van goedkope werkstations als grafische of tekstterminals van een GNU/Linux server.

In traditionele kantooromgevingen staan vaak relatief zware PC's met een Intel processor op ieder bureau. Elke computer beschikt over een eigen harde schijf met een capaciteit van meerdere Gigabytes. Gebruikers bewaren hun data op de lokale harde schijf en backups worden zelden (of nooit) gemaakt.

Heeft het echt zin om zo'n zware computer op ieder bureau te hebben?

Wij vinden van niet.

Gelukking is er een andere manier. Met LTSP is het mogelijk een eenvoudige computer te gebruiken, zonder harde schijf, floppy of cdrom. Alles wat nodig is, is een geïnstalleerde 'bootable' netwerkkaart. Veel netwerkkaarten hebben een voorziening voor het plaatsen van een bootrom, klaar voor gebruik.

Tijdens de opstartfase haalt het schijfloze workstation zijn IP informatie en een kernel op van de server, en koppelt zijn root bestandssysteem via NFS aan de server.

Het workstation kan op drie manieren worden geconfigureerd:

- **Graphisch X Window Systeem interface**

Met X Windows kan het workstation gebruik maken van alle toepassingen op de server of op andere servers van het netwerk.

- **Tekst georiënteerde telnet-sessies**

Het workstation kan meerdere telnet-sessies met de server opstarten. Iedere telnet-sessie heeft zijn eigen virtuele scherm. Met Alt-F1 tot en met Alt-F9 kan tussen de verschillende sessies gewisseld worden.

- **Shell prompt**

Het workstation kan zodanig worden geconfigureerd dat een lokale 'bash shell' wordt opgestart. Dit is handig bij het oplossen van problemen met X Windows of NFS.

Het mooie is dat meerdere werkstations door een GNU/Linux server kunnen worden bediend. Hoeveel werkstations? Dat hangt af van de eigenschappen van de server en de soort toepassingen die gebruikt worden.

Het is niet ongebruikelijk om met 50 werkstations Mozilla en OpenOffice te laten draaien op een Dual P4-2.4 server met 4GB ram. Wij weten dat dit werkt. De gemiddelde belasting van de server komt zelden boven 1.0!

1. Aansprakelijkheid

Noch de auteur, noch de distributeur, noch enige andere medewerker aan dit document zijn op enigerlei wijze verantwoordelijk voor de fysieke, financiële, morele of andere vorm van schade als gevolg van het opvolgen van de suggesties in deze tekst.

2. Auteursrechten en licentie

Het auteursrecht (2004) berust bij James McQuillan. De tekst is te gebruiken onder de voorwaarden van de 'GNU Free Documentation License' welke bij deze middels verwijzing is opgenomen in deze tekst.

Hoofdstuk 1. Hoe werkt het?

Het opstarten van een schijfloos werkstation omvat verschillende stappen. Begrip voor wat er gedurende dit proces gebeurt maakt het een stuk makkelijker eventuele problemen op te lossen.

Er zijn vier essentiële services nodig voor het opstarten van een LTSP werkstation, te weten:

- DHCP
- TFTP
- NFS
- XDMCP

LTSP is erg flexibel. Elk van de bovengenoemde services kan door een of door meerdere servers worden geleverd. Als voorbeeld beschrijven we een setup waarbij alle genoemde services door een server worden geleverd.

1.1. De stappen die een werkstation doorloopt

1. Laadt de Linux kernel in het werkstation geheugen. Dit kan op ver schillende manieren, waaronder:

- a. Bootrom (Etherboot,PXE,MBA,Netboot)
- b. Floppy
- c. Harde schijf
- d. cdrom
- e. usb-geheugen

Elk van bovengenoemde opstartmethodes zal later in dit hoofdstuk worden uitgelegd.

2. Nadat de kernel in het geheugen is geladen wordt hij opgestart.

3. De kernel initialiseert het hele systeem en alle randapparatuur die herkend wordt.

4. Op dit moment wordt het echt leuk. Tijdens het laden van de kernel wordt ook een ramdisk 'image' geladen. De kernel commandoregel **root=/dev/ram0** zorgt ervoor dat die image als root map wordt aangekoppeld.

5. Normaalgesproken zal de kernel als hij klaar is met opstarten het programma **init** aanroepen. In dit geval echter, instrueren we de kernel in plaats daarvan een klein shell script te draaien. Dit gebeurt door op de kernel commandoregel de optie **init=/linuxrc** mee te geven.

6. Het **/linuxrc** script begint met het scannen van de PCI bus op zoek naar een netwerkkaart. Voor ieder gevonden PCI apparaat wordt een overeenkomstige vermelding gezocht in het bestand `/etc/nci.list`. Wanneer die gevonden wordt, wordt de naam van de driver module teruggegeven en de betreffende netwerk kaart module wordt geladen. Voor ISA kaarten MOET de driver module op de kernel commando regel wordt gespecificeerd samen met eventueel benodigde IRQ of IO-adres parameters.

7. Vervolgens wordt een kleine dhcp-client genaamd **dhclient** opgestart om nogmaals een dhcp-verzoek uit te voeren. Dit aparte dhcp-verzoek als gewone gebruiker (in 'user-space') is nodig omdat meer informatie benodigd is dan dat de bootrom opgehaald had in het eerste DHCP verzoek.

8. Nadat **dhclient** een antwoord heeft ontvangen van de server zal deze het script **/etc/dhclient-script** uitvoeren, dat de opgehaalde informatie gebruikt om de eth0 interface te configureren.
9. Tot op dit moment is het root bestandssysteem een ram disk geweest. Nu zal het /linuxrc script een nieuw root bestandssysteem gaan aankoppelen via NFS. De map die hiervoor gebruikelijkerwijs wordt geëxporteerd door de server is **/opt/ltsp/i386**. Het nieuwe bestandssysteem kan niet zomaar als / . aangekoppeld worden. Het moet eerst als /mnt worden aangekoppeld. Vervolgens wordt **pivot_root** uitgevoerd. pivot_root verwisseld het huidige bestandssysteem voor een nieuw. Als dat klaar is, is het NFS bestandssysteem aangekoppeld aan / , en het oude root bestandssysteem aan /oldroot.
10. Nadat het aankoppelen en pivoteren van het nieuwe root bestandssysteem uitgevoerd is, is het /linuxrc shell script klaar en is het tijd het echte **/sbin/init** programma uit te voeren.
11. Init leest **/etc/inittab** en begint met het opzetten van de werkstation werkomgeving. ('environment')
12. Een van de eerste items in het inittab bestand is het **rc.sysinit** commando dat wordt uitgevoerd terwijl het werkstation in de '**sysinit**' fase is.
13. Het rc.sysinit script maakt een 1 Mb ramdisk aan waarop dingen kunnen worden weggeschreven of gewijzigd.
14. De ramdisk wordt aangekoppeld als de map /tmp. Alle bestanden die beschreven moeten kunnen worden staan in werkelijkheid in de map /tmp. Uiteindelijk verwijzen 'symbolic links' naar deze bestanden.
15. Het /proc bestandssysteem wordt aangekoppeld.
16. Het bestand **lts.conf** en de daarin vermelde parameters voor het betreffende werkstation worden als omgevingsvariabelen geïnitieerd voor gebruik door het script rc.sysinit.
17. Indien het werkstation geconfigureerd is om over NFS te swappen zal de map **/var/opt/ltsp/swapfiles** gekoppeld worden als /tmp/swapfiles. Vervolgens zal indien dat nog niet gebeurd is, automatisch een swapfile worden aangemaakt. De grootte van de swapfile wordt geconfigureerd in **lts.conf**.

De swapfile wordt geactiveerd met het commando **swapon**.
18. Het **loopback** netwerk interface wordt geconfigureerd met **127.0.0.1** als ip-adres.
19. Indien 'Local apps' aangezet is wordt de map **/home** aangekoppeld zodat de toepassingen toegang hebben tot de map /home van de gebruikers.
20. Verschillende mappen worden aangemaakt in het /tmp bestandssysteem ten behoeve tijdelijke bestanden die nodig zijn zolang het werkstation aan staat. Mappen zoals
 - a. /tmp/compiled
 - b. /tmp/var
 - c. /tmp/var/run

- d. `/tmp/var/log`
- e. `/tmp/var/lock`
- f. `/tmp/var/lock/subsys`

worden aangemaakt.

21. Het bestand `/tmp/syslog.conf` wordt aangemaakt. Dit bestand vertelt **syslogd** naar welke server op het netwerk de log-informatie gestuurd moet worden. De server voor het systeemlog wordt gespecificeerd in het bestand `lts.conf`. Een symbolische koppeling `/etc/syslog.conf` verwijst naar het bestand `/tmp/syslog.conf`.
22. De **syslogd** daemon wordt opgestart, gebruikmakend van de instelling uit de vorige stap.
23. Als het `rc.sysinit` script klaar is wordt de controle overgegeven aan het programma `/sbin/init`, dat het runlevel verandert van **sysinit** naar **5**.

Hierdoor worden de in `/etc/inittab` geconfigureerde items en scripts uitgevoerd.

24. Standaard is `inittab` ingesteld om het `/etc/screen_session` script uit te voeren op `tt1`, `tty2` en `tty3`. Dit betekent dat 3 sessies tegelijkertijd worden opgestart. Het type sessie wordt bepaald door de **SCREEN_01**, **SCREEN_02** and **SCREEN_03** instellingen in `lts.conf`.

Er kunnen indien gewenst meer regels aan `inittab` worden toegevoegd om meer sessies op te starten.

25. Indien **SCREEN_01** is ingesteld op de waarde **startx** dan wordt het script `/etc/screen.d/startx` uitgevoerd dat de grafische interface van het X Windows systeem opstart.

In het bestand `lts.conf` staat een parameter **XSERVER**. Indien deze parameter ontbreekt zal het systeem proberen automatisch de videokaart te detecteren. Als er een PCI of AGP videokaart aanwezig is zullen de gevonden PCI Vendor Id en Device Id worden gebruikt om een overeenkomstige vermelding te vinden in het bestand `/etc/vidlist`.

Indien de kaart ondersteund wordt door Xorg 6.7 zal de `pci_scan` routine de naam van de driver module teruggeven. Indien de kaart alleen ondersteund wordt door XFree86 3.3.6, dan zal `pci_scan` de naam van de te gebruiken Xserver teruggeven. Het `startx` script kan deze twee onderscheiden doordat de oudere 3.3.6 servernamen beginnen met 'XF86_', terwijl de nieuwere Xorg Xserver modules beginnen met kleine letters, bv. *ati* of *trident*.

26. Indien Xorg wordt gebruikt, dan wordt het script `/etc/build_x3_cfg` aangeroepen om een XF86Config bestand te maken. Indien XFree86 3.3.6 wordt gebruikt, dan wordt het script `/etc/build_x3_cfg` aangeroepen om het XF86Config te maken. Deze bestanden staan in de map `/tmp`, die zoals gezegd, een ramdisk is die alleen voor het werkstation zichtbaar is.

Het XF86Config bestand wordt gebaseerd op de instellingen in het bestand `/etc/lts.conf`.

27. Nadat het XF86Config bestand aangemaakt is start het script **startx** de X server op met het nieuwe configuratiebestand.

28. De X server stuurt een *XDMCP* verzoek aan de LTSP server, die vervolgens een aanmeldscherm aanbiedt.

29. Op dit punt aangekomen kunnen de gebruikers inloggen. Zij krijgen dan een sessie op de server.

Dit leidt bij veel mensen in eerste instantie tot verwarring. Men zit aan een workstation, maar de sessie draait op de server. Alle commando's worden op de server uitgevoerd maar de uitvoer wordt weergegeven op het workstation.

1.2. Laden van de kernel in het geheugen

Het laden van de Linux kernel in het geheugen kan op verschillende manieren gebeuren.

- Boot ROM
- Lokale media

1.2.1. Boot ROM

- Etherboot

Etherboot is een populair open-source bootrom project. Het bevat drivers voor diverse populaire netwerkkaarten en werkt erg goed samen met LTSP.

Linux kernels moeten worden 'gemerkt' met het commando **mknbi-linux**. Hiermee wordt de kernel aangepast voor opstarten vanaf het netwerk door voor aan de kernel wat code toe te voegen en aan het einde de *initrd* er aan vast te plakken.

De met LTSP meegeleverde kernels zijn al gemerkt en gebruiksklaar voor Etherboot.

Etherboot kan ook op een floppy worden geschreven, handig om te testen.

- PXE

Onderdeel van de 'Wired for Management' specificatie van eind jaren '90 was een specificatie voor een bootrom technologie bekend onder de naam *Pre-boot Execution Environment*, meestal afgekort als **PXE**

Een PXE bootrom is in staat maximaal 32 kilo-byte te laden. De Linux kernel is aanzienlijk groter. Daarom configureren we PXE om een zogenaamde '2nd stage boot-loader' genaamd **pxelinux** te laden. *pxelinux* is klein genoeg, en kan vervolgens veel grotere bestanden laden zoals de Linux kernel.

- MBA

Managed Boot Agent (MBA) is een bootrom van het bedrijf **emBoot**. Dit was vroeger de afdeling Lanworks van 3Com. MBA is eigenlijk een combinatie van 4 bootroms in een. Het kan overweg met PXE, TCP/IP, RPL en Netware.

De implementatie van PXE door MBA werkt erg goed. Hij kan met *pxelinux* gebruikt worden om een Linux kernel op te starten.

TCP/IP kan ook gebruikt worden, maar de kernel moet dan wel eerst aangepast worden met een hulpprogramma genaamd **imggen**.

- Netboot

Netboot is net als Etherboot open–source software voor het maken van boot ROM images. Het verschil is dat het in feite een schil vormt om de NDIS drivers of de packetdrivers die worden meegeleverd met netwerkkaarten.

1.2.2. Lokale media

- Floppy disk

Er zijn twee manieren om een LTSP werkstation met een floppy op te starten. De eerste manier is door Etherboot weg te schrijven in de bootsector van een floppy. De bootcode wordt dan uitgevoerd, de netwerkkaart wordt geïnitieerd en de kernel wordt geladen van de netwerkserver.

Het is ook mogelijk de kernel samen met initrd op een floppy te zetten en op die manier op te starten. Het is echter sneller om de kernel over het netwerk te laden.

- Harde schijf

De harde schijf kan in combinatie met LILO of GRUB worden gebruikt om de Linux kernel en initrd te laden. Ook kan de Etherboot bootrom van de harde schijf worden geladen, zodat die dus werkt als bootrom.

- cdrom

Een bootable cdrom met een Linux kernel of een Etherboot image kan worden gebruikt.

- USB–geheugen

Net als bij een CD–ROM, floppy of harde schijf kan ook USB geheugen gebruikt worden om een etherboot module of een complete Linux kernel met initrd image te laden.

Hoofdstuk 2. Installeren van LTSP op de server

LTSP kan het beste worden gezien als een complete distributie van Linux. Het is een distributie boven op een andere distributie. Deze 'andere' distributie kan iedere gewenste distributie zijn. In feite is het zelfs geen voorwaarde dat de server Linux draait. De enige voorwaarde is dat de server het NFS (Network File System) protocol ondersteunt. Vrijwel alle Unix systemen kunnen dat. Er zijn zelfs Windows servers die als LTSP server kunnen worden geconfigureerd.

Er zijn drie stappen voor het installeren van een LTSP server

- Installeren van de LTSP hulpprogramma's
- Installeren van de LTSP client pakketten
- Configureren van de services ten behoeve van LTSP

2.1. Installeren van de LTSP hulpprogramma's

Sinds versie 4.1 heeft LTSP een hulpprogramma-pakket voor het installeren en onderhouden van LTSP client pakketten (de programma's die op het werkstation draaien), en voor de configuratie van de services op de server.

Het administratie hulpprogramma (in het engels 'utility') heet **ltspadmin** en het configuratie hulpprogramma heet **ltspcfg**. Beide zijn onderdeel van het **ltsp-utils** pakket.

Het *ltsp-utils* pakket is beschikbaar in zowel *RPM* als *TGZ* formaat. Kies een bestandsformaat en volg de betreffende instructies op.

2.1.1. Installatie van het RPM pakket

Download de laatste versie van het *ltsp*-pakket en installeer dat met het volgende commando:

```
rpm -ivh ltsp-utils-0.1-0.noarch.rpm
```

Het bovenstaande commando installeert de hulpprogramma's op de server.

