

# Wiki / PXE-Boot

**Dieser Artikel wurde für die folgenden Ubuntu-Versionen getestet:**

- **Ubuntu 14.04** Trusty Tahr
- **Ubuntu 12.04** Precise Pangolin

## Artikel für fortgeschrittene Anwender

Dieser Artikel erfordert mehr Erfahrung im Umgang mit Linux und ist daher nur für fortgeschrittene Benutzer gedacht.

## Zum Verständnis dieses Artikels sind folgende Seiten hilfreich:

1. **Installation von Programmen**
2. **Ein Terminal öffnen**
3. **Einen Editor benutzen**
4. **Umgang mit der Kommandozeile**
5. **Mit dem Systemverwaltermodus arbeiten (sudo)**
6. **Einrichten eines NFS-Servers**
7. **Konfiguration des ISC-DHCPD-Servers**

## Inhaltsverzeichnis

1. Einrichten des NFS-Servers
2. Der TFTP-Server
3. Einrichtung der PXE Systeme
4. PXE-Client einrichten
5. Links

**PXE** [[http://de.wikipedia.org/wiki/Preboot\\_Execution\\_Environment](http://de.wikipedia.org/wiki/Preboot_Execution_Environment)] steht für **P**reboot **eX**ecution **E**nvironment - was soviel bedeutet wie "Umgebung zum Ausführen von XYZ vor dem eigentlichen Bootvorgang". PXE bezeichnet dabei den Vorgang, um mit Hilfe einer PXE-fähigen Netzwerkkarte via DHCP eine Netzwerkkonfiguration (IP-Adresse, Adresse eines TFTP-Servers, ...) zu erhalten, und anschließend vom TFTP-Server einen Bootloader zu laden und auszuführen.

Auf diese Weise kann ein Betriebssystem geladen werden, ohne dass ein Datenträger (Festplatte, Diskette, CD etc.) auf dem Rechner (Client) benötigt wird. Alternativ kann man beispielsweise eine "Rettungsdistribution" laden, um sein System wieder herzustellen.

## Hinweis:

Die Grundkonfiguration ist in **PXE-Installation** beschrieben, dies ist nur eine Erweiterung auf andere Bootmedien!

## Einrichten des NFS-Servers

Damit Ubuntu-Live-Images oder ganze Distributionen funktionieren, muss ein NFS-Server aufgesetzt werden. Ansonsten könnten Änderungen nicht gesichert werden.

- **nfs-kernel-server** (*main*)

**Jetzt installieren**[apt://nfs-kernel-server] mit **apturl** [https://wiki.ubuntuusers.de/apturl/]

```
sudo apt-get install nfs-kernel-server
```

Nach der Installation muss (für jede freizugebende Distribution) dann eine NFS-Freigabe erstellt werden.

```
sudo mkdir /var/lib/tftp/Freigabe_einer_Distribution
```

In der Datei **/etc/exports** muss noch folgende Zeile angehängt werden. Diese gibt das Verzeichnis **/var/lib/tftp/Freigabe\_einer\_Distribution** und seine Unterverzeichnisse für alle IP-Adressen von 192.168.178.0 bis 192.168.178.255 mit Lese- und Schreibzugriff frei.

- Ubuntu 12.04 und höher:

1	<pre>/var/lib/tftp/Freigabe_einer_Distribution 192.168.178.0/255.255.255.0(rw,no_root_squash, sync, no_subtree_check)</pre>
---	---

- Ubuntu 10.04:

1	<pre>/var/lib/tftp/Freigabe_einer_Distribution 192.168.178.0/255.255.255.0(rw,no_root_squash, async)</pre>
---	--

Mit dem folgenden Befehl<sup>[3][5]</sup> wird dann die **/etc/exports** neu eingelesen:

- Ubuntu 14.04:

```
sudo service nfs-kernel-server reload
```

- Ubuntu 12.04:

```
sudo /etc/init.d/nfs-kernel-server reload
```

Weitere Informationen über die Konfigurationsmöglichkeiten des NFS-Servers sind **im NFS-Artikel**

[https://wiki.ubuntuusers.de/NFS/#ber-die-Konsole] zu finden.

