

# Das Räderwerk hinter dem Linux Terminalserver

Eine technische Einführung zu LTSP

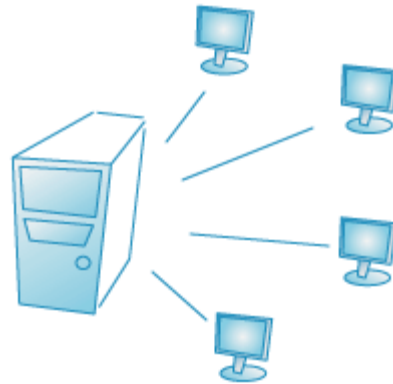


Das Monumentaluhrwerk im Zytglogge-Turm in Bern, das Kaspar Brunner 1530 vollendete, setzt noch heute Besucher und Fachleute in Erstaunen. Seine Robustheit, die Bauweise und seine Ausmasse sind einmalig. Alle Radteile sind geschmiedet, die Zähne einzeln eingesetzt und verkeilt oder verschraubt.

Ganz ähnliches Berühmtheit, wesentlich kleinere Abmessungen und annähernd so viele Zahnräder und Komplexität hat das Linux Terminal Server Projekt (kurz LTSP) in den letzten Jahren angenommen. Ein wesentliche Gemeinsamkeit zwischen den beiden eindrucksvollen Maschinerien ist es, dass Sie persönlich am Räderwerk hantieren dürfen und die Weiterentwicklung der einzelnen Bestandteile gesichert ist. Die Idee oder der Ursprung des Problems, welches zu diesem OpenSource Räderwerk führte, ist hier nachzulesen: <http://ltsp.sourceforge.net/shortstory.html>

Wie die einzelnen Räder und Rädchen zusammen wirken, möchte ich in den folgenden Sätzen kurz zusammen fassen.

# Hardware Voraussetzungen



## Client

- CPU: mind. 233MHz (für gute Performance 533 MHz)
- Memory: mind. 48MB (für gute Performance 128MB)
- Disk: keine notwendig
- Grafik: am PCI Bus und 2MB Memory
- Netzwerk: 100Mbps Karte ev. mit PXE oder BootROM Sockel



z.B. JakePC in der Grösse einer Unterputz-Steckdose.

## Server

- CPU: mind. P4 besser DualCore oder vergleichbare Leistung
- Memory: mind. 256MB für den Server und zusätzlich je 128MB pro Client
- Disk: empfehlenswert RAID 1 besser RAID 10 auf 2 SATA Disks
- Netzwerk: 2 x 100Mbps LAN (besser Giga LAN und entsprechend Giga Switch)

## Netzwerk

- Switch: mind. 100Mbps mit genügend Ports für alle Clients

## Boot-Methoden des Thin-Client

Es stehen eine Vielzahl von Möglichkeiten zum Booten eines Clients über das Netzwerk zur Verfügung. Nachfolgend finden Sie eine Auflistung unterteilt in die zwei Kategorien Boot-ROM und lokale Medien.

### Boot-ROM

Moderne Thin Clients haben einen Boot-ROM oder einen CF (Compact-Flash) Steckplatz. Es gibt auch für alte PCs CF-IDE Adapter. Hier können beliebige Systeme gespeichert werden. Kommerzielle Thin Clients haben meistens eigene abgespeckte Linux- oder Windows-Systeme mit einigen lokalen Programmen darauf. Bei LTSP ist dies nicht nötig: das System und sämtliche Programme werden vom Server bezogen.

### Etherboot

Abhängig von Ihrer Netzwerkkarte, kann darauf bereits ein Boot-ROM oder Steckplatz dafür vorhanden sein und Sie können theoretisch mit Hilfe eines EPROM-Programmiergerätes eigene ROMs brennen (siehe unten). Überprüfen Sie die Hardware-Dokumentation für die Netzwerkkarte in Ihrem Thin-Client. Schneller zum Ziel kommen Sie mit dem Kauf einer neuen z.B. PXE-fähigen Karte. Etherboot ist ein sehr beliebtes Open-Source-Boot-ROM-Projekt. Es enthält Treiber für viele gängige Netzwerkkarten und arbeitet sehr gut mit LTSP.

ROM-Image geeignet für das Booten von Diskette, CD-ROM, usw. sind unter der folgenden Adresse erhältlich <http://www.rom-o-matic.org>

### PXE

Ein Teil der "Wired for Management"-Spezifikation aus den späten 1990-er Jahren enthielt eine Spezifikation für eine Bootrom-Technologie bekannt als die *Pre-Boot Execution Environment* häufig abgekürzt als PXE.