2.1.2. Installeren van het TGZ pakket

Download de laatste versie van het *ltsp-utils* TGZ pakket en installeer dat met de volgende commando's:

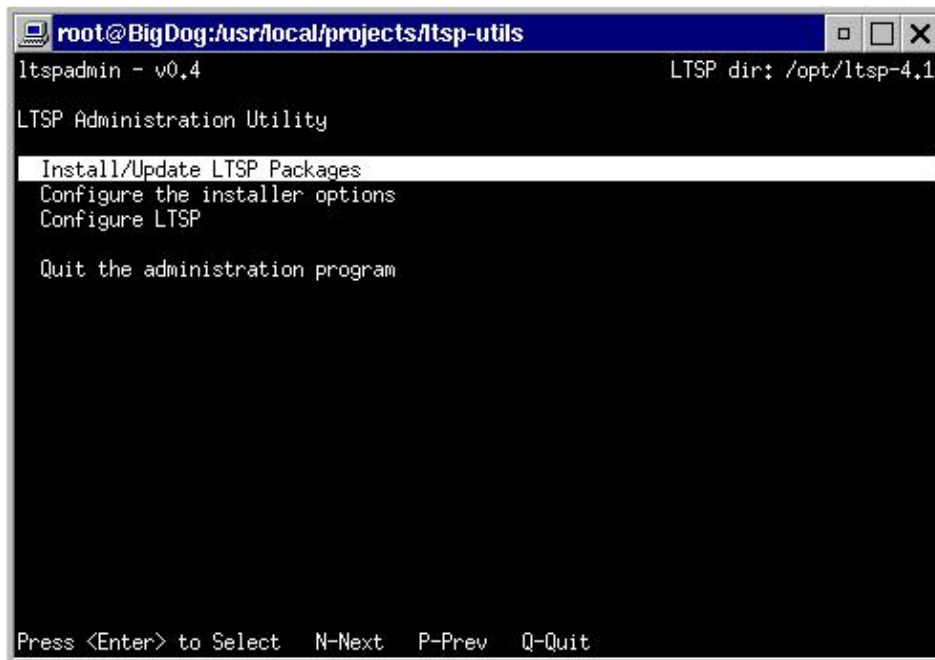
```
tar xzf ltsp-utils-0.1-0.noarch.tgz
cd ltsp_utils
./install.sh
cd ..
```

De bovenstaande commando's installeren de hulpprogramma's op de server. Gebruik deze op niet-RPM systemen.

2.2. Installeren van de LTSP client pakketten

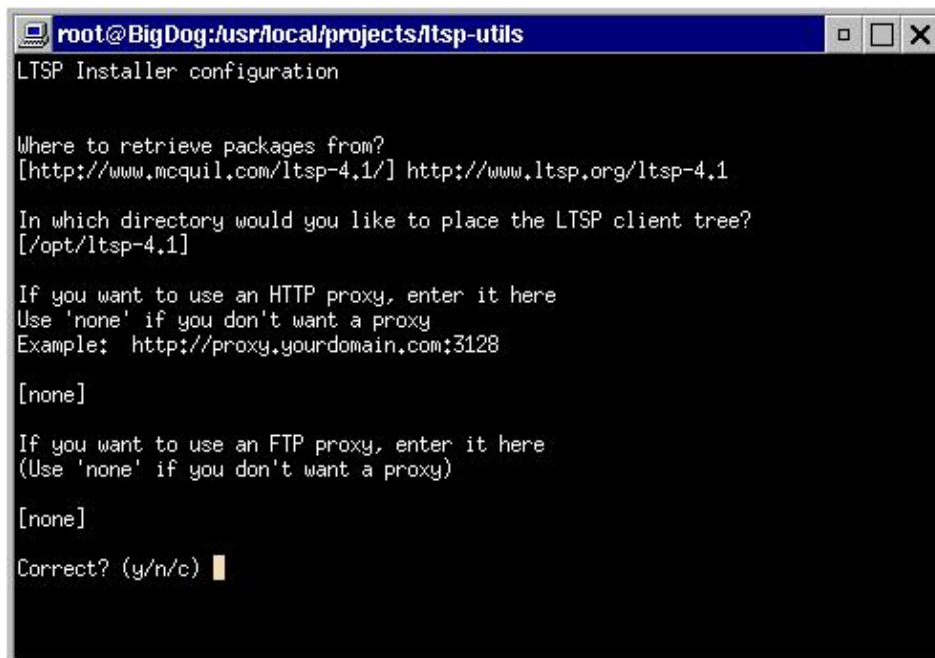
Als de *ltsp*-hulpprogramma's geïnstalleerd zijn kan het programma **ltspadmin** worden uitgevoerd. Dit hulpprogramma dient voor het beheren van de LTSP client pakketten. Het maakt contact met de LTSP downloadpagina en haalt de lijst met beschikbare pakketten op. Opstarten van **ltspadmin** levert het volgende

scherm op:



Figuur 2–1. LTSP installer – hoofdscherm

In dit scherm kan gekozen worden voor "Install/Update". Als het hulpprogramma voor de eerste keer wordt gebruikt dan wordt het 'Installer Configuration' scherm getoond:



Figuur 2–2. LTSP installer – Configuration scherm

In het *configuration* scherm kunnen verschillende waarden worden opgegeven voor het downloaden en installeren van LTSP pakketten:

Where to retrieve packages from?

(Vanaf waar moeten de pakketten worden opgehaald?) Dit is een URL die wijst naar de plaats van de pakketten. Gewoonlijk is dit `http://www.ltsp.org/ltsp-4.1`, maar als de pakketten vanaf een lokaal bestandssysteem moeten worden geïnstalleerd kan ook `file:` worden gebruikt. Als de pakketten bijvoorbeeld op een cd staan die aangekoppeld is als `/mnt/cdrom` dan zou `file:///mnt/cdrom` (let op: 3 slashes) moeten worden opgegeven.

In which directory would you like to place the LTSP client tree?

(In welke map moeten de LTSP mappen worden geïnstalleerd?) Dit is de map waar de mapstructuur van LTSP moet worden geïnstalleerd. Gewoonlijk is dit `/opt/ltsp`. De map wordt aangemaakt als hij nog niet bestaat.

In deze map worden de root mappen voor iedere architectuur aangemaakt. Op dit moment worden alleen x86 werkstations officieel ondersteund, maar er zijn verschillende mensen bezig met 'ports' naar andere architecturen zoals PPC en Sparc.

HTTP Proxy

Als de server achter een firewall staat en al het webverkeer door een proxy heen moet, kan hier de installer worden geconfigureerd om de proxy te gebruiken. Er moet een URL worden ingevuld inclusief het gebruikte protocol en poortnummer. Bijvoorbeeld:
`http://firewall.yourdomain.com:3128`.

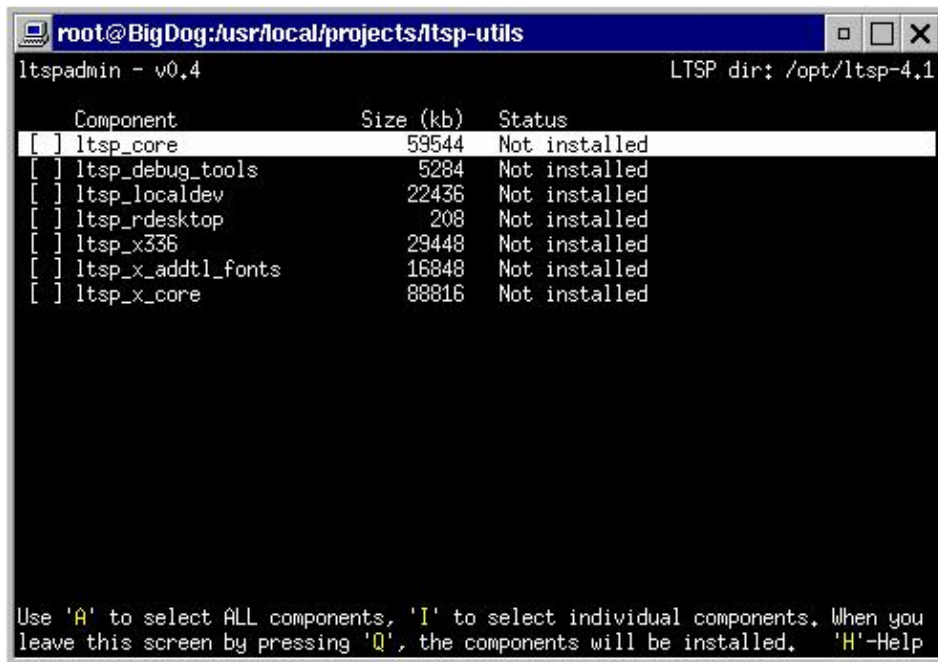
Als geen proxy nodig is kan worden volstaan met "none".

FTP Proxy

Als de pakketten op een FTP server staan of als een een FTP proxy gepasseerd moet worden dan kan dat hier worden opgegeven. De syntax is identiek aan de die van de HTTP proxy hierboven.

Als een proxy niet nodig is kan worden volstaan met "none".

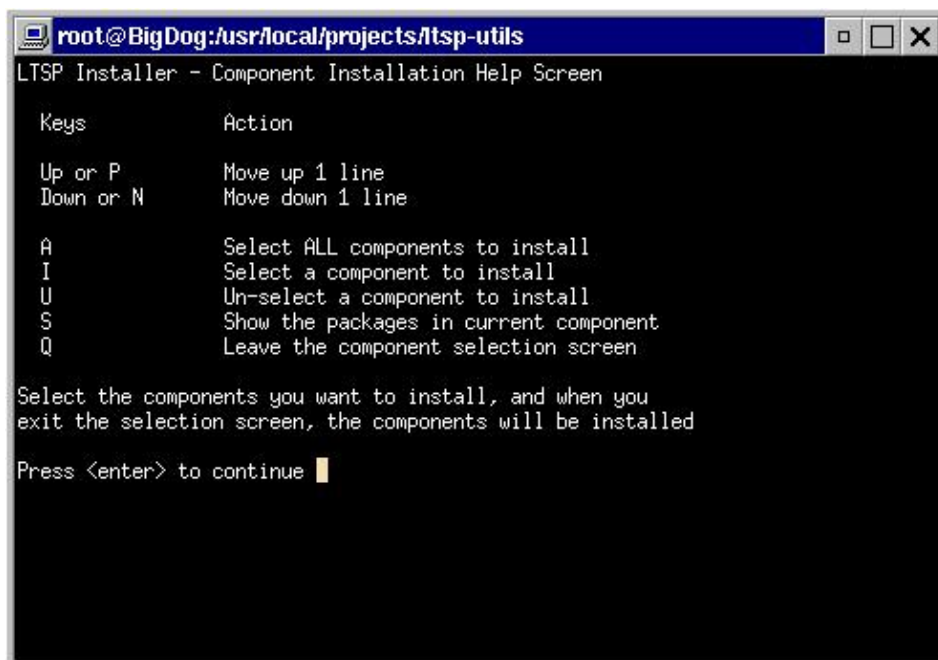
Als het configuratiescherm wordt afgesloten gaat de installer bij de opgegeven URL informatie ophalen over de lijst van op dat moment aanwezige installeerbare componenten.



Figuur 2–3. LTSP installer – Componentenlijst

Selecteer de componenten die geïnstalleerd moeten worden. Verplaats de oplichtende lijn naar de gewenste component en druk op 'T'. Het is ook mogelijk 'A' in te toetsen waarmee alle componenten geselecteerd worden. Dit is meestal het geval. Op deze manier kan de grootste range van mogelijk hardware in werkstations worden ondersteund.

Er zijn verschillende sneltoetsen voor dit scherm. Hulp is beschikbaar door 'H' in te drukken.



Figuur 2–4. LTSP installer – Help scherm

Voor het weergeven van de lijst met pakketten van een bepaalde component moet 'S' worden ingetoetst. De op dit moment geïnstalleerde versies worden getoond samen met de meest recente beschikbare versies.

```

ltspadmin - v0.4                               LTSP dir: /opt/ltsp-4.1
Package      Installed ver  Available ver  New package available
ltsp-audiofile  None          1.0-0          New package available
ltsp-bash       None          1.0-0          New package available
ltsp-busybox    None          1.0-0          New package available
ltsp-devfsd     None          1.0-0          New package available
ltsp-e2fsprogs  None          1.0-0          New package available
ltsp-esd        None          1.0-0          New package available
ltsp-getltscfg  None          1.0-0          New package available
ltsp-glibc      None          1.0-0          New package available
ltsp-haltsys    None          1.0-0          New package available
ltsp-libgcc_s   None          1.0-0          New package available
ltsp-libpng     None          1.0-0          New package available
ltsp-libvncserver None         1.0-0          New package available
ltsp-lp_server  None          1.0-0          New package available
ltsp-ltspinfo   None          1.0-0          New package available
ltsp-ltsp-tree  None          1.2-0          New package available
ltsp-modutils   None          1.0-0          New package available
ltsp-ncurses    None          1.0-0          New package available
ltsp-net-tools  None          1.0-0          New package available
ltsp-open       None          1.0-0          New package available
N-Next  P-Prev  Q-Quit
    
```

Figuur 2–5. LTSP installer – Package list

Nadat de gewenste componenten zijn geselecteerd kan het 'component selection' scherm worden afgesloten. De installer vraagt dan of de geselecteerde pakketten echt moeten worden geïnstalleerd of bijgewerkt. Als 'Y' wordt ingetoetst worden de pakketten opgehaald en geïnstalleerd.

2.3. Configureren van de benodigde services voor LTSP

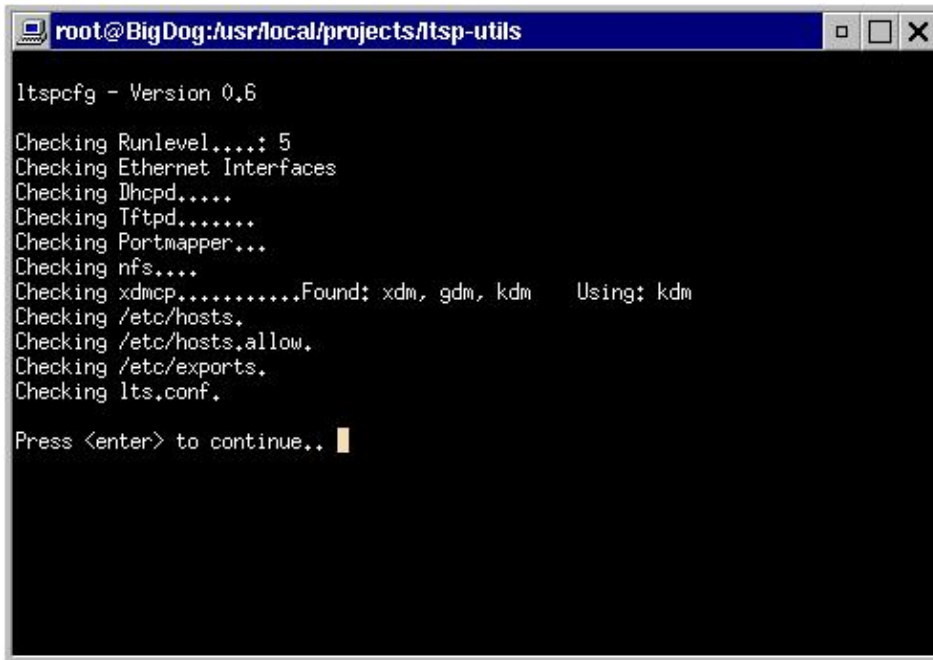
Er zijn vier essentiële services nodig voor de ondersteuning van opstartende LTSP werkstations, te weten:

- DHCP
- TFTP
- NFS
- XDMCP

Het **ltspcfg** hulpprogramma wordt gebruikt voor het configureren van deze services en tevens voor een aantal andere aan LTSP gerelateerde zaken.

ltspcfg kan worden opgestart vanuit **ltspadmin**, of vanaf de commandoregel door middel van het commando **ltspcfg**.

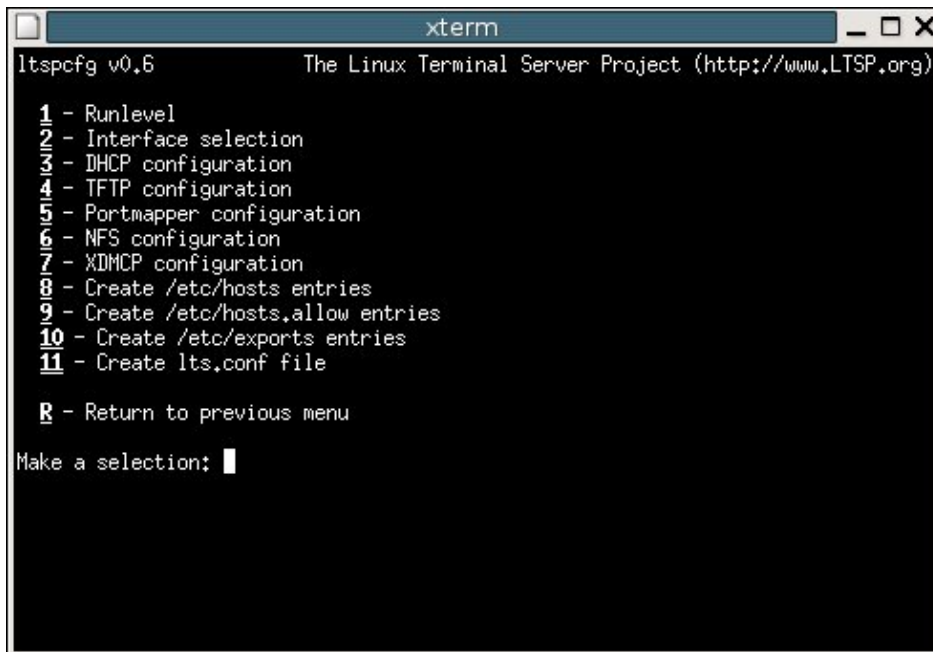
Wanneer **ltscfg** wordt opgestart onderzoekt het de server om te zien wat op dat moment geïnstalleerd is en draait. Het onderstaande scherm verschijnt:



Figuur 2–6. ltspcfg – Initieel Scherm

Dit laat zien waar het hulpprogramma naar zoekt.