## Der TFTP-Server

### Die Verzeichnisstruktur des TFTP-Servers

Nach der bisher beschriebenen Installation gibt es nun einige neue interessante Verzeichnisse:

- **/var/lib/tftp/** ist das Rootverzeichnis des tftp-Servers. Hier werden die Kernel bzw. die zu bootenden Daten hinterlegt. Des Weiteren **muss** dieses Verzeichnis das Konfigurations**verzeichnis pxelinux.cfg** beinhalten, es wird nicht automatisch erstellt und muss manuell angelegt werden.
- **/var/lib/tftpboot/pxelinux.cfg/** ist das Konfigurationsverzeichnis. Darin muss die Datei **default** enthalten sein. Auch werden hier alle clientspezifischen Konfigurationen abgelegt. Die Datei **default** wird nicht automatisch erstellt, sondern muss manuell angelegt werden, zur Syntax siehe unter "Beispielkonfigurationen".
- **/var/lib/tftp/Freigabe\_einer\_Distribution/** - dieses Verzeichnis ist das Rootverzeichnis des Clients in der jeweiligen gebooteten Distribution.

### Die Konfigurationsdateien

Es gibt zum einen die Datei mit dem Namen **default**. Diese Datei wird immer geladen, wenn keine client-spezifische

Konfigurationsdatei gefunden werden kann. Somit kommen wir zu den client-spezifischen Dateien. Diese Dateien sind auf identische Art und Weise zu erstellen <sup>[3]</sup> wie die default-Datei, haben jedoch einen anderen Dateinamen.

Diese Dateinamen, können entweder die MAC-Adresse der clientseitigen Netzwerkkarte mit vorangestellter 01 sein (01-XX-XX-XX-XX-XX-XX) oder die IP des Clients in hexadezimaler Form. Beispielweise entspricht die IP-Adresse 192.0.2.91 dem hexadezimalen Wert/Dateiname **C000025B**, wobei dieser Dateiname auch am Ende gekürzt werden kann, um eine Konfiguration auch für mehrere Clients zu haben. Dies wird Anhand des folgenden grauen Kastens aufgezeigt, denn in diesem Kasten wird die Reihenfolge der Konfigurationsdateiaufrufe angeführt. Wenn eine Konfigurationsdatei nicht vorhanden ist, dann wird die Nächste aufgerufen (in **/var/log/syslog** wird die Wahl auch vermerkt).

```
/var/lib/tftpboot/pxelinux.cfg/01-88-99-aa-bb-cc-dd
/var/lib/tftpboot/pxelinux.cfg/C000025B
/var/lib/tftpboot/pxelinux.cfg/C000025
/var/lib/tftpboot/pxelinux.cfg/C00002
/var/lib/tftpboot/pxelinux.cfg/C0000
/var/lib/tftpboot/pxelinux.cfg/C000
/var/lib/tftpboot/pxelinux.cfg/C00
/var/lib/tftpboot/pxelinux.cfg/C0
/var/lib/tftpboot/pxelinux.cfg/C
/var/lib/tftpboot/pxelinux.cfg/default
```

## Beispielkonfigurationen

An dieser Stelle ein paar Möglichkeiten, wie die Konfigurationsdateien unter verschiedenen Bedingungen aussehen könnten.

### Minimalkonfiguration

Bei dieser Konfigurationsdatei wird als Defaultwert das Label **memtest** gewählt. Da keine weiteren Parameter wie **PROMPT** oder **TIMEOUT** angegeben sind, wird direkt **memtest** und damit der Kernel **mt86plus** im Verzeichnis **memtest/** aufgerufen.