Ein PXE-Bootrom kann höchstens eine 32 KB grosse Datei laden. Ein Linux-Kernel ist einiges grösser. Aus diesem Grund wird das PXE-Setup zum Laden eines "2. Etappe Bootloaders" pxelinux veranlasst, er ist klein genug, um geladen zu werden. Er weiß wie mit viel grösseren Dateien umzugehen und wie ein Linux-Kernel zu laden ist.

Obige Methoden funktionieren nur, wenn das BIOS des Rechners sie unterstützt. Bei ganz alten PCs ist dies meist nicht der Fall. Schauen Sie im BIOS, ob sie "Network-Boot" oder so etwas ähnliches finden und setzen es an erster Stelle in der Boot-Reihenfolge. Manchmal kann man das auch zur Boot-Zeit mit F12 machen.

### Lokale Medien

Wenn Ihre Netzwerkkarte im Thin-Client nicht über eine Boot-ROM verfügt, Sie

keinen Zugriff auf ein EPROM-Brenner haben oder die alten PCs keinen Netzwerk-Boot zulassen, müssen Sie noch nicht aufgeben wenn die alte Maschine über ein Disketten oder CD-ROM-Laufwerk verfügt. Wenn ja, können Sie lokale Medien zum Booten des Thin-Client verwenden.

### **Diskette**

Etherboot von einer Diskette ist ein ausgezeichnete Weg zum Booten eines LTSP Thin-Clients, der kein Boot-ROM hat. Etherboot wird in den Bootsektor der Diskette geladen und wirkt somit genau wie ein Boot-ROM. Der Boot-Code wird ausgeführt, die Netzwerkkarte wird initialisiert und der Kernel wird vom Server geladen.

### **Festplatte**

Die Festplatte kann mit LILO oder GRUB zum Laden des Linux-Kernels und initrd genutzt werden. Sie können auch das Etherboot Boot-ROM-Image von der Festplatte starten und es arbeitet wie ein Boot-ROM.

### **CD-ROM**

Eine bootfähige CD-ROM können entweder mit einem Linux-Kernel oder einem Etherboot Image geladen werden.

### **USB-Memory-Stick**

Genau wie eine CD-ROM, Disketten und Festplatten können Sie einen USB-Memory-Stick zu booten eines Etherboot-Modul einsetzen, vorausgesetzt das BIOS unterstützt USB-Sticks als Bootmedien.

## **Theorie zum LTSP Boot Vorgang**

Das Booten eines Thin-Clients beinhaltet mehrere Schritte. Es ist nützlich zu verstehen, was passiert – wie die Räder ineinander greifen und erleichtert das Lösen von Problemen, falls diese auftreten sollten. ;-)

Es werden vier grundlegende Dienste benötigt, um LTSP Thin-Clients zu booten. Es sind dies:

- DHCP (Dynamic Host Control Protocol)
- TFTP (Trivial File Transfer Protocol)
- NFS oder NBD (Network File System oder Network Block Device)
- SSH (Secure Shell)

### **Die LTSP chroot-Umgebung**

Um ein Computer als Thin-Client nutzen zu können, brauchen wir eine Mini-Version von GNU / Linux. Da Sie wahrscheinlich auf eine Festplatte verzichten möchten, kann diese Boot-Mini-Version von GNU / Linux über das Netzwerk bezogen werden.

Diese Mini-GNU / Linux-Installation muss irgendwo leben, und der beste Ort dafür ist auf einem Server.

Diese auf "effizient über das Netzwerk zu bootende" optimierte GNU / Linux-Installation, wird als eine chroot Umgebung bezeichnet. Sie können mehrere von ihnen, welche auf verschiedene CPU-Architekturen laufen anbieten. Sie existieren normalerweise unter /opt/ltsp/ auf dem Server in den Unterverzeichnissen für jede Architektur. Wenn Sie ältere PC's haben, benötigen Sie ein Verzeichnis /opt/ltsp/i386 auf dem Server.

Der Grund warum es eine chroot-Umgebung genannt wird besteht darin, dass es über den GNU / Linux-Befehl chroot (Change Root) erstellt wird, welcher den Installations Root Pfad nach /opt/ltsp/\* setzt. Dort ist eine abgespeckte Version der Distribution installiert. Das bedeutet, dass Sie mit chroot in die Umgebung wechseln um z.B. Updates vorzunehmen. In diesem Umfeld können Sie alle Ihre Werkzeuge wie gewohnt verwenden. Nach den Änderungen und dem Verlassen (mit exit) der chroot-Umgebung, muss zuerst ltsp-update-image ausgeführt werden, damit die Neuerungen aktiv werden.