Druk 'C' om het configuratiemenu te zien. Ieder item uit het configuratiemenu moet worden bekeken om er zeker van te zijn dat alles goed is ingesteld om LTSP werkstations te kunnen bedienen.



Figuur 2–7. ltspcfg – Initial screen

1 – Runlevel

Het commando **Runlevel** wordt gebruikt door het programma **init**. In Linux en Unix wordt het systeem verondersteld in een zeker "Runlevel" te zijn. Runlevel 2 of 3 worden gewoonlijk gebruikt om aan te geven dat het systeem in tekstmodus werkt. Runlevel 5 geeft aan dat het systeem in grafische modus werkt, al of niet op een netwerk.

Voor een LTSP server wordt traditioneel Runlevel 5 gebruikt. De meeste systemen zijn standaard al geconfigureerd om in runlevel 5 NFS en XDMCP services te leveren. Dit hulpprogramma configureert de systemen waarvoor dat niet het geval is.

2 – *Interface selectie*

Voor systemen met meerdere netwerkkaarten kan hier worden opgegeven via welke netwerk interface de werkstations verbonden zijn.

Door de juiste interface op te geven kan configuratie hulpprogramma op correcte wijze ander configuratiebestanden zoals `dhcpd.conf` en `/etc/exports` aanmaken.

3 – *DHCP configuratie*

DHCP moet worden geconfigureerd om de juiste velden aan de werkstations te kunnen doorgeven. Het gaat om onder andere de volgende velden: **fixed-address**, **filename**, **subnet-mask**, **broadcast-address** en **root-path**.

Door dit menu te selecteren kan het bestand `dhcpd.conf` worden aangemaakt en kan **dhcpd** worden ingesteld op te starten als het systeem wordt aangezet.

4 – *TFTP configuratie*

TFTP wordt gebruikt door het werkstation om de Linux kernel te downloaden. Daarvoor moet de **tftpd** daemon op de server worden geactiveerd.

5 – *Portmapper configuratie*

De **Portmapper** wordt gebruikt door RPC services zoals NFS

6 – *NFS configuratie*

NFS is de service die er voor zorgt dat lokale mappen kunnen worden aangekoppeld door externe computers. Dit is nodig voor LTSP omdat de werkstations hun root bestandssysteem op de server hebben staan.

Dit menu item zorgt er voor dat NFS gelijktijdig met het systeem opstart. Het aanmaken van het betrokken configuratie bestand `/etc/exports` wordt verderop in deze sectie beschreven.

7 – *XDMCP configuratie*

XDMCP is het "X Display Manager Protocol". Om een aanmeldscherm te krijgen stuurt de X server een XDMCP verzoek naar de displaymanager op de server.

Veel gebruikte displaymanagers zijn **XDM**, **GDM** en **KDM**. Dit menu item toont alle gevonden displaymanagers en geeft aan welke op dit moment geconfigureerd voor gebruik.

Uit veiligheidsoverwegingen wordt het de displaymanager standaard NIET toegestaan om externe werkstations te bedienen. Dit is meestal de oorzaak van het beruchte **Grijze scherm met grote X cursor**. `ltspcfg` is meestal in staat de displaymanager zo te configureren dat externe werkstations wel een verbinding kunnen krijgen.

8 – Aanpassen van */etc/hosts*

Verschillende services moeten voor hun werking het ip–adres van het werkstation kunnen vertalen in een hostnaam. Het is mogelijk dit voor elkaar te krijgen met de Berkeley Internet Naming Daemon (BIND) maar dan moeten de *reverses* goed worden ingesteld. Eigenlijk is dat ook de manier waarop het hoort, maar de configuratie van BIND gaat voor dit document en voor *ltspcfg* iets te ver.

Een veel eenvoudiger methode is om de ip–adressen en hostnamen op te nemen in het bestand */etc/hosts*.

9 – Aanpassen van */etc/hosts.allow*

Een aantal services maakt gebruik van een veiligheidslaag bekend onder de naam *tcpwrappers*. Deze wordt geconfigureerd met het bestand */etc/hosts.allow*. Dat bestand wordt door dit menu item configureerd.

10 – Aanpassen van */etc/exports*

Dit is het bestand dat NFS gebruikt om vast te stellen welke mappen mogen worden aangekoppeld door externe machines. Het bestand wordt met dit menu–item aangemaakt danwel aangepast.

11 – Aanmaken van *lts.conf*

De configuratie van ieder werkstation wordt bepaald door de items in het bestand *lts.conf*. Voor min of meer moderne werkstations met een PCI bus zijn eigenlijk geen aanpassingen in dit bestand nodig, maar het moet wel bestaan. Dit menu–item maakt een standaard *lts.conf* aan.

2.4. Werkstationspecifieke configuratie

Nu is het moment aangebroken om de LTSP server in te lichten over uw specifieke werkstation. Er zijn drie bestanden die informatie bevatten over het werkstation:

1. */etc/dhcpd.conf*
2. */etc/hosts*
3. */opt/ltsp/i386/etc/lts.conf*

2.4.1. */etc/dhcpd.conf*

Het werkstation heeft een ip–adres en andere informatie nodig. Het volgende kan van de server worden verkregen:

- ip–adres
- hostnaam
- Server ip–adres
- Default gateway
- Padnaam van de te laden kernel
- Server and map voor root bestandssysteem

Voor dit voorbeeld hebben we DHCP gekozen om ip–adressen uit te delen aan de werkstations.

Tijdens het draaien van het *lts_initialize* script wordt een voorbeeld *dhcpd.conf* bestand geïnstalleerd. Dat bestand heet */etc/dhcpd.conf.example*. U kunt dat bestand kopiëren naar */etc/dhcpd.conf* als het gebruikt moet worden voor de DHCP configuratie. Het is wel nodig de delen aan te passen die betrekking hebben op uw werkstation en serverinstallatie.

```

default-lease-time      21600;
max-lease-time          21600;

option subnet-mask      255.255.255.0;
option broadcast-address 192.168.0.255;
option routers          192.168.0.254;
option domain-name-servers 192.168.0.254;
option domain-name      "ltsp.org";
option root-path        "192.168.0.254:/opt/ltsp/i386";

shared-network WORKSTATIONS {
    subnet 192.168.0.0 netmask 255.255.255.0 {
    }
}

group {
    use-host-decl-names on;
    option log-servers 192.168.0.254;

    host ws001 {
        hardware ethernet 00:E0:18:E0:04:82;
        fixed-address 192.168.0.1;
        filename "/lts/vmlinuz.ltsp";
    }
}

```

Figuur 2–8. `/etc/dhcpd.conf`

Vanaf LTSP versie 2.09pre2 is het niet langer nodig een specifieke kernel op te geven. De standaard kernel ondersteund nu alle netwerkkaarten die Linux kent. Er worden twee kernel bestanden meegeleverd in het LTSP pakket. Een is bijgewerkt met de Linux Progress Patch (LPP) en de ander niet. De bestandsnamen zijn:

```

vmlinuz-2.4.9-ltsp-5
vmlinuz-2.4.9-ltsp-lpp-5

```

Het is u misschien opgevallen dat de kernel in `/tftpboot/lts` staat, maar dat in de "filename" parameter in `/etc/dhcpd.conf` de inleidende `/tftpboot` ontbreekt. De reden is dat in Redhat 7.1 en later, TFTP wordt opgestart met de '-s' optie. Daardoor draait de tftpd in *secure* mode, veilige modus. Dat betekent dat tftpd bij het opstarten een **chroot** uitvoert naar de map `/tftpboot`. Daarna zijn alle bestandsnamen voor tftpd relatief ten opzichte van die map.

In andere Linux distributies is deze '-s' optie misschien niet standaard ingeschakeld voor tftp. In dat geval zal de voorlooptekst `'/tftpboot'` moeten worden toegevoegd.

2.4.2. `/etc/hosts`

IP adres vertalen naar hostnaam

Computers kunnen prima met elkaar communiceren via ip-adressen. Pas als mensen zich er mee gaan bemoeien moeten er opeens namen gegeven worden want wij kunnen geen nummers onthouden. Daarom is er DNS, of het bestand `/etc/hosts`. In het algemeen is de vertaling van ip-adres naar hostnaam niet noodzakelijk. Voor LTSP omgevingen is het echter wel noodzakelijk. Dat is omdat anders NFS sputtert over toegangsrechten als het werkstation zijn root bestandssysteem wil aankoppelen.

Daarnaast zijn er, als het werkstation niet opgenomen is in `/etc/hosts`, problemen te verwachten met de **GDM** of **KDM** displaymanagers.

2.4.3. /opt/ltsp/i386/etc/ltsp.conf

Er zijn een aantal configuratie-items die kunnen worden ingesteld in het bestand `ltsp.conf`.

Het bestand `ltsp.conf` heeft een eenvoudige syntax, bestaande uit meerdere secties. Er is een standaard sectie genaamd **[default]**, en er zijn secties voor individuele werkstations. De workstation kunnen worden aangeduid met hostnaam, ip-adres of MAC adres (hardware adres).

Een gebruikelijke `ltsp.conf` ziet er als volgt uit:

```
#
# Config file for the Linux Terminal Server Project (www.ltsp.org)
#

[Default]
    SERVER            = 192.168.0.254
    XSERVER           = auto
    X_MOUSE_PROTOCOL = "PS/2"
    X_MOUSE_DEVICE   = "/dev/psaux"
    X_MOUSE_RESOLUTION = 400
    X_MOUSE_BUTTONS  = 3
    USE_XFS           = N
    LOCAL_APPS       = N
    RUNLEVEL          = 5

[ws001]
    USE_NFS_SWAP      = Y
    SWAPFILE_SIZE     = 48m
    RUNLEVEL          = 5

[ws002]
    XSERVER           = XF86_SVGA
    LOCAL_APPS       = N
    USE_NFS_SWAP      = Y
    SWAPFILE_SIZE     = 64m
    RUNLEVEL          = 3
```

Voorbeeld 2–1. `ltsp.conf` file

Hieronder volgt een lijst van enkele van deze items:

XSERVER

Indien de videokaart een PCI kaart is die ook ondersteund wordt door X.org 6.7.0 dan is alleen het `ltsp_x_core` pakket nodig. Dit bevat alle driver modules voor X4.

Voor LTSP zijn diverse XFree86 3.3.6 pakketten beschikbaar voor het geval de videokaart niet door X.org 6.7.0 wordt ondersteund.

Men kan in `ltsp.conf` voor ieder workstation apart items instellen, of men kan standaardwaarden instellen voor items die door alle workstations worden gebruikt.

Ons workstation heeft een Intel i810 video chipset die automatisch kan worden gedetecteerd. Er is dus geen `XSERVER` item nodig in het `ltsp.conf` bestand. Het item `XSERVER` mag echter wel, indien gewenst, worden ingesteld. Het is ook mogelijk de waarde 'auto' op te geven, om aan te geven dat hij

automatisch gedetecteerd zal worden.

RUNLEVEL

We willen dat het werkstation in grafische modus draait dus het runlevel moet op '5' worden gezet. Dit gebeurt door weer een ander item in het bestand `lts.conf`.

2.5. Huidige configuratie weergeven

Met `ltspcfg` is het ook mogelijk een overzicht van de huidige status van alle services te krijgen. Druk in het hoofdmenu van `ltspcfg` op 'S' en het overzicht wordt getoond.

```

xterm
ltspcfg v0.6      The Linux Terminal Server Project (http://www.LTSP.org)
Interface IP Address Netmask Network Broadcast Used
eth0      192.168.0.254 255.255.255.0 192.168.0.0 192.168.0.255 <-----

Service Installed Enabled Running Notes
-----
dhcpd    Yes     no     no     Version 3
tftpd    Yes     Yes    Yes    No '-s' flag
portmapper Yes     no     Yes
nfs      Yes     Yes    Yes
xdmcp    Yes     no     Yes    xdm, gdm Using: gdm

File Configured Notes
----
/etc/hosts          Yes
/etc/hosts.allow   Yes
/etc/exports        Yes
/opt/ltsp/i386/etc/lts.conf Yes

Configured runlevel: 5 (value of initdefault in /etc/inittab)
Current runlevel: 5 (output of the 'runlevel' command)

Installation dir...: /opt/ltsp

Press <enter> to continue..
    
```

Figuur 2–9. ltspcfg – Huidige Status

Hoofdstuk 3. Installeren van het werkstation

Als de server geïnstalleerd is wordt het tijd om ons te richten op het installeren van het werkstation

In het LTSP project gaat het om wat er gebeurt als de kernel in het geheugen geladen is. Er zijn verschillende manieren om de kernel in het geheugen te krijgen, zoals Etherboot, Netboot, PXE en floppy.

3.1. Opstarten met PXE

Als er PXE in de netwerkkaart of PC zit dan kan dat gebruikt worden om de kernel te laden. PXE is een bootrom technologie vergelijkbaar met Etherboot en Netboot.

Mogelijk moet PXE eerst geactiveerd worden op de netwerkkaart. Ook kan het zijn dat opstartvolgorde in het BIOS moet worden aangepast, zodat "Boot from Lan" op de eerste plaats staat in plaats van bv floppy of harde schijf.

PXE heeft de beperking dat alleen bestanden kleiner dan 32 kb geladen kunnen worden. De Linux kernel is aanzienlijk groter dus die kan niet direct worden geladen met PXE. Daarom moet eerst een zogenaamd 'Network Bootstrap Program', kortweg NBP, geladen worden.

Er is een NBP voor Linux beschikbaar met de naam **pxelinux.0**. Het is onderdeel van het **syslinux** pakket van kernel ontwikkelaar H. Peter Anvin.

Het LTSP kernel pakket bevat het pxelinux.0 NBP, de benodigde configuratiebestanden voor het laden van een Linux kernel, plus een ramdisk image voor het opstarten.

Het werk op de volgende manier:

- Het PXE bootrom initialiseert de netwerkkaart en zend een dhcp-verzoek.
- De dhcp-server antwoordt met een ip-adres en de naam van het NBP dat geladen moet worden.
- De PXE bootrom haalt het NBP op, zet het in het geheugen, en voert het uit.
- Het NBP gebruikt tftp om het configuratie bestand van de server op te halen.
- Het configuratiebestand bevat de naam van de kernel, de naam van de opstart ramdisk, en de opties die aan de kernel moeten worden doorgegeven als die is geladen.

Hier is een voorbeeld van een pxe configuratiebestand:

```
prompt=0
label linux
    kernel bzImage-2.4.24-ltsp-4
    append init=/linuxrc rw root=/dev/ram0 initrd=initrd-2.4.24-ltsp-4.gz
```

- Het NBP gebruikt daarna tftp om de kernel en de opstart ramdisk op te halen.
 - Vervolgens wordt de controle overgedragen aan de Linux-kernel die opstart, initrd aankoppelt (mount), en doorgaat met het verder opstarten van het werkstation.
-

3.2. Opstarten met Etherboot

Etherboot is een software pakket voor het maken van ROM images die programmacode voor x86 computers downloaden over een Ethernet netwerk. Veel netwerkkaarten hebben een voorziening voor het plaatsen van een ROM chip. Etherboot is code die in zo'n chip gezet kan worden.

—Ken Yap

Etherboot is ook Open Source, vrijgegeven onder de GNU General Public License, versie 2 (GPL2)

Als men Etherboot wil gebruiken en over een netwerkkaart met bootrom beschikt, dan kan het nodig zijn dat de BIOS-configuratie moet worden ingesteld om op te starten van het netwerk ('Boot from LAN') in plaats van de harde schijf of de floppy.

Als er nog geen Etherboot bootrom is kan men ofwel een bootrom maken, ofwel een floppy maken met een Etherboot image in de bootsector.

Etherboot ondersteund meer dan 200 typen netwerkkaarten, en regelmatig worden er typen toegevoegd. Of er nu een bootrom gemaakt wordt of dat er een floppy wordt gebruikt, het is in ieder geval nodig het juiste type netwerkkaart vast te stellen.

3.2.1. Het kiezen van een Etherboot driver voor een ISA netwerkkaart

Voor oudere ISA kaarten is het niet zo belangrijk om het type exact vast te stellen. Meestal gaat het om ne2000 of 3Com 3c509 kaarten. Men hoeft dan alleen maar de goede Etherboot driver uit te kiezen die overeenkomt met het type kabel dwz 10 base-2 (Coax) of 10 base-T (Twisted pair)

3.2.2. Het kiezen van een Etherboot driver voor een PCI netwerkkaart

Voor PCI is het belangrijk dat de goed driver gekozen wordt met het juiste PCI Vendor en Device ID.