DEFAULT	memtest
LABEL	memtest
KERNEL	memtest/mt86plus

### Ohne Menü

Bei dieser Konfiguration kommen zusätzlich noch die Parameter **PROMPT** und **TIMEOUT** hinzu. Wenn **PROMPT** auf 1 gesetzt, erscheint eine Eingabezeile beim Bootvorgang, in der dann das zu bootende Label eingegeben werden kann. Bei dem Parameter **TIMEOUT** handelt es sich um die Zeitangabe, wie lange auf eine Eingabe gewartet werden soll, bevor das mit **DEFAULT** gesetzte **LABEL** aufgerufen wird. Die Einheit für **TIMEOUT** ist 1/10 Sekunde.

Beispiel:

Würde man **memtest** in die Befehlszeile eintippen, dann würde sofort der Kernel **mt86plus** gestartet werden. andernfalls würde die Ubuntu LiveDVD nach 10 Sekunden geladen werden.

Des Weiteren, sieht man an dieser Stelle das erste Mal den Parameter **APPEND**. Mit diesem Parameter werden dem Kernel zusätzliche Informationen übergeben, beispielsweise, dass der **Rootverzeichnis des Clients** im Verzeichnis "192.168.178.2:/var/lib/tftp/ubuntu-1404" zu finden ist.

```
DEFAULT ubuntu-1404
PROMPT 1
TIMEOUT 100

LABEL      ubuntu-1404
KERNEL     ubuntu-1404/casper/vmlinuz.efi
APPEND     root=/dev/nfs boot=casper netboot=nfs nfsroot=192.168.178.2:/var/lib/tftp
/ubunt-1404 initrd=ubuntu-1404/initrd.lz


LABEL      memtest
KERNEL     memtest/mt86plus
```

Man kann pxelinux auch dazu anweisen, eine Textdatei vor dem Prompt anzuzeigen. Die hierzu benötigte Direktive lautet `DISPLAY`, gefolgt von einem Leerzeichen und dem Namen der Textdatei. Ebenfalls lassen sich, wie von diversen Linux Installations-CDs (z.B. Debian) bekannt, auch die Funktionstasten belegen. Hierzu nutzt man den Namen der Funktionstaste als Direktive, sprich F1 bis F10, ebenfalls gefolgt von einem Leerzeichen und einem Dateinamen.

## Mit menu.c32

Wenn man es etwas komfortabler haben möchte, kann man mit Hilfe von **menu.c32** auch ein einfaches Menü erstellen. Hier jedoch zuvor die Datei `/usr/lib/syslinux/menu.c32` in das Rootverzeichnis des TFTP-Servers kopieren oder eine Verknüpfung erstellen, in unserem Falle nach: `/var/lib/tftp/`.

Das `^` markiert in dem Menü den "HotKey" zur Anwahl der entsprechenden Zeile. (Bei dem Tastaturlayout "Deutsch" ist diese Taste nicht belegt! Näheres siehe **Spracheinstellungen** [<https://wiki.ubuntuusers.de/Spracheinstellungen/#Tastatur>].)

An dieser Stelle ist in der Beispielkonfiguration nur ein sehr einfaches Menü angelegt worden. Auf **dieser Seite** [<http://syslinux.zytor.com/wiki/index.php/Comboot/menu.c32>]  lassen sich die vielfältigen Einstellmöglichkeiten bezüglich des Menüs nachlesen.

```
DEFAULT menu.c32
ALLOWOPTIONS 0
PROMPT 0
TIMEOUT 0

MENU TITLE Server PXE Boot Menu

LABEL      ubuntu-1404
MENU LABEL Ubuntu ^14.04 Desktop Live
KERNEL     ubuntu-1404/casper/vmlinuz.efi
APPEND     root=/dev/nfs boot=casper netboot=nfs nfsroot=192.168.178.2:/var/lib/tftp
/ubunt-1404 initrd=ubuntu-1404/initrd.lz

LABEL      memtest
MENU LABEL ^Memtest86+ v2.11
KERNEL     memtest/mt86plus
```

## Mit vesamenu.c32

Für die Variante mit **vesamenu.c32** liegen noch keine praktischen Erfahrungen vor. Hierbei sollte es dann, wenn man alles richtig macht, in etwa so aussehen, wie das Bootmenü einer beliebige LiveCD/DVD. Um es kurz zu sagen: das

Vesamenü ist sehr mächtig, jedoch auch komplizierter.