## **Der Boot-Vorgang von einem Thin-Client unter Edubuntu**

1. Starten des Systems ist über verschiedene Boot-Methoden möglich, welche oben bereits erwähnt wurden. PXE ist weit verbreitet und äusserst solid. Die folgenden Steps beschreiben diese Variante.
2. Die oben erwähnte "Starthilfe" versucht von Ihrem DHCP Server (kann, aber muss nicht der LTSP-Server sein) eine IP zu ordern. Der DHCP Server gibt als Antwort zusätzlich einen Pfad zum Linux-Kernel. Die Konfiguration von dhcpd ist standardmässig unter /etc/ltsp/dhcpd.conf zu finden.
3. Via TFTP wird auf dem Server die Datei aus /var/lib/tftpboot/ltsp/i386/pxelinux.0 abgeholt und ausgeführt. In einem zweiten Schritt wird aus demselben Verzeichnis der Linux-Kernel von Ihrem LTSP-Server geladen und gestartet.
4. Sobald der Kernel in den Speicher geladen wurde, beginnt dessen Ausführung.
5. Der Kernel initialisiert das gesamte System und alle Peripheriegeräte, die er erkennen kann.
6. Während des Ladens des Kernels wird auch ein initramfs (Initial RAM File System) Abbild in den Speicher geladen.
7. Normalerweise wird an dieser Stelle, wenn der Kernel fertig mit booten ist, die Aufgabe an den neuen Launcher upstart übergeben, welcher das Starten des Computers handhabt. In diesem Fall jedoch beauftragen wir den Kernel stattdessen zum Laden eines kleinen Shell-Skripts namens /init, das in der Wurzel des initramfs lebt.
8. Das init-Skript beginnt mit der Montage von /proc und /sys, startet udev zum entdecken und initialisieren von Hardware, insbesondere der Netzwerkkarte, die notwendig für jeden weiteren Aspekt des Bootens ab

diesem Punkt ist. Ausserdem erzeugt das Skript eine kleine RAM-Disk, welche für eine lokale Speicherung von Daten benötigt wird z.B. zur Konfiguration der xorg.conf Datei.

9. Die Loopback-Netzwerk-Schnittstelle ist konfiguriert. Dies ist die Netzwerk-Schnittstelle und hat `127.0.0.1` als IP-Adresse.
10. Dann wird ein kleiner DHCP-Client namens `ipconfig` ausgeführt, welcher eine weitere Anfrage an den DHCP-Server abschickt. Diese Abfrage erhält Informationen aus der Datei `/etc/ltsp/dhcpd.conf` vom LTSP-Server, wie die NFS-Root-Server, Standard-Gateway und andere wichtige Parameter.
11. Wenn `ipconfig` eine Antwort vom Server bekommt, werden die Informationen zur Konfiguration der Ethernet-Schnittstelle verwendet und definiert welcher Server das Root-Dateisystem bereitstellt. Falls Sie IP Adressen später von Hand anpassen, sollten Sie nicht vergessen `sudo ltsp-update-sshkeys` auszuführen, damit Sie die Thin-Clients ordentlich starten können.
12. Bis zu diesem Punkt ist das Root-Dateisystem eine RAM-Disk. Nun wird das `init`-Skript ein neues Root-Dateisystem mounten entweder über NFS oder NBD. Im Falle von NBD das Image, das in der Regel unter `/opt/ltsp/images/i386.img` abgelegt ist. Wenn die Wurzel per NFS gemountet wird, dann das Verzeichnis, das vom Server typischerweise via `/opt/ltsp/i386` exportiert wird. Das neue Dateisystem `/` kann nun noch nicht geladen werden. Es muss zuvor in ein eigenes Verzeichnis gemountet werden. Dann wird durch `run-init`, das aktuelle Root-Dateisystem zum neuen Dateisystem gewapt. Wenn dies abgeschlossen ist, wird das Dateisystem auf `/` gemountet. An dieser Stelle werden alle Verzeichnisse, welche Schreibrechte für den rechtmässigen Start benötigen, wie z.B. `/tmp` oder `/var`, eingehängt.
13. Sobald die Montage des neuen Root-Dateisystem abgeschlossen ist, ist das `init` Shell-Skript beendet und das wahre `/sbin/init` Programm wird gestartet.
14. Das `init`-Programm liest das Verzeichnis `/etc/event.d` und beginnen mit der Einrichtung des Thin-Client-Umfelds. Von hier aus beginnt die Auslesung der Start-Kommandos aus dem Verzeichniss `/etc/rcS.d`.
15. Es führt den Befehl `S32ltsp-Client-Setup` aus, welcher die Konfiguration vieler Aspekte der Thin-Client-Umgebung, wie zum Beispiel die Überprüfung, ob für die lokalen Geräte spezifische Module geladen werden müssen, usw.
16. Als nächstes beginnt das `init`-Programm mit der Ausführung von Befehlen aus dem Verzeichnis `/etc/rc2.d`
17. Eines der Elemente im Verzeichnis `/etc/rc2.d` ist das Kommando `S20ltsp-Client-Core`, es wird ausgeführt während der Thin-Client gebootet wird.
18. Die Datei `/var/lib/tftpboot/ltsp/i386/lts.conf` wird, falls vorhanden sonst finden Sie unter `/opt/ltsp/i386/usr/share/doc/ltsp-client-core/examples/lts.conf` eine Beispieldatei, analysiert und alle Parameter aus dieser Datei, welche global oder sich auf diesen Thin-Client beziehen, werden als Environment-Variablen über das `S20ltsp-Client-Core` Skript

gesetzt.