Met wat geluk is het model kaart bekend. Bijvoorbeeld omdat het modelnummer dat op de kaart geprint staat overeenkomt met de beschrijving van de Etherboot modules. Meestal echter moet het PCI ID nummer opgezocht worden.

Als het werkstation een floppy station heeft kan worden opgestart met tomsrtbt (Tom's Root Boot) floppy. Als het werkstation een cdrom station heeft kan met een Knoppix cd worden opgestart. Als op geen enkele manier Linux opgestart kan worden zit er niets anders op dan de kaart over te brengen naar een computer die wel Linux kan booten.

Wanneer eenmaal Linux opgestart is kan het commando **lspci** gebruikt worden met de optie **-n**.

```
[root@jamlap root]# lspci -n
0000:00:00.0 Class 0600: 8086:7190 (rev 03)
0000:00:01.0 Class 0604: 8086:7191 (rev 03)
0000:00:03.0 Class 0607: 104c:ac1c (rev 01)
0000:00:03.1 Class 0607: 104c:ac1c (rev 01)
0000:00:07.0 Class 0680: 8086:7110 (rev 02)
0000:00:07.1 Class 0101: 8086:7111 (rev 01)
0000:00:07.2 Class 0c03: 8086:7112 (rev 01)
0000:00:07.3 Class 0680: 8086:7113 (rev 03)
```

```
0000:00:08.0 Class 0401: 125d:1978 (rev 10)
0000:01:00.0 Class 0300: 1002:4c4d (rev 64)
0000:06:00.0 Class 0200: 8086:1229 (rev 09)
```

In het bovenstaande voorbeeld is voor iedere PCI kaart een informatieregel zichtbaar. Alleen de **Class 0200** regels zijn van belang. Nogmaals het commando intypen maar nu met alleen weergave van de Ethernet kaarten maakt de lijst wat overzichtelijker:

```
[root@jamlap root]# lspci -n | grep "Class 0200"
0000:06:00.0 Class 0200: 8086:1229 (rev 09)
```

De PCI ID nummers zijn hier: **8086:1229**. Het eerste veld **8086** is het PCI Vendor ID, in dit geval de Intel Corporation. Het tweede veld **1229** is het PCI device ID. Dit vertelt ons het model van de netwerkkaart, in dit geval een EtherExpress 100.

3.2.3. Het maken van een opstartfloppy

Men kan ervoor kiezen het Etherboot pakket te downloaden en te configureren voor een specifiek type kaart, en vervolgens de broncode te compileren voor het benodigde type bootrom. Dit levert dan de gewenste bootrom image op die uiteindelijk in een EPROM of op een floppy geschreven kan worden.

Een veel eenvoudigere manier is om naar Marty Connor's www.Rom-O-Matic.net website te gaan.

Marty heeft prima werk verricht met het maken van een webpagina die het configuratie en compilatie proces van bootrom images voor zijn rekening neemt. Op de pagina kan aangegeven worden om welk type netwerkkaart het gaat, en welk soort image gemaakt moet worden. Ook is er nog een mogelijkheid om eventuele verdere opties te configureren. Na een druk op de knop 'Get ROM' wordt er vervolgens in een paar seconden een bootrom image gemaakt.

Kies bij de optie 'ROM output' voor 'Floppy Bootable ROM Image'. Dit plakt een 512 bytes lang opstart programma vooraan het image. Als het resultaat in de bootsector van een floppy wordt geschreven dan wordt tijdens het opstarten de etherboot image in het geheugen geladen waar het kan worden uitgevoerd.

Druk op 'Get ROM' en na een paar seconden verschijnt in de browser een pop-up venster met de vraag waar het bestand in de computer opgeslagen moet worden.

Als het image op de harde schijf staat moet hij nog naar een floppy worden gekopieerd. Doe een floppy in het station en typ het volgende commando in om de floppy te beschrijven:

```
dd if=Etherboot_Image of=/dev/fd0
```

3.2.4. Het maken van een bootrom

Om het Etherboot image in een EPROM weg te schrijven is een programmeer apparaat nodig. De prijs van dit soort apparatuur varieert van enige honderden tot duizenden dollars, afhankelijk van de mogelijkheden van het apparaat.

Het proces van het programmeren van een EPROM is volledig afhankelijk van de gebruikte apparatuur. Een beschrijving valt daarom buiten de doelstelling van dit document.

Hoofdstuk 4. Opstarten van het werkstation

Als de server en het werkstation goed zijn geconfigureerd is het gewoon een kwestie van floppy in het station en de computer aanzetten.

De Etherbootcode wordt van de floppy in het geheugen geladen, de netwerkkaart wordt gevonden en geïnitieerd, het dhcp-verzoek wordt verzonden en een antwoord komt terug waarna de kernel van de server wordt opgehaald. Als het werkstation door de kernel is geïnitieerd start X windows op en verschijnt een aanmeldvenster op het scherm vergelijkbaar met het voorbeeld hieronder:



Figuur 4–1. Aanmeldscherm

Op dit moment kan de gebruiker zich aanmelden. Het is belangrijk te onthouden dat men zich bij een server aanmeldt. Alle commando's worden op die server uitgevoerd en de uitvoer wordt op het werkstation weergegeven. Dit is de kracht van X windows.

Alle programma's die op de server staan kunnen worden gebruikt.

Hoofdstuk 5. Afdrukken

Los van de rol van GUI of tekstterminal kan het werkstation ook als printserver dienen. In totaal 3 printers kunnen worden verbonden met de parallelle of seriële poorten.

Dit gebeurt allemaal onmerkbaar voor de gebruikers van het werkstation. Zij zullen niet in de gaten hebben dat er dataverkeer door het werkstation naar de printers toe gaat.

5.1. Installeren gebruikerszijde

LTSP gebruikt het `lp_server` programma op het werkstation om de binnenkomende printopdrachten van de server door te sturen naar een printer aan een van de poorten van het werkstation.

Een printer op het werkstation kan worden aangesloten door een aantal configuratie items in het bestand `lts.conf` van een waarde te voorzien.

```
[ws001]
    PRINTER_0_DEVICE = /dev/lp0
    PRINTER_0_TYPE   = P
```

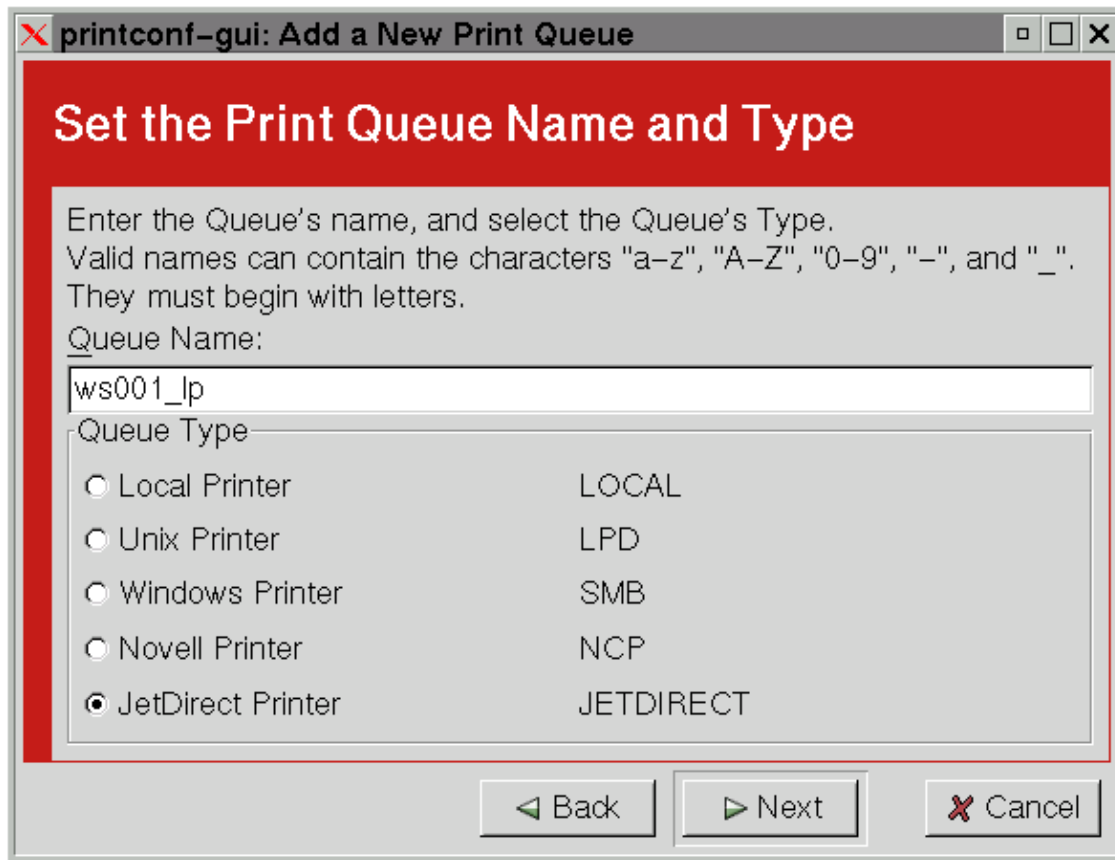
Bovenstaande items zorgen ervoor dat het `lp_server` programma op de achtergrond gedraaid wordt, wachtend op een printopdracht via poort 9100. De printdata worden doorgestuurd naar de parallelle poort `/dev/lp0`.

Er zijn meer opties. Kijk in de sectie over `lts.conf` voor meer informatie over het configureren van printers.

5.2. Installeren serverzijde

Het opzetten van een printer op de server is een kwestie van het definiëren van een printer wachtrij. Dit kan met het printerconfiguratie-programma op de server.

Op redhat 7.2 is zowel een GUI als een tekstgebaseerd programma aanwezig voor de printerconfiguratie. De GUI heet `printconf-gui`, de tekstversie heet `printconf-tui`. Oudere versies van Redhat hebben het programma `printtool`. Printtool bestaat ook in Redhat 7.2, maar dan roept het alsnog `printconf-gui` aan. Andere Linux distributies hebben hun eigen printer configuratie programma's.



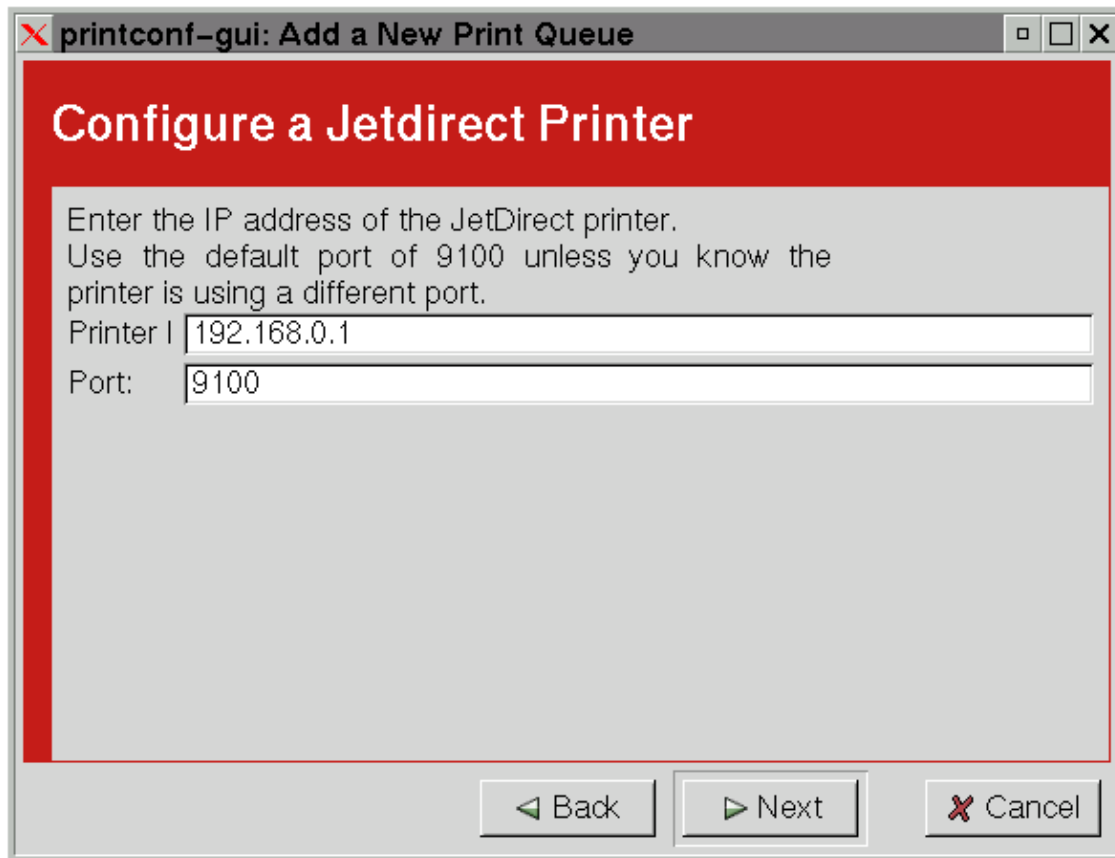
Figuur 5–1. Printconf–gui Toevoegen van een nieuwe printer

Indien het printer configuratieprogramma wordt opgestart moet eerst een nieuwe printer worden toegevoegd. Het lp-server programma op het werkstation emuleert een HP JetDirect print server. Het is nodig een **JetDirect** printer toe te voegen.

De printer heeft een naam voor de printer wachtrij nodig. De naam kan van alles zijn maar maak er een betekenisvolle naam van, met uitsluitend de onderstaande tekens:

- "a-z" kleine letters
- "A-Z" hoofdletters
- "0-9" getallen
- "-" minteken
- "_" underscore

De gekozen naam in het bovenstaande voorbeeld is **ws001_lp**. Door deze naam is het makkelijk te zien dat de printer is verbonden met **ws001**.



Figuur 5–2. Printconf–gui Detail informatie

Twee velden zijn noodzakelijk voor communicatie met de printer:

1. Het ip–adres of de hostnaam van het werkstation waarmee de printer verbonden is.
2. De tcp–poort waarop de **lp_server** daemon luistert.

De eerste printer die aan een werkstation vast zit krijgt tcp–poort **9100**. De tweede printer zit op poort **9101**, en de derde printer op **9102**.

Hoofdstuk 6. Schermscripts

Een van de faciliteiten die sinds versie 4.0 aan LTSP zijn toegevoegd zijn de zg '**Schermscripts**'. Deze scripts dienen voor het opstarten van de verschillende type sessies.

Er kunnen meerdere schermen voor een workstation worden gedefinieerd. Hierdoor kunnen meerdere sessies tegelijkertijd actief zijn. Het kunnen verschillende type sessies zijn of allemaal dezelfde. Men kan bijvoorbeeld het volgende opgeven:

```
SCREEN_01 = startx  
SCREEN_02 = shell
```

Dit start een X server op het eerstescherm en een shell op het tweede scherm. Wisselen tussen schermen kan met Ctrl-Alt-F1, om naar het eerste scherm te gaan, en Ctrl-Alt-F2, voor het tweede scherm.

Op deze manier kunnen 12 'schermscripts' worden opgegeven, maar de meeste mensen hebben aan 1 genoeg.

Beschikbare typen schermscripts:

startx

Dit script start de X server op met de optie `-query`, waardoor een XDMCP verzoek naar een displaymanager gestuurd wordt om een aanmeldvenster op het scherm te krijgen.

shell

Dit script start een shell op het workstation. Eigenlijk is dit script speciaal voor het oplossen van problemen op het workstation. Omdat de sessie op het workstation plaats vindt, en niet op de server, is hij niet erg handig voor het draaien van programma's.

telnet

Dit script start een telnet-sessie die verbonden is met de server. Het is een tekstsessie.

Standaard wordt de telnet-sessie met de LTSP server verbonden maar als men een verbinding met een andere server wil kan dat op de dezelfde regel aangegeven worden. Bijvoorbeeld:

```
SCREEN_01 = telnet server2.mydomain.com
```

Alle opties die het telnet programma kent kunnen op de regel worden opgegeven, maar zodra opties worden opgegeven moet altijd ook de server opgegeven worden waarmee verbonden moet worden.

rdesktop

Dit script start het rdesktop programma dat contact maakt met een Microsoft Windows server. Opties voor dit programma kunnen op dezelfde regel worden opgegeven, direct na de scriptnaam. Als bijvoorbeeld de server moet worden opgegeven kan dat op de volgende manier:

```
SCREEN_01 = rdesktop -f w2k.mydomain.com
```

Het bovenstaande voorbeeld start een desktop in volledige scherm. De gebruiker ziet een Windows aanmeldvenster, en hoeft maar een keer in te loggen. Dit is erg handig als alleen een windows aanmeldscherm nodig is, zonder Linux. De gebruikers weten niet eens dat ze in feite Linux gebruiken.

De schermscripts staan in de map `/opt/ltsp/i386/etc/screen.d`. Men kan eigen schermscripts maken en toevoegen aan deze map. Het handigste is om een van de scripts als voorbeeld te gebruiken.

Hoofdstuk 7. Problemen oplossen

Indien na het volgen van de aanwijzingen in de voorgaande hoofdstukken het werkstation niet opstart begint het proces van probleemoplossing.