## Einrichtung der PXE Systeme

Die folgenden Konfigurationsbeispiele sind für menu.c32 Menüs gedacht.

### Lokales System booten

Um über PXE vom lokalen System zu booten, gibt es die Anweisung "LOCALBOOT 0". Hierfür sind die folgenden Zeilen in die Konfigurationsdatei einzufügen.

1	LABEL	localhdd1
2	MENU LABEL	Local ^HDD
3	LOCALBOOT	0

Das Booten des lokalen Systems über PXE mag zwar auf den ersten Blick unsinnig erscheinen, jedoch macht es Sinn, wenn der Client nicht die Möglichkeit hat, über einen Hotkey beim Booten über die Netzwerkkarte zu booten und man sich das Umstellen im BIOS ersparen will. Des Weiteren wäre es denkbar, dass man dem Benutzer nicht die Möglichkeit geben will, in das Bios zu gelangen bzw. die Bootreihenfolge zu ändern.

### Memtest86+

**Memtest86+** [<https://wiki.ubuntuusers.de/memtest/>] ist ein Programm, mit dem man den Arbeitsspeicher eines PCs auf Fehler überprüfen kann. Es eignet sich hervorragend zum Testen der Einstellungen des DHCP- sowie des TFTP-Servers, da man keine großen Kopieraktionen oder andere großartige Manipulationen an den PXE-Einstellungen vornehmen muss. Des Weiteren ist dieses Programm auf vielen Linux-CDs wie auch auf der Ubuntu Desktop CD zu finden.

Zuerst erstellt man im Verzeichnis **tftp** das Unterverzeichnis **memtest** und kopiert anschließend die Datei **mt86plus** von der Ubuntu-CD/DVD dort hinein. Die Datei **mt86plus** ist auf der Ubuntu CD im Verzeichnis **/install** zu finden.

```
sudo mkdir /var/lib/tftp/memtest && sudo cp /path-to-ubuntu-cd/install/mt86plus /var/lib/tftp/memtest/
```

Anschließend noch in der **default** Konfigurationsdatei unter **/var/lib/tftp/pxelinux.cfg/** folgende Zeilen anzufügen

1	LABEL	memtest
2	MENU LABEL	^Memtest86+ v2.11
3	KERNEL	memtest/mt86plus

```
sudo service dnsmasq restart
```

und nach einem Neustart von Dnsmasq, kann es dann auch schon mit dem ersten PXE-Boot losgehen.

### Ubuntu NetInstall

Für Ubuntu NetInstall wird auf den Artikel **PXE-Installation** [<https://wiki.ubuntuusers.de/PXE-Installation/#Bootimage-vorbereiten>] verwiesen, da an dieser Stelle schon eine gute Anleitung speziell für diesen Fall vorhanden ist.

### Ubuntu-Live-DVD

Eine Ubuntu-Live-DVD in Reichweite zu haben, ist immer gut, falls man mal keinen USB-Stick oder DVD zur

Verfügung hat. Kann eine Live-DVD als PXE-Boot helfen.

Da die Geschmäcker verschieden sind und auf der normalen Live-DVD nicht alle Programme und Sprachen vorhanden sind, sei hier zum Bauen eines eigenen Images auf das **Ubuntu Customization Kit** [https://wiki.ubuntuusers.de/Ubuntu\_Customization\_Kit/] verwiesen.