19. Wenn Sound konfiguriert ist, wird der Daemon pulseaudio gestartet. Dieser Dienst ermöglicht Remote-Audio-Verbindungen vom Server um Audio Dateien auf dem Thin-Client abzuspielen.
20. Wenn der Thin-Client lokale Gerät-Unterstützung aktiviert hat, dann wird das Programm Itspfsd gestartet, damit der Server zum Lesen von Geräten wie Memory-Sticks oder CD-Rom am Thin-Client berechtigt ist.
21. Wenn Sie an diesem Punkt angelangt sind, wird jede Bildschirm-Session, welche in Its.conf definiert wurde, ausgeführt. Screen-Sessions starten alle virtuellen Bildschirme auf Ihrem Terminal. Dies sind die als Standard gesetzten virtuellen Bildschirme, welche alle gängigen GNU / Linux-Distributionen haben, d.h. von Alt-F1 bis Alt-F10.

Wenn nichts anderes in Its.conf angegeben ist, wird ein ldm Bildschirm-Skript in SCREEN\_07 gestartet. Die LTSP-Display Manager (LDM) ist der Standard-Login-Manager für LTSP. Es gibt keine eigene Konfigurationsdatei zu ldm. Alle individuellen Anpassungen müssen in Its.conf gemäss der Dokumentation

<http://doc.ubuntu.com/edubuntu/edubuntu/handbook/C/customizing-thin-client.html> hinterlegt werden.

22. Wenn SCREEN\_07 auf einen Wert von ldm oder startx stösst, dann wird das X Windows-System ins Leben gerufen, damit verfügen Sie über eine grafische Benutzeroberfläche. Standardmässig wird der Xorg-Server automatisch die Karte prüfen, erstellt eine default /etc/X11/xorg.conf Datei in der RAM-Disk und startet dann xorg mit einer benutzerdefinierten Konfiguration.
23. Der X-Server startet entweder einen verschlüsselten SSH-Tunnel zum Server, im Falle von ldm, oder sendet eine XDMCP-Abfrage an den LTSP-Server, im Falle von startx. Wie dem auch sei kurz darauf erscheint ein Login-Feld auf dem Terminal – Ihrem Bildschirm des Thin-Clients.
24. Nun kann sich der Benutzer einloggen und bekommt eine Session auf dem Server.

Bei der ersten Nutzung und vielleicht auch noch später verunsichert dies natürlich eine Menge Leute. Sie sitzen vor einem Thin-Client, aber arbeiten mit einer Session auf dem Server. Alle Befehle laufen direkt auf dem Server, nur die Ein- und Ausgabe wird auf dem Thin-Client vorgenommen.

## Weiterführende Informationen

### Bücher

Linux-Terminalserver, Galileo Computing, ISBN 3-89842-329-8

Technical Review [http://www.linux-magazin.de/technical\\_review/technical\\_review\\_06\\_server\\_based\\_computing](http://www.linux-magazin.de/technical_review/technical_review_06_server_based_computing)

## **Zeitschriften**

Linux Magazin [http://www.linux-magazin.de/heft\\_abo/ausgaben/2007/10](http://www.linux-magazin.de/heft_abo/ausgaben/2007/10)

## **Internet**

Edubuntu Installations Anleitung <https://wiki.edubuntu.org/HardyClassroomServer>

Edubuntu Handbuch <http://doc.ubuntu.com/edubuntu/edubuntu/handbook/C/>

Edubuntu FAQ <https://help.ubuntu.com/community/EdubuntuFAQ>

Its.conf Edubuntu

<http://doc.ubuntu.com/edubuntu/edubuntu/handbook/C/customizing-thin-client.html>

Its.conf Optionen <http://www.ltsp.org/twiki/bin/view/Ltsp/LtsConf>

NDB Info [http://www.linux-magazine.com/w3/issue/70/Network\\_Block\\_Devices.pdf](http://www.linux-magazine.com/w3/issue/70/Network_Block_Devices.pdf)

LTSP Demoanlage <http://www.openexpo.ch/openexpo-2008-bern/thin-client-demoanlage/>

Anwendungsbeispiel [http://www.espace.ch/artikel\\_329392.html](http://www.espace.ch/artikel_329392.html)