Het eerste dat uitgezocht moet worden is hoever het opstartproces van het werkstation gekomen is.

7.1. Problemen oplossen van het Etherboot floppy image

Als van een floppy wordt opgestart zou ongeveer het volgende op het scherm te zien moeten zijn:

```
loaded ROM segment 0x0800 length 0x4000 reloc 0x9400
Etherboot 5.0.1 (GPL) Tagged ELF for [LANCE/PCI]
Found AMD Lance/PCI at 0x1000, ROM address 0x0000
Probing...[LANCE/PCI] PCnet/PCI-II 79C970A base 0x1000, addr 00:50:56:81:00:01
Searching for server (DHCP)...
<sleep>
```

Het bovenstaande voorbeeld laat zien wat verwacht kan worden bij het opstarten vanaf een floppy. Als deze boodschappen, die aangeven dat Etherboot opgestart is, niet te zien zijn kan het zijn dat de floppy defect is, of dat de image niet goed op de floppy weggeschreven is.

De volgende boodschap is waarschijnlijk zichtbaar het Etherboot image dat gemaakt is niet correct is voor de gebruikte netwerkkaart.

```
ROM segment 0x0800 length 0x8000 reloc 0x9400
Etherboot 5.0.2 (GPL) Tagged ELF for [Tulip]
Probing...[Tulip]No adapter found
<sleep>
<abort>
```

Als het opstartproces aankomt op het punt waarop het correcte MAC adres van de kaart wordt weergegeven dan is de floppy waarschijnlijk goed.

7.2. Problemen oplossen van DHCP

Als de netwerkkaart is geïntialiseerd zendt hij een dhcp-verzoek over het lokale netwerk, op zoek naar een dhcp-server.

Wanneer het werkstation een bruikbaar antwoord heeft gekregen van de dhcp-server wordt de netwerkkaart geconfigureerd. Of dit gelukt is valt af te leiden uit de ip-adres die in dat geval op het scherm wordt weergegeven. Hier is een voorbeeld hoe dat eruit zou moeten zien:

```
ROM segment 0x0800 length 0x4000 reloc 0x9400
Etherboot 5.0.1 (GPL) Tagged ELF for [LANCE/PCI]
Found AMD Lance/PCI at 0x1000, ROM address 0x0000
Probing...[LANCE/PCI] PCnet/PCI-II 79C970A base 0x1000, addr 00:50:56:81:00:01
Searching for server (DHCP)...
<sleep>
Me: 192.168.0.1, Server: 192.168.0.254, Gateway 192.168.0.254
```

Als de regel zichtbaar is die begint met 'Me:', gevolgd door een ip-adres, dan is duidelijk dat DHCP correct werkt. In dat geval kan worden doorgedaan met controle van TFTP

Als in plaats daarvan de volgende boodschap op het scherm verschijnt gevolgd door een heleboel <sleep> meldingen, dan is er iets mis. Het is normaal als er een of twee van deze <sleep> meldingen te zien zijn en pas daarna het antwoord van de dhcp-server.

```
Searching for server (DHCP)...
```

Uitzoeken wat er mis is kan soms moeilijk zijn, maar er zijn een paar zaken om aandacht aan te besteden.

7.2.1. Controleer de verbindingen

Is het werkstation fysiek verbonden met hetzelfde netwerk als waar de server aan vast zit?

Kijk, met het werkstation aan, of alle verbindingen-indicatorlichtjes op alle verbindingen aanstaan.

Als een directe verbinding (geen hub of switch) gelegd is tussen werkstation en server, wees er dan zeker van dat een cross-over kabel gebruikt wordt. Als een hub of switch gebruikt wordt, controleer dan of gewone 'straight-through' kabels worden gebruikt tussen zowel het werkstation en de hub, als tussen de hub en de server.

7.2.2. Draait dhcpd?

Er dient te worden vastgesteld of **dhcpd** draait op de server. We kunnen het antwoord op een aantal manieren vinden.

dhcpd draait normaal gesproken op de achtergrond en luistert op poort 67. Probeer het commando **netstat** om te kijken of er iets luistert op die poort:

```
netstat -an | grep ":67 "
```

Dit zou ongeveer het volgende moeten opleveren:

```
udp      0      0  0.0.0.0:67          0.0.0.0:*
```

De vierde kolom bevat het ip-adres en poortnummer, gescheiden door een dubbelepunt (':'). Een adres met allemaal nullen ('0.0.0.0') geeft aan dat er geluisterd wordt op alle interfaces (netwerkkarten). Dat wil zeggen, er is een **eth0** en een **eth1** interface, en **dhcpd** luistert op beide interfaces.

Dat netstat laat zien dat er iets luistert op udp poort 67 betekend nog niet dat het ook **dhcpd** is die luistert. Het zou ook **bootpd** kunnen zijn, al is dat onwaarschijnlijk omdat **bootpd** in de meeste Linux distributies niet meer meegeleverd wordt.

Om er zeker van te zijn dat inderdaad **dhcpd** draait kunnen we het commando **ps** gebruiken.

```
ps aux | grep dhcpd
```

Dit zou ongeveer het volgende moeten opleveren:

```
root 23814 0.0 0.3 1676 820 ?        S 15:13 0:00 /usr/sbin/dhcpd
root 23834 0.0 0.2 1552 600 pts/0  S 15:52 0:00 grep dhcp
```

De eerste regel laat zien dat **dhcpd** draait. De tweede regel laat alleen maar ons **grep** commando zien.

Als er helemaal geen regels zijn die aangeven dat dhcpd draait, controleer dan of **dhcpd** geconfigureerd is om onder runlevel 5 te draaien. Op Redhat systemen kan het commando **ntsysv** ingetypt worden. Naar beneden scrollend kan dan gecontroleerd worden of **dhcpd** geconfigureerd is om op te starten.

De **dhcpcd** daemon kan worden opgestart met het commando:

```
service dhcpcd start
```

Let goed op de schermuitvoer, er kunnen foutmeldingen in staan.

7.2.3. Dubbele controle van de dhcpcd-configuratie

Heeft het bestand `/etc/dhcpcd.conf` wel voor ieder werkstation aparte instellingen?

Controleer nogmaals de 'fixed-address' instelling in het configuratiebestand om er zeker van te zijn dat die exact overeenkomt met die van de kaart in het werkstation.

7.2.4. Blokkeert ipchains of iptables het verzoek?

7.2.4.1. Onderzoeken van ipchains

Draai het volgende command en kijk naar de uitvoer:

```
ipchains -L -v
```

Als dit ongeveer het volgende oplevert:

```
Chain input (policy ACCEPT: 229714 packets, 115477216 bytes):
Chain forward (policy ACCEPT: 10 packets, 1794 bytes):
Chain output (policy ACCEPT: 188978 packets, 66087385 bytes):
```

Dan is het niet ipchains dat in de weg zit.

7.2.4.2. Onderzoeken van iptables

Draai het volgende command en kijk wat het zegt:

```
iptables -L -v
```

Als dit ongeveer het volgende oplevert:

```
Chain INPUT (policy ACCEPT 18148 packets, 2623K bytes)
  pkts bytes target      prot opt in      out     source      destination
Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
  pkts bytes target      prot opt in      out     source      destination
Chain OUTPUT (policy ACCEPT 17721 packets, 2732K bytes)
  pkts bytes target      prot opt in      out     source      destination
```

Dan is het niet iptables dat in de weg zit.

7.2.5. Stuurt het werkstation dhcp-verzoeken?

Bekijk `/var/log/messages` terwijl het werkstation opstart. Dat kan met het volgende commando:

```
tail -f /var/log/messages
```

Dit 'volgt' het logbestand terwijl er nieuwe regels worden bijgeschreven.

```
server dhcpcd: DHCPDISCOVER from 00:50:56:81:00:01 via eth0
server dhcpcd: no free leases on subnet WORKSTATIONS
```

```
server dhcpd: DHCPDISCOVER from 00:50:56:81:00:01 via eth0
server dhcpd: no free leases on subnet WORKSTATIONS
```

Als dit meldingen zoals hierboven oplevert, met 'no free leases', dan betekent dat dat **dhcpd** wel draait maar dat het niks weet over het werkstation dat het ip-adres verzoek doet.

7.3. Problemen oplossen van TFTP

Etherboot gebruikt TFTP om een Linux kernel op te halen van de server. Het is een vrij eenvoudig protocol maar toch zijn er soms problemen om het aan de praat te krijgen.

Als een melding zoals hieronder zichtbaar is:

```
Loading 192.168.0.254:/lts/vmlinuz-2.4.24-ltsp-4.....
```

met puntjes die snel het scherm vullen, dan is dat een indicatie dat TFTP goed werkt en de kernel wordt opgehaald.

Als daarentegen geen puntjes verschijnen is er een probleem. Mogelijke problemen kunnen zijn:

7.3.1. tftpd draait niet

Als **tftpd** niet geconfigureerd is om op te starten dan zal het zeker geen verzoeken van het werkstation kunnen beantwoorden. Om te zien of het draait kan het commando **netstat** worden gebruikt:

```
[root@bigdog]# netstat -anp | grep ":69 "
udp        0      0 0.0.0.0:69          0.0.0.0:*           453/inetd
```

Als er geen uitvoer van dat commando verschijnt is tftp waarschijnlijk niet actief.

Er zijn twee gebruikelijke methoden voor het opstarten van tftpd. Dat zijn **inetd** en het nieuwere **xinetd**

inetd gebruikt een configuratie bestand genaamd `/etc/inetd.conf`. Zorg er voor dat in dat bestand tftpd NIET uitgeschakeld is met het commentaarteken '#'. De regel zou er zo uit moeten zien:

```
tftp dgram udp wait nobody /usr/sbin/tcpd /usr/sbin/in.tftpd -s /tftpboot
```

xinetd gebruikt een map met aparte configuratiebestanden. Een bestand voor iedere service. Als de server xinetd gebruikt dan is er een bestand `/etc/xinetd.d/tftp`. Hieronder staat een voorbeeld:

```
service tftp
{
  disable          = no
  socket_type      = dgram
  protocol         = udp
  wait             = yes
  user             = root
  server           = /usr/sbin/in.tftpd
  server_args      = -s /tftpboot
}
```

Wees er zeker van dat op de **disable** regel geen 'yes' staat.

7.3.2. Kernel staat niet op de plaats waar tftp hem verwacht

De kernel moet in een map staan waar de tftp daemon toegang heeft. Indien de optie '-s' wordt meegegeven aan de **tftpd** daemon dan moeten alle bestandspaden die het werkstation noemt relatief zijn ten opzichte van de map /tftpboot. Dus, als de instelling voor **filename** in het bestand /etc/dhcpd.conf ingesteld is op /lts/vmlinuz-2.4.24-ltsp-4, dan moet de kernelnaam zijn:
/tftpboot/lts/vmlinuz-2.4.24-ltsp-4

7.4. Problemen oplossen van het NFS root bestandssysteem

Er zijn verschillende oorzaken waardoor het root bestandssysteem niet geladen kan worden, zoals:

7.4.1. Geen init gevonden

Als de volgende foutmelding verschijnt:

```
Kernel panic: No init found. Try passing init= option to kernel.
```

wordt waarschijnlijk de foute map als root bestandssysteem opgegeven, of de map /opt/ltsp/i386 is leeg.

7.4.2. Server meldt error -13

Als de volgende foutmelding verschijnt:

```
Root-NFS: Server returned error -13 while mounting /opt/ltsp/i386
```

betekent dat de map /opt/ltsp/i386 niet vermeld staat in het bestand /etc/exports.

Bekijk het bestand /var/log/messages om te zien of aanwijzingen in staan. Een melding als:

```
Jul 20 00:28:39 bigdog rpc.mountd: refused mount request from ws004
for /opt/ltsp/i386 (/): no export entry
```

bevestigt ons vermoeden dat de instelling in /etc/exports niet goed is.

7.4.3. NFS Daemon problemen (portmap, nfsd & mountd)

Voor het oplossen van problemen kan NFS kan een complexe en moeilijke service zijn. Begrip van wat moet worden ingesteld en welke gereedschappen daarvoor nodig zijn, kan de diagnose aanzienlijk vereenvoudigen.

Er zijn drie daemons die op de server moeten draaien wil NFS goed werken: **portmap**, **nfsd** en **mountd**.

7.4.3.1. De Portmapper (portmap)

Als de volgende foutmelding verschijnt:

```
Looking up port of RPC 100003/2 on 192.168.0.254
portmap: server 192.168.0.254 not responding, timed out
Root-NFS: Unable to get nfsd port number from server, using default
Looking up port of RPC 100005/2 on 192.168.0.254
```

```
portmap: server 192.168.0.254 not responding, timed out
Root-NFS: Unable to get mountd port number from server, using default
mount: server 192.168.0.254 not responding, timed out
Root-NFS: Server returned error -5 while mounting /opt/ltsp/i386
VFS: unable to mount root fs via NFS, trying floppy.
VFS: Cannot open root device "nfs" or 02:00
Please append a correct "root=" boot option
Kernel panic: VFS: Unable to mount root fs on 02:00
```

wordt dat waarschijnlijk veroorzaakt doordat de **portmap** daemon niet draait. Controleren of **portmap** draait kan met het commando **ps**:

```
ps -e | grep portmap
```

Als de portmapper draait is de uitvoer:

```
30455 ?          00:00:00 portmap
```

Een ander controle is met het commando **netstat**. De portmapper gebruikt tcp- en udp-poort 111. Probeer:

```
netstat -an | grep ":111 "
```

Dit zou als uitvoer moeten geven:

```
tcp    0    0 0.0.0.0:111          0.0.0.0:*           LISTEN
udp    0    0 0.0.0.0:111          0.0.0.0:*
```

Als de uitvoer hier niet op lijkt dan draait de portmapper niet. De portmapper kan worden opgestart met:

```
/etc/rc.d/init.d/portmap start
```

Daarna moet er voor worden gezorgd worden dat de portmapper gestart wordt bij het opstarten van de server. Gebruik **ntsysv** om daarvoor te zorgen dat zo is.

7.4.3.2. De NFS en MOUNT daemons (nfsd & mountd)

NFS heeft 2 daemons die moeten draaien. **nfsd** en **mountd**. Beide worden gestart met het script `/etc/rc.d/init.d/nfs`.

Met het commando **ps** kan gecontroleerd worden of ze draaien.

```
ps -e | grep nfs
ps -e | grep mountd
```

Als dit aangeeft dat een van de twee of beide niet draaien moeten ze worden opgestart.

Normaal gesproken zou in dat geval het opstartscript moeten worden gedraaid met als argument **restart**. Echter, om een of andere reden herstart het script `/etc/rc.d/init.d/nfs` niet **nfsd** op die manier. Alleen **mountd** wordt herstart (bug?). Geef daarom de volgende commando's:

```
/etc/rc.d/init.d/nfs stop
/etc/rc.d/init.d/nfs start
```

Eventuele fouten in het commando **stop** zijn niet erg. Het commando **start** moet wel **OK** als uitvoer terug geven.

Als de daemons draaien maar NFS nog steeds niet werkt, dan moet gecontroleerd worden of zij zich hebben geregistreerd bij portmapper. Dit kan met het comando **rpcinfo**.

```
rpcinfo -p localhost
```

Zou ongeveer het volgende resultaat moeten opleveren:

```
program vers proto  port
```

```

100000 2 tcp 111 portmapper
100000 2 udp 111 portmapper
100003 2 udp 2049 nfs
100003 3 udp 2049 nfs
100021 1 udp 32771 nlockmgr
100021 3 udp 32771 nlockmgr
100021 4 udp 32771 nlockmgr
100005 1 udp 648 mountd
100005 1 tcp 651 mountd
100005 2 udp 648 mountd
100005 2 tcp 651 mountd
100005 3 udp 648 mountd
100005 3 tcp 651 mountd
100024 1 udp 750 status
100024 1 tcp 753 status

```

Dit geeft aan dat **nfs** (nfsd) en **mountd** beide draaien en zich hebben geregistreerd bij de portmapper.

7.5. Problemen oplossen met de Xserver

Oh Jee! Waarschijnlijk is het goed configureren van de X server het moeilijkste onderdeel van het installatie van een LTSP werkstation. Als een tamelijk nieuwe videokaart gebruikt wordt, met een relatief nieuwe monitor die een grote range aan frequenties en resoluties aan kan, dan is het vrij makkelijk. Als het dan niet werkt is de oorzaak meestal een verkeerde X server voor de betreffende kaart.

Als een X server niet werkt met een kaart is dat meestal wel duidelijk. De X server start niet of de weergave is niet goed.

Als een werkstation de X server gaat opstarten, wordt het script `startx` aangeroepen met de optie `-query`, wijzend naar een server een displaymanager zoals bv **XDM**, **GDM** or **KDM** draait

De X server wordt opgestart door het script `startx` dat op zijn beurt weer wordt opgestart door het programma **init**. Als het `startx` script faalt dan start **init** tot 10 keer toe het script opnieuw op. Uiteindelijk geeft **init** het op en blijft een foutmelding op het scherm achter.