Verwendet werden kann jede Live-DVD von Ubuntu und all seinen Derivaten. Dieses muss lediglich **heruntergeladen** [https://wiki.ubuntuusers.de/Downloads/] werden und wird für die folgende Veranschaulichung in **/home/Benutzer/Downloads** abgelegt

Als nächstes muss dieses nun ausgelesen und in das TFTP-Server-Verzeichnis kopiert werden. Dies geschieht mit folgendem Befehl

```
sudo mount /home/Benutzer/Downloads/ubuntu.iso /mnt
sudo cp -R /mnt /var/lib/tftp/ubuntu-live
```

Danach muss die **/var/lib/tftp/pxelinux.cfg/default** angepasst und folgende Zeilen hinzugefügt werden:

```
LABEL          ubuntu-live
MEMU LABEL     ^Ubuntu-Live-DVD 14.04          # Nur wenn menu.c32 benutzt wird!
KERNEL         ubuntu-live/casper/vmlinuz.efi
APPEND         root=/dev/nfs boot=casper netboot=nfs nfsroot=192.168.178.2:/var/lib/tftp
               /ubuntu-live/ initrd=ubuntu-live/casper/initrd.lz quiet splash --
```

## Ubuntu Diskless

### Hinweis:

Dieser Abschnitt der Anleitung ist der Seite **Ubuntu.com**  entnommen und lediglich weitestgehend ins Deutsche übersetzt.

Es gibt diverse Möglichkeiten, eine komplette Ubuntuinstallation über das Netzwerk zu booten:

- debootstrap (um direkt eine Installation über das Netzwerk aufzusetzen)
- die Installation des Servers kopieren
- [K/X]Ubuntu auf dem Client von CD installieren. Nachdem Alles installiert ist und läuft, das Netzlaufwerk **/var/lib/tftp/NFS-Freigabe** mounten und das komplette System dorthin kopieren.

Wir werden an dieser Stelle auf Letzteres näher eingehen.

### Hinweis:

Die **Installation** kann selbstverständlich auch auf einem USB-Stick erfolgen, um das ursprüngliche System nicht zu beeinflussen oder zu beschädigen. Bei der Installation muss aber darauf geachtet werden, wo der Bootloader installiert wird, damit das ursprüngliche System nicht beeinträchtigt wird.

Die hier aufgeführten Befehle sind auf den Client-Rechner mit dem **bereits installierten und funktionsfähigem System** auszuführen, es sei denn, es ist explizit etwas Anderes erwähnt. Auf dem Client muss zusätzlich noch das Paket **nfs-common** installiert werden <sup>[1]</sup>, damit später der Client seine Daten über das Netzwerk beziehen kann.

Zuerst findet man mit **uname** [https://wiki.ubuntuusers.de/Shell/uname/] heraus, welcher Kernel aktiv ist. Die dazugehörige **initrd.img...** wird dann in das **/home**-Verzeichnis kopiert:

```
ls -ltr /boot/initrd*
```

```
-rw-r--r-- 1 root root 3,4M 2009-04-17 05:41 /boot/initrd.img-2.6.28-11-generic
-rw-r--r-- 1 root root 3,4M 2009-06-02 11:41 /boot/initrd.img-2.6.28-13-generic
```

```
sudo cp /boot/vmlinuz-2.6.28-13-generic /home/BENUTZERNAME
sudo cp /boot/initrd.img-2.6.28-13-generic /home/BENUTZERNAME
```

Jetzt muss eine neue **initrd.img**-Datei erstellt werden. Dafür muss das Bootflag in der Datei **initramfs.conf** von local nach nfs geändert werden. Die Konfigurationsdatei befindet sich im Verzeichnis **/etc/initramfs-tools/**.

```
1  #
2  # BOOT: [ local | nfs ]
3  #
4  # local - Boot off of local media (harddrive, USB stick).
5  #
6  # nfs - Boot using an NFS drive as the root of the drive.
7  #
8  #
9  BOOT=nfs
```

Mit dem untenstehenden Befehl wird dann die neue **initrd.img** erzeugt:

```
mkinitramfs -o /home/BENUTZERNAME/initrd.img-2.6.28-13-generic
```