Wachten op een X server die 10 keer niet opstart is behoorlijk irritant. Een manier op dat te voorkomen is door het werkstation in runlevel 3 te starten zodat de X sever niet automatisch gestart wordt. In plaats daar van verschijnt een **bash** prompt. Vanaf de bash prompt kan de X server worden opgestart met het volgende commando:

```
sh /tmp/start_ws
```

De X server zal dan proberen op te starten en als dat niet lukt, terugkeren naar de bash prompt met een leesbare foutmelding.

7.6. Problemen oplossen met de Displaymanager

De displaymanager is een daemon die op de LTSP-server draait en die wacht tot een X server van een werkstation contact met hem zoekt. Als het contact gelegd is zet de displaymanager een aanmeldvenster op het scherm waarmee de gebruiker in kan loggen bij de server.

De drie meest voorkomende displaymanagers zijn:

- XDM – Is er altijd al geweest. Zit standaard bij het X windowssysteem.

- GDM – De 'Gnome Display Manager'. Onderdeel van het Gnome-pakket.
- KDM – De 'KDE Display Manager'. Onderdeel van het KDE-pakket

In de meeste GNU/Linux distributies zitten alle drie de displaymanagers.

7.6.1. Grijs scherm met grote X cursor

Dit betekent dat de X server draait maar dat geen contact gelegd kon worden met de displaymanager. Mogelijke oorzaken zijn:

1. De displaymanager draait niet

Op recente versies van Redhat (7.0 en hoger) wordt de displaymanager gestart vanuit **init**. In het bestand `/etc/inittab` staat een regel die er als volgt uit ziet:

```
x:5:respawn:/etc/X11/prefdm -nodaemon
```

Het script **prefdm** stelt vast welke displaymanager gedraaid moet worden.

Wat de standaard displaymanager is hangt af van welke pakketten geïnstalleerd zijn. Als Gnome is geïnstalleerd is GDM de standaard displaymanager. Als Gnome niet is geïnstalleerd gaat `prefdm` kijken of KDE geïnstalleerd is. Is dat het geval dan is KDM de standaard displaymanager. Als KDE ook niet geïnstalleerd is wordt XDM de standaard displaymanager.

Met het commando **netstat** kan worden gecontroleerd of er een displaymanager draait. Typ op de server het volgende commando in:

```
netstat -ap | grep xdmcp
```

Dit zou als resultaat moeten geven dat er een proces luistert op de xdmcp poort (177).

```
udp      0      0  *:xdmcp          *: *              1493/gdm
```

Dit laat duidelijk zien dat **gdm** draait met een PID van 1493, en luistert op de de xdmcp poort.

Als je, zoals hierboven, een regel ziet die aangeeft dat er zeker een displaymanager luistert moet gecontroleerd worden of het werkstation wel zijn XDMCP verzoek naar de juiste server stuurt.

In het bestand `lts.conf` kan met een item worden opgegeven wat het ip-adres is van de server waar de displaymanager draait. Het item hoeft niet aanwezig te zijn maar als het er is moet het er als volgt uit zien:

```
XDM_SERVER = 192.168.0.254
```

Uiteraard zal het ip-adres per netwerk verschillen.

Als het item 'XDM_SERVER' niet aanwezig is wordt de waarde van 'SERVER' gebruikt. Als ook die ontbreekt wordt **192.168.0.254** gebruikt.

Hoe het ook wordt aangegeven, zorg er in ieder geval voor dat het opgegeven ip-adres echt van een server is waarop de displaymanager draait.

2. De displaymanager is ingesteld om externe verzoeken te negeren.

Als vastgesteld is dat de displaymanager draait, kan het zijn dat hij ingesteld is om XDMCP verzoeken van buiten te negeren. Dat kan in de configuratie bestanden van de betreffende displaymanager worden gecontroleerd.

◆ XDM

In Redhat wordt standaard de mogelijk voor extern aanmelden uitgeschakeld. Het commando **ltsp_initialize** zou dit moeten inschakelen maar als dat niet werkt kan het bestand `/etc/X11/xdm/xdm-config` handmatig worden aangepast. Zoek naar een regel die er ongeveer als volgt uit ziet:

```
DisplayManager.requestPort: 0
```

Deze regel MOET uitgekommeneerd (met een kommentaarteken aan het begin van de regel uitgeschakeld) zijn om XDM te laten luisteren op poort 177.

Er is nog een belangrijk configuratiebestand dat nodig is om XDM te laten luisteren naar externe aanmeldingsverzoeken. In het bestand `/etc/X11/xdm/Xaccess` MOET een regel staan die start met het '*' teken. De regel staat normaal al in het bestand maar Redhat kommenteed hem uit. Het script **ltsp_initialize** schakelt de regel weer in maar als XDM toch niet werkt moet dit bestand worden gecontroleerd. Een geldige regel ziet er als volgt uit:

```
*          #any host can get a login window
```

◆ KDM

Nieuwe versies van KDM kennen het configuratiebestand **kdmrc**. Verschillende Linux distributies plaatsen het bestand op verschillende locaties. In redhat 7.2 is de bestandsnaam `/etc/kde/kdm/kdmrc`. In ander distributies kan met het commando **locate** het bestand worden getraceerd.

Het gedeelte waar extern aanmelden voor werkstations kan worden ingesteld is de sectie **[Xdmcp]**. Zorg ervoor dat **Enable = true**.

Oudere versies van KDM gebruiken de configuratiebestanden van XDM in `/etc/X11/xdm`.

◆ GDM

GDM gebruikt een andere groep configuratiebestanden. Ze staan in de map: `/etc/X11/gdm`

Het belangrijkste bestand is `gdm.conf`. Localiseer de sectie: **[xdmcp]**. Daar staat een instelling met de naam 'Enable' die op '1' of 'true' moet staan afhankelijk van de GDM versie. Hier is een voorbeeld:

```
[xdmcp]
Enable=true
HonorIndirect=0
MaxPending=4
MaxPendingIndirect=4
MaxSessions=16
MaxWait=30
```

```
MaxWaitIndirect=30  
Port=177
```

Let op de regel 'Enable=true'. Ouder GDM versies gebruiken '0' of '1' voor Disable en Enable. Nieuwere versies gebruiken 'False' en 'True'.

3. Als het zeker is dat de displaymanager draait en luistert naar verzoeken van externe werkstations, kan het ook nog zijn dat de displaymanager eenvoudigweg het ip-adres en de hostnaam niet aan elkaar kan koppelen. Het werkstation moet genoemd worden in het bestand `/etc/hosts`, of de DNS tabellen moeten worden aangepast.
-

Hoofdstuk 8. Kernels

Ten aanzien van de kernel versie die op een werkstation draait kunnen een aantal keuzes gemaakt worden. Er kan gekozen worden tussen de standaard kernels die voor download beschikbaar zijn, of voor het zelf compileren van de kernel. Bovendien kan gekozen worden voor het weergeven van een grafisch scherm met voortgangsbalk tijdens het opstarten door middel van de **Linux Progress Patch (LPP)**.

8.1. Beschikbare standaard LTSP kernels

Het kernel pakket van LTSP bevat twee kernels. Een met, en een zonder 'Linux Progress Patch'.

Voor beide kernels is de NFS swap patch al toegepast.

8.2. Zelf een Kernel maken

De LTSP kernel kan op twee manier worden geconfigureerd. De standaardmethode is met een zogenaamde 'Initial Ram Disk' (in het kort: **initrd**), een RAM opstartschijf. Het initrd is een klein bestandssysteem image dat in de kernel ingebouwd wordt. Als de kernel opstart wordt dit bestandssysteem als root bestandssysteem in het RAM geheugen geladen. Er zijn een aantal voordelen aan het gebruik van dit rdinit bestandssysteem. De eerste is dat netwerkdrivers als modules kunnen worden geladen tijdens het opstarten. Hierdoor kan met een standaard kernel een onbeperkt assortiment aan netwerkkaarten ondersteund worden. Een ander voordeel is dat de dhcp-client als gewone gebruiker ('in user space') in plaats van als root gedraaid kan worden. Hierdoor is er meer controle over de DHCP-instellingen door de server. Bijkomend voordeel is dat kernel iets kleiner wordt. De andere methode waarmee de kernel kan worden geconfigureerd is zonder **initrd**. In dat geval moet de specifieke driver voor de netwerkkaart statisch worden meegelinkt in de kernel. Ook moeten de opties 'IP-autoconfig' en 'Root filesystem on NFS' worden ingeschakeld voordat de kernel wordt gecompileerd. Het voordeel van het weglaten van initrd is een iets kleinere kernel en een iets versnelde opstartprocedure. Als het werkstation eenmaal draait is er vrijwel geen verschil meer tussen de twee methodes.

De standaard LTSP kernel bevat een Initial Ramdisk (initrd) die zorgt voor het detecteren van de netwerkkaart en voor het uitvoeren van het eerste dhcp-verzoek. Het doel was deze kernel zo klein mogelijk te maken. Daarom werd uClinux gebruikt als bibliotheek voor libc, en werd busybox gekozen voor de hulpprogramma's die voor het opstarten nodig zijn.

Voor het zelf maken van kernels kan het pakket 'ltsp-initrd-package' opgehaald worden. Dit pakket bevat de hiërarchie van het root bestandssysteem alsmede een script voor het aanmaken van het image.

8.2.1. Verkrijgen van de broncode voor de kernel

Als u een aangepaste kernel wilt maken dan is het meestal een goed idee om met een 'schone' versie van de broncode van **ftp.kernel.org** te beginnen. In de meeste distributies zijn namelijk patches toepast op de broncode die voor download beschikbaar is, waardoor de broncode niet meer exact gelijk is aan de officiële distributie.

Kies een broncodepakket en bewaar deze in de map `/usr/src`. De kernels worden opgeslagen in de map `/pub/linux/kernel` van de FTP-server `ftp.kernel.org`. U moet beslist een kernel uit de recente 2.4.x serie nemen in verband met de ondersteuning voor **devfs**.

Als u ook nog ondersteuning voor NFS swap en de Linux Progress Patch (LPP) wilt geven moet u er voor zorgen dat de benodigde patches bij de kernel broncode passen. Om het moment van dit schrijven is de nieuwste kernel versie die u kunt gebruiken versie 2.4.9.

Voor dit voorbeeld wordt kernel versie 2.4.9 gebruikt. De volledige padnaam voor deze kernel is:
`ftp://ftp.kernel.org/pub/linux/kernel/v2.4/linux-2.4.9.tar.bz2`

Plaats de broncode van de kernel in de map `/usr/src`. Let op! Als het bestand met `untar` wordt uitpakkt wordt de broncode in de map `linux` geplaatst. Als deze map al bestaat wordt de eventuele inhoud overschreven! Hernoem de map voordat het bestand wordt uitpakkt.

Het broncodepakket is gecompriemd met het commando **bzip2**. Het moet daarom eerst worden gedeprimeerd voordat het wordt uitpakkt met het commando **tar**. Het de-comprimeren en uitpakken kan met het volgende commando:

```
bunzip2 <linux-2.4.9.tar.bz2 | tar xf -
```

Na het uitpakken resulteert een map genaamd `linux`. Meestal hernoem ik nu de map naar iets met meer betekenis zoals:

```
mv linux linux-2.4.9
```

Ga naar de nieuwe map zodra deze hernoemd is:

```
cd linux-2.4.9
```

Meestal verander ik nog wat aan de `Makefile` voordat ik verder ga met het configureren van de nieuwe kernel. Aan het begin van het bestand staat een variabele genaamd **EXTRAVERSION**. Deze zet ik op `'ltsp-1'`, zodat het versienummer van de kernel wordt: `'2.4.9-ltsp-1'`. Dit is handig om de kernel later makkelijk te kunnen identificeren. Het begin van de `Makefile` zou er ongeveer als volgt uit moeten zien:

```
VERSION = 2
PATCHLEVEL = 4
SUBLEVEL = 9
EXTRAVERSION = -ltsp-1

KERNELRELEASE=$(VERSION) . $(PATCHLEVEL) . $(SUBLEVEL) $(EXTRAVERSION)
```

8.2.2. Kernel Patches

Na het uitpakken van de kernel is het tijd om de verschillende patches toe te passen. Voorbeelden hiervan zijn de NFS swap patch en de Linux Progress Patch. Deze Patches **MOETEN** worden uitgevoerd voordat de kernel verder wordt geconfigureerd.

8.2.2.1. NFS Swap patch

De NFS Swap patch maakt het mogelijk dat het werkstation een swap bestand gebruikt dat op een NFS server staat. Ofschoon het aangeraden wordt om een werkstation met voldoende geheugen uit te rusten, zodat een swapbestand helemaal niet nodig is, kan het met name in oudere machines moeilijk zijn het geheugen uit te breiden. Met behulp van een swapbestand op een NFS server kan een anders onbruikbare machine toch nog van dienst zijn.

Als de huidige map gelijk is aan `/usr/src/linux-2.4.9`, en de patch staat in `/usr/src`, dan kan de patch met volgende commando worden getest:

```
patch -p1 --dry-run <../linux-2.4.9-nfs-swap.diff
```

Hiermee wordt gekeken of de patch probleemloos kan worden uitgevoerd. Als er geen foutmeldingen zijn kan de patch worden toegepast zonder de optie **--dry-run**:

```
patch -p1 <../linux-2.4.9-nfs-swap.diff
```

8.2.2.2. Linux Progress Patch (LPP)

De Linux Progress Patch maakt het mogelijk een grafisch logo weer te geven tijdens het opstarten van het werkstation. De opstartmeldingen die gewoonlijk over het scherm heen lopen worden naar een ander tty scherm omgeleid. Speciale instructies in het opstartscript zorgen ervoor dat de voortgangsbalk aangeeft hoe ver het opstarten gevorderd is.

Net als de NFS swap patch kan de LPP patch worden getest met het commando:

```
patch -p1 --dry-run <../lpp-2.4.9
```

Als de test zonder problemen verloopt kan de patch worden toegepast met:

```
patch -p1 <../lpp-2.4.9
```

8.2.3. Het configureren van kernel opties

Het configureren van de opties voor een nieuwe kernel kan op verschillende manieren:

- make xconfig

Dit start het X-Windows programma voor het configureren van kernelopties.

- make menuconfig

Dit start het tekstschermbaseerde programma voor het configureren van de kernelopties.

- make config

Dit start het meest eenvoudige programma voor het instellen van kernelopties. De opties worden regel voor regel gepresenteerd.

8.2.3.1. Kernel configuratie met initrd

Voor het configureren van een kernel met initrd moeten de volgende opties worden ingesteld:

- File systems → /dev filesystem support

/dev file system support moet worden ingeschakeld. Dit gebeurt in de sectie 'File systems'. 'Automatically mount at boot' moet NIET worden ingeschakeld! Het aankoppelen van het bestandssysteem wordt verzorgd door het script /linuxrc.

- Block devices → RAM disk support

De kernel van een werkstation moet het gebruik van een RAM disk ondersteunen. Dit kan worden ingeschakeld in de sectie 'Block devices'.

- Block devices → Initial RAM disk (initrd) support

Dit moet worden ingeschakeld.

- Processor type and features → Processor family

U moet er zeker van zijn dat de nieuwe kernel kan draaien op de CPU van het werkstation. Dit kan worden ingesteld in de sectie 'Processor type and features'. Meestal moet ook de ondersteuning voor SMP worden uitgeschakeld, tenzij er meerdere CPU's in het werkstation zitten.

- File systems → Network file systems → NFS Client support

Het werkstation koppelt zijn root bestandssysteem via NFS, dus 'NFS client support' moet worden ingeschakeld.

Dit zijn de minimaal benodigde opties. Om de kernel zo klein mogelijk te maken is het raadzaam zoveel mogelijk ongebruikte opties in de kernel uit te schakelen.

8.2.3.2. Kernel configuratie zonder initrd

De configuratie van een kernel zonder initrd verschilt in een aantal opzichten van de configuratie van een kernel met initrd:

- Block devices → RAM disk support

De kernel van een LTSP werkstation moet het gebruik van een RAM disk ondersteunen.

- Block devices → Initial RAM disk (initrd) support

Dit moet worden uitgeschakeld.

- Networking options → IP:kernel level autoconfiguration

Dit moet worden ingeschakeld. Hierdoor zal de kernel automatisch de eth0 interface configureren met behulp van parameters die zijn meegegeven op de commando regel voor het opstarten van de kernel.

Het is niet nodig DHCP, BOOTP of RARP opties op te geven. De Etherboot bootrom heeft namelijk al een DHCP, BOOTP of RARP verzoek gedaan en de daarmee verkregen IP parameters worden via de commandoregel aan de kernel meegegeven. Dit zorgt ervoor dat de kernel niet zelf nog op ontdekkingstocht moet.

- Network device support → Ethernet (10 or 100Mbit)

Als geen initrd gebruikt wordt moet de netwerkkaart statisch worden meegelinkt in de kernel. Dit MOET echt omdat de netwerkkaart nodig is voor het root bestandssysteem wordt aangekoppeld. Dit is een van de belangrijkste verschillen tussen een kernel met en zonder initrd.