Nun wird das **/nfsroot** Verzeichnis vom Server gemountet und das komplette System auf den Server kopiert:

```
mount -tnfs -onolock 192.168.2.2:/var/lib/tftp/NFS-Freigabe /mnt
cp -ax /. /mnt/ubuntu-diskless/.
cp -ax /dev/. /mnt/ubuntu-diskless/dev/.
```

Wenn ein Computer per PXE ein System bootet, hat er bereits eine Adresse über den DHCP-Server bezogen. Da wäre es fatal, wenn während des Bootvorgangs die Netzwerkkarte vom Betriebssystem neu konfiguriert werden würde. Aus diesem Grund muss die Datei **/nfsroot/ubuntu-diskless/etc/network/interfaces** dahingehend modifiziert werden, dass die Neukonfiguration verhindert wird. Dafür muss die Netzwerkkarte auf manual und nicht auf dhcp oder auto konfiguriert werden.

```
1  # The loopback network interface
2  auto lo
3  iface lo inet loopback
4
5  # The primary network interface, commented out for NFS root
6  #auto eth0
7  #iface eth0 inet dhcp
8  iface eth0 inet manual
```

Die Datei **/var/lib/tftp/NFS-Freigabe/ubuntu-diskless/etc/fstab** beinhaltet, wo welches Laufwerk / Partition beim Systemstart gemountet werden soll. Die **fstab** [https://wiki.ubuntuusers.de/fstab/] sollte in etwa wie die unteren Beispiele aussehen, je nachdem, ob schon tmpfs Dateisysteme eingebunden sind oder nicht.

### Hinweis:

Zu beachten ist, dass es keine swap-Partition gibt!

1	# <file system> <mount point> <type> <options> <dump> <pass>					
2	proc /proc proc defaults 0 0					
3	/dev/nfs / nfs defaults 1 1					
1	# /etc/fstab: static file system information.					
2	#					
3	# <file system> <mount point> <type> <options> <dump> <pass>					
4	proc /proc proc defaults 0 0					
5	/dev/nfs / nfs defaults 1 1					
6	none /tmp tmpfs defaults 0 0					
7	none /var/run tmpfs defaults 0 0					
8	none /var/lock tmpfs defaults 0 0					
9	none /var/tmp tmpfs defaults 0 0					
10	/dev/hdc /media/cdrom0 udf,iso9660 user,noauto 0 0					



Die dazugehörige Konfiguration in der `/var/lib/tftp/pxelinux.cfg/default` sieht dann wie folgt aus:

1	BENUTZERNAME
2	LABEL ubuntu-904-diskless
3	MENU LABEL Ubuntu 9.04 ^Diskless gup159xp
4	KERNEL ubuntu-diskless/vmlinuz-2.6.28-13-generic
5	APPEND netboot=nfs nfsroot=192.168.2.2:/var/lib/tftp/NFS-Freigabe/ubuntu-diskless rw initrd=ubuntu-diskless/initrd.img-2.6.28-13-generic

Nach erfolgter Erstellung der Konfiguration kann das Bootmedium am Client entfernt werden und dieser dann über PXE das System booten. Es ist darauf zu achten, dass nur der Client, auf dem das System installiert war, auch das System bootet. Hierfür eignet sich eine spezielle Konfigurationsdatei, die anhand der IP- oder der MAC-Adresse nur von diesem Client geladen werden kann (Siehe Abschnitt "Die **Konfigurationsdateien**).

## Clonezilla

Folgende Schritte sind der Reihe nach auszuführen, um Clonezilla über PXE booten zu können.