- File systems → /dev filesystem support

Vanaf LTSP versie 2.09pre2 is ondersteuning voor **devfs** nodig. Dit geldt zowel voor kernels met, als voor kernels zonder initrd.

- File systems → Automatically mount at boot

Als initrd niet wordt gebruikt moet het /dev bestandssysteem door de kernel zelf worden aangekoppeld. Zet deze optie daarom op 'Y'.

- File systems → Network file systems → NFS Client support

Het root bestandssysteem wordt door het werkstation via NFS aangekoppeld dus moet 'NFS client support' worden ingeschakeld.

8.2.4. Het compileren van de kernel

Om een en ander te vergemakkelijken is een kopie van het bestand `.config` opgenomen in het `ltsp_initrd_kit` pakket. Dit bestand kan worden gekopieerd naar de map `/usr/src/linux-2.4.9`.

Wanneer het instellen van de kernel opties klaar is kan worden begonnen met compileren. De volgende commando's moeten daarvoor worden uitgevoerd:

```
make dep
make clean
make bzImage
make modules
make modules_install
```

Deze commando's kunnen ook allemaal in dezelfde regel worden opgegeven:

```
make dep && make clean && make bzImage && make modules
&& make modules_install
```

Met de dubbele ampersand (&) wordt aangegeven dat het volgende commando alleen wordt uitgevoerd als het voorgaande commando succesvol is afgerond. Het tweede commando wordt dus pas uitgevoerd na het eerste, het derde commando wordt uitgevoerd na het tweede, enz.

Als het compileren klaar is de bestandsnaam van de nieuwe kernel `/usr/src/linux-2.4.9/arch/i386/boot/bzImage`.

8.2.5. Het 'merken' van de kernel ten behoeve van Etherboot

Een kernel die door etherboot moet worden gebruikt heeft nog een speciale behandeling nodig. Dit heet het 'merken' (in het Engels 'tagging') van de kernel. Hiermee wordt een stukje extra code aan de kernel vastgeplakt die wordt uitgevoerd voordat de controle aan de kernel wordt overgedragen. Het programma waarmee de kernel kan worden gemerkt heet '**mknbi-linux**'.

In het pakket `ltsp_initrd_kit` is shell script **buildk** opgenomen waar alle commando's in staan die nodig zijn voor het prepareren van een kernel ten behoeve van het opstarten van een netwerk.

Hoofdstuk 9. Its.conf items

Toen wij LTSP ontwierpen was een van de dingen waar we rekening mee moesten houden de diversiteit in hardware van de werkstations. In drie maanden tijd zouden de processor, video- en netwerkkaart van de nieuw aan het netwerk toe te voegen werkstations volledig verschillen van de eerste werkstations.

Daarom ontwierpen we een manier om de configuratie van ieder workstation op te geven. Het configuratiebestand heet `lts.conf`, en staat in de map `/opt/ltsp/i386/etc`.

De opmaak van `lts.conf` maakt het mogelijk zowel standaard als individuele workstation instellingen op te geven. Als alle werkstations identiek zijn, kan worden volstaan met de configuratie-instellingen in de sectie '[Default]'.

9.1. Its.conf voorbeeldbestand

Hier is een voorbeeld van een `lts.conf` bestand:

```
[Default]
SERVER           = 192.168.0.254
X_MOUSE_PROTOCOL = "PS/2"
X_MOUSE_DEVICE   = "/dev/psaux"
X_MOUSE_RESOLUTION = 400
X_MOUSE_BUTTONS  = 3
USE_XFS          = N
SCREEN_01        = startx

[ws001]
XSERVER          = auto
X_MOUSE_PROTOCOL = "Microsoft"
X_MOUSE_DEVICE   = "/dev/ttyS1"
X_MOUSE_RESOLUTION = 50
X_MOUSE_BUTTONS  = 3
X_MOUSE_BAUD     = 1200

[ws002]
XSERVER          = XF86_Mach64

[ws003]
SCREEN_01        = shell
```

9.2. Beschikbare Its.conf parameters

9.2.1. Algemene parameters

Commentaar

Commentaar begint met een hekje '#' en loopt door tot het einde van de regel.

LTSP_BASEDIR

Dit geeft aan waar het LTSP root bestandssysteem zich bevindt. De standaard waarde is `/opt/ltsp`.

SERVER

Dit is de server die gebruikt wordt voor de XDM_SERVER, TELNET_HOST, XFS_SERVER en de SYSLOG_HOST. Tenminste, als deze niet apart worden opgegeven. Als er een server is voor alle taken dan kan dat hier worden opgegeven. De andere serverparameters kunnen dan weggelaten worden. Als geen waarde wordt opgegeven dan wordt **192.168.0.254** gebruikt.

SYSLOG_HOST

Indien de log meldingen naar een andere dan de standaard server moeten worden gestuurd kan dat hier worden opgegeven. Standaard wordt 'SERVER' gebruikt.

NFS_SERVER

Het ip-adres van de NFS server indien de map /home moet worden aangekoppeld. Standaard wordt 'SERVER' gebruikt.

USE_NFS_SWAP

Zet deze parameter op **Y** om NFS swap te gebruiken. Standaardwaarde is **N**.

SWAPFILE_SIZE

Dit stelt de grootte van het swapbestand in. Standaard is **64m**.

SWAP_SERVER

Het swapbestand kan op iedere server staan die daarmee om kan gaan. Hier kan het adres van die server worden opgegeven. De standaardwaarde is gelijk aan de waarde die voor NFS_SERVER is ingesteld.

NFS_SWAPDIR

De naam van de map op de server die wordt geëxporteerd door NFS. De standaardwaarde is /var/opt/ltsp/swapfiles. Zorg er voor dat deze map ook in het bestand /etc/exports wordt genoemd.

TELNET_HOST

Indien het werkstation als tekstgebaseerde terminal wordt geconfigureerd, wordt deze waarde gebruikt als de host (of server) waarmee een telnet-sessie wordt gestart. Als deze parameter NIET wordt ingesteld dan wordt de waarde voor **SERVER** gebruikt.

DNS_SERVER

Wordt gebruikt om het bestand resolv.conf aan te maken.

SEARCH_DOMAIN

Wordt gebruikt om het bestand resolv.conf aan te maken.

SCREEN_01 thru SCREEN_12

Er kan een maximum van 12 schermscripts worden opgegeven per werkstation. Dit resulteert in de mogelijkheid voor 12 sessies op het werkstation. Wisselen tussen sessie kan met Ctrl-Alt-F1 tot en met Ctrl-Alt-F12.

```
SCREEN_01 = startx
SCREEN_02 = shell
```

Op dit moment zijn de volgende waarden mogelijk.

- ◆ startx
- ◆ telnet

- ◆ rdesktop
- ◆ shell

Kijk in de map `/opt/ltsp/i386/etc/screen.d` voor meer schermscripts, of schrijf een eigen schermscript en plaats die in deze map.

MODULE_01 thru MODULE_10

Met deze items kunnen maximaal 10 modules worden opgegeven die moeten worden geladen. De volledige commandoregel die voor insmod nodig zou zijn kan worden opgegeven. Bijvoorbeeld:

```
MODULE_01 = uart401.o
MODULE_02 = "sb.o io=0x220 irq=5 dma=1"
MODULE_03 = opl3.o
```

Indien de opgegeven parameter waarde een volledige padnaam is, wordt het commando **insmod** gebruikt om de module te laden. Anders wordt het commando **modprobe** gebruikt.

RAMDISK_SIZE

Als het werkstation opstart wordt een ramdisk aangemaakt die als `/tmp` wordt aangekoppeld. De grootte van die ramdisk kan met deze parameter worden ingesteld. De grootte wordt opgegeven in kilobytes (1024 bytes). Geef **RAMDISK_SIZE = 1024** op om een ramdisk van 1 megabyte te maken.

Als de grootte van de ramdisk hier wordt veranderd moet dat ook worden veranderd in de kernel. Dit kan in de kernel worden ingecompileerd, of, indien Etherboot of Netboot wordt gebruikt, de ramdisk size kan aan de kernel worden kenbaar gemaakt door deze op een aangepaste manier te 'merken' met `mknbi-linux`.

De standaard waarde voor deze parameter is 1024 (1 mb)

RCFILE_01 thru RCFILE_10

Het script `rc.local` kan extra RC scripts uitvoeren. Zet de scripts in de map `/etc/rc.d` en geef de naam van het script met deze parameter.

SOUND

Indien het LTSP Sound pakket (Geluidsondersteuning) is geïnstalleerd moet deze parameter op **Y** worden ingesteld. In dat geval wordt het script **rc.sound** uitgevoerd om de geluidskaart en de daemon op te starten. De standaardwaarde is **N**.

9.2.2. X-Windows parameters

XDM_SERVER

Indien XDM gebruik moet maken van een andere dan de standaard server kan dat met deze parameter worden ingesteld. Als de waarde NIET is ingesteld wordt de instelling van 'SERVER' gebruikt.

XSERVER

Dit definieert de X server die het werkstation gebruikt. Voor PCI en AGP videokaarten is deze instelling meestal niet nodig omdat het script `rc.local` die wel goed kan detecteren. De waarde kan op **auto** gezet worden om aan te geven dat automatische detectie gebruikt moet worden.

Voor ISA kaarten, of in andere gevallen waarin dat nodig is, kan de naam van de driver of X server worden opgegeven.

Als de waarde begint met **XF86_** dan wordt XFree86 3.3.6 gebruikt. Anders wordt X.org 6.7.0 gebruikt. De standaardwaarde voor deze parameter is **auto**.

X_MODE_0 tot en met X_MODE_2

Er kunnen maximaal 3 'modelines' worden opgegeven. Een modeline kan twee verschillende types invoer bevatten: een resolutie of een volledige modeline.

```
X_MODE_0 = 800x600
    of
X_MODE_0 = 800x600 60.75 800 864 928 1088 600 616 621 657 -HSync -VSync
```

Indien geen X_MODE_x parameter wordt ingesteld dan worden ingebouwde modelines gebruikt met de resoluties 1024x768, 800x600 and 640x480.

Indien een of meer X_MODE_x parameters worden opgegeven hebben die voorrang op de ingebouwde modelines.

X_MOUSE_PROTOCOL

Alle waarden die toepasbaar zijn voor het X.org protocol voor aanwijsapparaten kunnen hier ook worden gebruikt. Typische waarden zijn: "Microsoft" en "PS/2". De standaardwaarde is "**PS/2**".

X_MOUSE_DEVICE

Dit is de poort waar de muis op aangesloten is. Voor een seriële muis is dat een seriële poort zoals **/dev/ttyS0** of **/dev/ttyS1**. Voor een PS/2 muis is de waarde **/dev/psaux**. De standaardwaarde is **/dev/psaux**.

X_MOUSE_RESOLUTION

Dit is de waarde voor de 'Resolution' parameter in het bestand **XF86Config**. Voor een seriële muis is de gebruikelijke waarde **50**, voor een PS/2 muis **400**. De standaardwaarde is **400**.

X_BUTTONS

Deze parameter vertelt het systeem hoeveel knoppen de muis heeft. Wordt meestal op **2** of **3** gezet. De standaard waarde is **3**.

X_MOUSE_EMULATE3BTN

Deze parameter vertelt het systeem een 3 knops muis te emuleren door middel van het gelijktijdig indrukken van de linker en rechter muisknop. De standaard waard is **N**.

X_MOUSE_BAUD

De baudrate van de seriële muispoort. De standaard waarde is **1200**.

X_COLOR_DEPTH

Dit is het aantal bits voor de kleurdiepte. Mogelijke waarden zijn **8**, **15**, **16**, **24** en **32**. 8 bits geeft 256 kleuren, 16 geeft 65536 kleuren, 24 geeft 16 miljoen kleuren en 32 bits geeft 4.2 miljard kleuren! Niet alle X server accepteren alle waarden. De standaard waarde is **16**.

USE_XFS

Er is een keuze mogelijk tussen het gebruiken van de Z Font Server (XFS) of het het uitlezen van de fonts via het NFS bestandssysteem. Met de Fontserver kunnen in principe op een eenvoudige manier alle fonts op een plaats worden beheerd. Er zijn echter wat problemen als het aantal werkstations boven de 40 komt. De twee waarden voor deze parameter zijn **Y** en **N**. De standaardwaarde is **N**. Indien voor een fontserver gekozen wordt, kan met de parameter **XFS_SERVER** worden ingesteld

om welke computer het dan gaat.

XFS_SERVER

Indien een X Font Server wordt gebruikt om fonts aan te leveren kan dit worden opgegeven met deze parameter het ip-adres van de fontserver worden opgegeven. Als er niets wordt opgegeven dan wordt de waarde voor **SERVER** gebruikt.

X_HORZSYNC

Dit stelt de **HorizSync** parameter in van het X.org configuratiebestand. De standaardwaarde is "31-62".

X_VERTREFRESH

Dit stelt de **VertRefresh** parameter in van het X.org configuratiebestand. De standaardwaarde is "55-90".

XF86CONFIG_FILE

Het is ook mogelijk een volledig zelfgemaakt XF86Config bestand te gebruiken. Het bestand moet dan in de map `/opt/lts/i386/etc` worden gezet en de naam moet worden meegegeven met dit item. De keuze van de bestandsnaam is vrij. Een voorbeeld:

```
XF86CONFIG_FILE = XF86Config.ws004
```

9.2.3. Touch screen parameters

USE_TOUCH

Indien een zogenaamd 'touch screen' (aanraakscherm) wordt gebruikt kan dat worden aangegeven via dit item met de waarde **Y**. In dat geval kunnen specifieke instellingen van het touchscreen worden opgegeven met extra items. De standaardwaarde is **N**.

X_TOUCH_DEVICE

Een touch screen werkt als een muis en wordt meestal ook via een seriële poort met het werkstation worden verbonden. Met dit item kan opgegeven worden welke seriële poort zoals `/dev/ttyS0`. Er is geen standaardwaarde voor dit item.

X_TOUCH_MINX

Een calibratiewaarde voor een EloTouch touch screen. Standaard **433**.

X_TOUCH_MAXX

Een calibratiewaarde voor een EloTouch touch screen. Standaard **3588**.

X_TOUCH_MINY

Een calibratiewaarde voor een EloTouch touch screen. Standaard **569**.

X_TOUCH_MAXY

Een calibratiewaarde voor een EloTouch touch screen. Standaard **3526**.

X_TOUCH_UNDELAY

Een calibratiewaarde voor een EloTouch touch screen. Standaard **10**.

X_TOUCH_RPTDELAY

Een calibratiewaarde voor een EloTouch touch screen. Standaard **10**.

9.2.4. Parameters voor Lokale toepassingen

LOCAL_APPS

Als de werkstations in staat moeten zijn om lokaal toepassingen te draaien moet dit item op **Y** worden gezet. Zie het onderdeel 'Lokale Applicaties' in de LTSP handleiding. Daarna moeten nog een aantal acties worden ondernomen op de server. De standaard waarde is **N**.

NIS_DOMAIN

Voor het gebruik van LOCAL_APPS is een NIS server op het netwerk nodig. Met het item NIS_DOMAIN wordt de NIS domeinnaam opgegeven. Deze moet overeenkomen met de domeinnaam die op de server is gedefinieerd. Dit is **NIET** het zelfde als het internet domein!. De standaardwaarde is **ltsp**.

NIS_SERVER

Stel dit item in op het ip-adres van de NIS server als een broadcast naar deze sever ongewenst is.

9.2.5. Toetsenbord parameters

Alle bestanden voor toetsenbordondersteuning worden in de /opt/ltsp/i386 mapstructuur geplaatst waardoor het configureren van een toetsenbord neerkomt op het configureren van X.org. Hiervoor zijn een aantal parameters beschikbaar.

De waarden van de bovengenoemde parameters komen uit de documentatie van X.org. Wat voor X.org geldig is geldt ook voor deze parameters.

We zouden hier graag documentatie toevoegen over de waarden die nodig zijn voor het configureren van ieder type internationaal toetsenbord. Als u hiermee werkt en u heeft het toetsenbord goed geconfigureerd dan zouden de LTSP ontwikkelgroep uw feedback zeer op prijs stellen.

XkbTypes

De standaardwaarde hiervoor is het woord **default**'.

XkbCompat

De standaardwaarde hiervoor is het woord **'default'**.

XkbSymbols

De standaardwaarde hiervoor is **'us(pc101)'**.

XkbModel

De standaardwaarde hiervoor is **'pc101'**.

XkbLayout

De standaardwaarde hiervoor is **'us'**.

9.2.6. Printer configuratieparameters

Er kunnen maximaal 3 printers op het werkstation worden aangesloten. Met de volgende items in het bestand **lts.conf** kan een combinatie van seriële en parallelle printers worden ingesteld:

PRINTER_0_DEVICE

De device naam van de eerste printer. Het gaat om namen als: **/dev/lp0**, **/dev/ttyS0** of **/dev/ttyS1**.

PRINTER_0_TYPE

Het type printer. Geldige keuzes zijn: **'P'** voor Parallel, en **'S'** voor Serieel.

PRINTER_0_PORT

Het TCP/IP poortnummer. Standaard wordt de waarde **'9100'** gebruikt.

PRINTER_0_SPEED

Als het een seriële printer betreft kan hier de instelling voor baudrate opgegeven worden. De standaardwaarde is **'9600'**.

PRINTER_0_FLOWCTRL

Voor seriële printers kan hier de flowcontrol worden opgegeven. Ofwel **'S'** voor Software (XON/XOFF) flow control, ofwel **'H'** voor Hardware (CTS/RTS) flow control. Als geen van beide wordt opgegeven wordt standaard **'S'** gebruikt

PRINTER_0_PARITY

Voor seriële printers kan hier de pariteit worden ingesteld. De mogelijke keuzes zijn: **'E'**–Even, **'O'**–Odd of **'N'**–None. Indien niets wordt opgegeven wordt **'N'** gebruikt.

PRINTER_0_DATABITS

Voor seriële printers kan hier het aantal databits worden ingesteld. Mogelijke keuzes zijn: **'5'**, **'6'**, **'7'** en **'8'**. Indien niet opgegeven wordt **'8'** gebruikt.

PRINTER_1_DEVICE

Tweede printer device naam

PRINTER_1_TYPE

Tweede printer device type

PRINTER_1_PORT

Tweede printer device TCP/IP poort

PRINTER_1_SPEED

Tweede printer baud rate instelling (serieel)

PRINTER_1_FLOWCTRL

Tweede printer flow control instelling (serieel)

PRINTER_1_PARITY

Tweede printer pariteit instelling (serieel)

PRINTER_1_DATABITS

Second printer data bits (serial)

PRINTER_2_DEVICE

Derde printer device naam

PRINTER_2_TYPE

Derde printer device type

PRINTER_2_PORT

Derde printer device TCP/IP poort

PRINTER_2_SPEED

Derde printer baud rate instelling (serieel)

PRINTER_2_FLOWCTRL

Derde printer flow control instelling (serieel)

PRINTER_2_PARITY

Derde printer pariteit instelling (serieel)

PRINTER_2_DATABITS

Derde printer data bits instelling (serieel)

Hoofdstuk 10. Lokale toepassingen

Binnen een LTSP omgeving kunnen toepassingen zowel lokaal als op afstand, dwz op de server, draaien.

Het is verreweg het eenvoudigst om een LTSP omgeving in te richten waarbij de toepassingen op de server draaien. Dat betekent dat de toepassingen ook de CPU en het geheugen van de server tot hun beschikking hebben, en dat van het werkstation het scherm, toetsenbord en muis worden gebruikt.

Dit is een functionaliteit die eigen is aan X windows. Het werkstation werkt als een standaard 'X Windows terminal'.

Als een gebruiker een toepassing op het werkstation wil laten uitvoeren moet het werkstation een aantal dingen van de gebruiker weten zoals:

- gebruikersnummer (User id)
- De primaire gebruikersgroep waartoe de gebruiker behoort (group)
- De persoonlijke map /home van de gebruiker

LTSP maakt gebruik van Network Information Service, kortweg NIS (voorheen genaamd *Yellow Pages*) om de gebruikers- en groepsinformatie beschikbaar te maken aan werkstations.

10.1. Voordelen van het lokaal draaien van programma's

Er zijn voordelen aan het laten uitvoeren van toepassingen op het werkstation.

- Het vermindert de belasting op de server. Het lokaal uitvoeren van toepassingen kan, als het werkstation krachtig genoeg is, leiden tot betere prestaties. Vooral bij het uitvoeren van geheugenintensieve programma's zoals Netscape op een druk bezette server van een groot netwerk.
- Op hol geslagen programma's hebben geen invloed op andere gebruikers.
- Ondersteuning voor geluid is voor lokale programma's op het werkstation makkelijker te configureren.

10.2. Instellingen voor het gebruik van lokale toepassingen

Het gebruik van lokale toepassingen stelt echter ook bepaalde eisen.

- Hogere eisen aan het werkstation. Er is meer RAM en een krachtiger processor nodig. 64mb op het werkstation is een leuk begin.
- NIS- Om programma's op het werkstation te kunnen draaien moet men zich eerst bij het werkstation identificeren. Dat wil zeggen, het werkstation moet weten wie de gebruiker is. Dit vereist identificatie met behulp van een wachtwoord. Voor de identificatieprocedure over het netwerk is NIS gekozen.
- Er moeten een aantal extra mappen worden geëxporteerd voor het aankoppelen met NFS.
- Men moet genoeg nemen met het iets langzamer opstarten van toepassingen. Dit komt omdat ze via NFS ingelezen moeten worden, wat extra netwerkactiviteit veroorzaakt. Bovendien kan geen voordeel meer worden gehaald uit het gebruik van overlappende programmaonderdelen in het geheugen van de server, wat er op de server voor kan zorgen dat herhaaldelijk inladen van het programma minder tijd kost.

10.3. Server Configuratie voor Lokale toepassingen

10.3.1. Its.conf items

In het bestand `lts.conf` moeten een paar items worden ingesteld:

LOCAL_APPS

Dit item moet op **Y** worden gezet. Hierdoor wordt de volgende opeenvolging van acties uitgevoerd tijdens het opstarten van het werkstation:

1. De `/home` map op de server wordt aangekoppeld via NFS.
2. Het bestand `/var/yp/nicknames` wordt op het werkstation aangemaakt.
3. De **portmapper** op het werkstation wordt gestart.
4. **xinetd** wordt op het werkstation gestart.
5. Het bestand `/etc/yp.conf` wordt op het werkstation aangemaakt.
6. Het commando **domainname** wordt uitgevoerd met de waarde van **NIS_DOMAIN** uit het bestand `lts.conf`.
7. Het commando **ypbind** wordt op het werkstation uitgevoerd.

NIS_DOMAIN

Met NIS moeten alle computers die bij een specifieke NIS server willen horen onder hetzelfde NIS domein vallen (dit is niet hetzelfde als het DNS domein) Met **NIS_DOMAIN** kan worden aangegeven tot welk NIS domein een computer behoort.

NIS_SERVER

NIS kan zich direct bij een specifieke NIS-server aanmelden, of het kan een broadcast uitzenden om een server te zoeken. Als u een bepaalde server wilt kiezen kan het hostadres van die server met de parameter **NIS_SERVER** worden opgegeven.

10.3.2. Network Information Service – NIS

NIS is een service met clients en servers. Op de server draait een daemon die verzoeken van werkstations accepteert. Die daemon heet **ypserv**.

Op het werkstation draait een proces genaamd **ypbind**. Als het werkstation informatie over een bepaalde gebruiker wil opzoeken, bijvoorbeeld voor het controleren van een wachtwoord, wordt met **ypbind** een verbinding gemaakt met **ypserv** op de server.

Indien NIS al op het netwerk aanwezig is hoeft NIS niet op de LTSP server geïnstalleerd te worden. De items **NIS_DOMAIN** en **NIS_SERVER** in het bestand `lts.conf` kunnen in dat geval gebruikt worden om aan te sluiten bij de bestaande NIS configuratie.

Als NIS nog NIET op het netwerk aanwezig is moet de server worden geconfigureerd om **ypserv** op te starten.

Voor een volledige uiteenzetting over het opzetten van een NIS server wordt verwezen naar *The Linux NIS(YP)/NYS/NIS+HOWTO* op de LDP website. Onder aan dit document staat een lijst met verwijzingen voor verdere informatie.

10.4. Configuratie van toepassingen

Om een toepassing te kunnen laten draaien op een werkstation moeten alle componenten van die toepassing op een plaats worden gezet waar het werkstation ze kan vinden.

In oudere versies van LTSP (2.08 en eerder) werden een heleboel mappen door de server geëxporteerd en vervolgens door het werkstation aangekoppeld. Het ging dan om mappen als `/bin`, `/usr/bin`, `/lib` en `/usr`.

Het probleem met deze werkwijze is dat het alleen werkt als de server en het werkstation dezelfde architectuur hebben. Zelfs kleine verschillen als een Pentium II (i686) op de server en een klassieke Pentium (i586) op het werkstation zijn problematisch omdat de server waarschijnlijk alleen de i686 bibliotheken heeft en niet de i386, i486, en i586 bibliotheken.

Dit probleem is opgelost als de server over een volledige mapstructuur beschikt met alle programma's en bibliotheken voor een specifiek werkstation. Deze bestanden zijn onafhankelijk van de programma's en bibliotheken van de server zelf.

Voor het lokaal uitvoeren van een toepassing is het nodig alle benodigde programmaonderdelen in die mapstructuur te plaatsen. Een van de LTSP pakketten die voor download beschikbaar is, is het 'Local Netscape pakket'. Dit pakket installeert een aanzienlijke hoeveelheid files in de map `/opt/ltsp/i386/usr/local/netscape`. Het gaat om java classes, hulpbestanden, programmabestanden en scripts die in de genoemde map worden geïnstalleerd.

Netscape heeft geen extra systeembibliotheken nodig, dus er wordt niets in de map `/opt/ltsp/i386/lib` geplaatst. Een hoop toepassingen hebben echter wel extra bibliotheken nodig.

Hoe kan nu worden bepaald welke bibliotheken voor een toepassing nodig zijn? Welnu, daar kan het commando **ldd** voor worden gebruikt.

Stel dat we een bepaalde toepassing lokaal willen draaien. Als voorbeeld nemen we **gaim**. Dat is een AOL instant messaging client die het mogelijk maakt met andere mensen op een AOL forum te communiceren.

Eerst moet de locatie van het **gaim** programma bestand worden vastgesteld. Op een Redhat 7.2 systeem is dat in de map `/usr/bin`.

Als de locatie gevonden is kan **ldd** op de volgende manier worden aangeroepen:

```
[jam@server /]$ ldd /usr/bin/gaim
    libaudiofile.so.0 => /usr/lib/libaudiofile.so.0 (0x40033000)
    libm.so.6         => /lib/i686/libm.so.6 (0x40051000)
    libnsl.so.1       => /lib/libnsl.so.1 (0x40074000)
    libgnomeui.so.32  => /usr/lib/libgnomeui.so.32 (0x4008a000)
    libart_lgpl.so.2  => /usr/lib/libart_lgpl.so.2 (0x4015d000)
    libgdk_implicit.so.1 => /usr/lib/libgdk_implicit.so.1 (0x4016c000)
    libSM.so.6        => /usr/X11R6/lib/libSM.so.6 (0x40191000)
    libICE.so.6       => /usr/X11R6/lib/libICE.so.6 (0x4019a000)
    libgtk-1.2.so.0   => /usr/lib/libgtk-1.2.so.0 (0x401b1000)
    libdl.so.2        => /lib/libdl.so.2 (0x402df000)
    libgdk-1.2.so.0   => /usr/lib/libgdk-1.2.so.0 (0x402e3000)
    libgmodule-1.2.so.0 => /usr/lib/libgmodule-1.2.so.0 (0x40319000)
    libXi.so.6        => /usr/X11R6/lib/libXi.so.6 (0x4031d000)
    libXext.so.6      => /usr/X11R6/lib/libXext.so.6 (0x40325000)
```



```

libX11.so.6      => /usr/X11R6/lib/libX11.so.6 (0x40333000)
libgnome.so.32   => /usr/lib/libgnome.so.32 (0x40411000)
libgnomesupport.so.0 => /usr/lib/libgnomesupport.so.0 (0x40429000)
libesd.so.0      => /usr/lib/libesd.so.0 (0x4042e000)
libdb.so.2       => /usr/lib/libdb.so.2 (0x40436000)
libglib-1.2.so.0 => /usr/lib/libglib-1.2.so.0 (0x40444000)
libcrypt.so.1    => /lib/libcrypt.so.1 (0x40468000)
libc.so.6        => /lib/i686/libc.so.6 (0x40495000)
libz.so.1        => /usr/lib/libz.so.1 (0x405d1000)
/lib/ld-linux.so.2 => /lib/ld-linux.so.2 (0x40000000)

```

De uitdraai hierboven laat alle gedeelde bibliotheken zien die voor het draaien van het programma **gaim** nodig zijn.

De meeste programma's die gedeelde bibliotheken gebruiken vertrouwen op **ld-linux** om de benodigde bibliotheken te lokaliseren en te laden. Echter, sommige programma's laden hun bibliotheken handmatig, met de functieaanroep **dlopen()**. Voor die toepassingen werkt **ldd** niet. In dat geval moet het commando **strace** gebruikt worden. Dit laat alle **dlopen()** aanroepen zien, met de naam naam van de bibliotheken.

Als de lijst met bibliotheken uiteindelijk compleet is, moeten de betreffende bibliotheken gekopieerd worden naar de juiste plaats in de `/opt/ltsp/i386` mapstructuur.

10.5. Het starten van lokale toepassingen

Normaal gesproken draaien toepassingen in X Windows op dezelfde plaats als waar de windowmanager draait. Dat wil zeggen: als de windowmanager op de server draait dan draaien opgestarte toepassingen ook op de server. Hun schermuitvoer wordt doorgestuurd naar het werkstation.

Er is een truc nodig om het programma door het werkstation uit te laten voeren. Dit kan met het commando **rsh**.

Hier is een voorbeeld hoe het **gaim** programma op het werkstation kan worden opgestart:

```

HOST=`echo $DISPLAY | awk -F: '{ print $1 }'`
rsh ${HOST} /usr/bin/gaim -display ${DISPLAY}

```

Het bovenstaande voorbeeld kan in een **xterm** venster worden ingetoetst, of het kan in een shell script worden gezet dat via een icoon op de desktop kan worden aangeroepen.

Het opstarten van Netscape gebeurt op ongeveer dezelfde manier, maar er is een extra omgevingsvariabele nodig die eerst moet worden gedefinieerd.

```

HOST=`echo $DISPLAY | awk -F: '{ print $1 }'`
rsh ${HOST} MOZILLA_HOME=/usr/local/netscape \
    /usr/local/netscape/netscape -display ${DISPLAY}

```

Hoofdstuk 11. Configuratievoorbeelden

Bijna alle onderdelen van het workstation kunnen worden ingesteld met items in het bestand `lts.conf` dat meestal staat in de map `/opt/ltsp/i386/etc`.

11.1. Seriële Muis

Hier een voorbeeld van `lts.conf` items voor een tweeknops seriële muis:

```
X_MOUSE_PROTOCOL = "Microsoft"
X_MOUSE_DEVICE   = "/dev/ttyS0"
X_MOUSE_RESOLUTION = 400
X_MOUSE_BUTTONS  = 2
X_MOUSE_EMULATE3BTN = Y
```

11.2. PS/2 Wielmuis

Hier een voorbeeld van de `lts.conf` items voor een Intellimouse:

```
X_MOUSE_PROTOCOL = "IMPS/2"
X_MOUSE_DEVICE   = "/dev/psaux"
X_MOUSE_RESOLUTION = 400
X_MOUSE_BUTTONS  = 5
X_ZAxisMapping   = "4 5"
```

11.3. USB printer op een ThinkNic

Het ThinkNIC workstation heeft een USB poort die voor een lokale printer gebruikt kan worden. Hier is een voorbeeld van de daarvoor benodigde instellingen in `lts.conf`:

```
MODULE_01 = usb-ohci
MODULE_02 = printer
PRINTER_0_DEVICE = /dev/usb/lp0
PRINTER_0_TYPE = S
```

11.4. Geforceerd laden van de XFree86 3.3.6 Xserver op een workstation

Standaard wordt X.org 6.7.0 gebruikt. Indien voor een workstation de oudere XFree86 3.3.6 Server moet worden gebruikt moet die eerst apart worden geïnstalleerd. Vervolgens moet in het bestand `lts.conf` met een item de naam worden opgegeven. Hier een voorbeeld waar de **SVGA** X server wordt ingesteld:

```
XSERVER = XF86_SVGA
```

Hoofdstuk 12. Andere informatiebronnen

12.1. Online referenties

1. De LTSP home page

www.LTSP.org

2. Schijfeloze computers HOW-TO document voor Linux

www.linuxdoc.org/HOWTO/Diskless-HOWTO.html

3. Etherboot Home Page

etherboot.sourceforge.net

4. De Rom-O-Matic site

www.Rom-O-Matic.net

5. X.org Muisondersteuning

www.xfree86.org/current/mouse.html

6. XFree86-Video-Timing-HOWTO

www.linuxdoc.org/HOWTO/XFree86-Video-Timings-HOWTO.html

7. De Linux NIS(YP)/NYS/NIS+ HOWTO

www.linuxdoc.org/HOWTO/NIS-HOWTO.html

12.2. Gedrukte publicaties

- 1.

Managing NFS and NIS
Hal Stern
O'Reilly & Associates, Inc.
1991
ISBN 0-937175-75-7

- 2.

TCP/IP Illustrated, Volume 1
W. Richard Stevens
Addison-Wesley
1994
ISBN 0-201-63346-9

X Window System Administrator's Guide

Linda Mui and Eric Pearce

O'Reilly & Associates, Inc.

1993

ISBN 0-937175-83-8

(Volume 8 of the The Definitive Guides to the X Window System)