- Zuerst muss das **Clonezilla Live zip-File** [<http://clonezilla.org/downloads.php>]   besorgt werden. Dabei ist darauf zu achten, dass es sich um die Version 1.2.0-25 oder neuer handelt (je neuer desto besser):

### Hinweis:

Für das Klonen/Wiederherstellen von Ubuntu 9.10 sollte Clonezilla Live 1.2.3-24 oder neuer verwendet werden, da ab dieser Version die Wiederherstellung des Bootloaders GRUB2 unterstützt wird. In neueren Versionen von Clonezilla wurden die Dateien `vmlinuz1` und `initrd1.img` umbenannt in `vmlinuz` und `initrd.img`. Die unten genannten Pfade sind daher je nach Version ggf. anzupassen.


```
wget http://surfnet.dl.sourceforge.net/sourceforge/clonezilla/clonezilla-  
live-1.2.2-14.zip
```

- Nun wird die Datei nach `/nfsroot/clonzilla/` entpackt:




```
sudo unzip /PFAD/ZU/DATEI/clonezilla-live-1.2.2-14.zip -d /var/lib/tftp/clonezilla/
```

## Eintrag in der default Konfigurationsdatei

Zusätzlich zu der **Anleitung** [<http://clonezilla.org/clonezilla-live/livepxe.php>]  auf Clonzilla.org muss man in die APPEND-Zeile noch den nfsroot-Parameter angefügt. Ohne diesen Parameter bleibt das System beim Aufstarten von Clonezilla an der Stelle "Begin tying nfsmount ...." stehen und gibt als Fehler aus: "Unable to find a live file system on the network".

```
1  BENUTZERNAME
2  LABEL        clonezilla
3  MENU LABEL    ^Clonezilla Live 1.2.2-14
4  KERNEL        clonezilla/live/vmlinuzl
5  APPEND        initrd=clonezilla/live/initrd1.img boot=live union=aufs noswap noprompt
                vga=788 fetch=tftp://192.168.178.2/clonezilla/live/filesystem.squashfs
                nfsroot=192.168.178.2:/var/lib/tftp/clonezilla
```

### Experten-Info:


- Der fetch-Parameter unterstützt nicht nur tftp, sondern auch ftp und http. Wenn man ftp oder http verwenden möchte, dann muss tftp nach dem fetch-Parameter durch ftp oder http ersetzt werden und die Datei **filesystem.squashfs** in das entsprechende Verzeichnis des ftp- oder http-Servers kopiert oder verlinkt werden.
- In der APPEND-Zeile können Clonezilla noch weitere **Parameter**  zum Booten übergeben werden. Beispielsweise ocs\_live\_keymap=NONE ocs\_lang=en\_US.UTF-8, was dazu führt, dass automatisch das Standard-Tastaturlayout und Englisch als Sprache ausgewählt wird.
- Mit dem Einfügen von ocs\_live\_keymap="/usr/share/keymaps/i386/qwertz/de-latin1-nodeadkeys.kmap.gz" in die APPEND-Zeile wird das Deutsche Tastaturlayout gewählt.










## PXE-Client einrichten

Am Client an sich muss eigentlich so gut wie nichts eingerichtet werden. Es muss lediglich im BIOS eingestellt werden, dass man per PXE booten möchte. Hierzu muss u.U. das BootRAM der Netzwerkkarte zuerst aktiviert werden, bevor man die Option "Network" als (erstes) Bootmedium wählen kann.

Viele Systeme besitzen eine weitere Taste, die nicht ins BIOS, sondern in ein Auswahlmenü für das Bootmedium führt. Bei dieser Variante wird nur für den jeweiligen Bootvorgang von dem entsprechenden Medium gebootet. Bei einem aktuellen Dell Notebook ist die Bootmenütaste **F12**, bei HP **F9**, bei ASUS **F8** und bei IBM Thinkpads ebenfalls **F12**.

## Links

- Wikipedia:
  - **Dynamic Host Configuration Protocol (DHCP)** [[http://de.wikipedia.org/wiki/Dynamic\\_Host\\_Configuration\\_Protocol](http://de.wikipedia.org/wiki/Dynamic_Host_Configuration_Protocol)]
  - **Trivial File Transfer Protocol (TFTP)** [[http://de.wikipedia.org/wiki/Trivial\\_File\\_Transfer\\_Protocol](http://de.wikipedia.org/wiki/Trivial_File_Transfer_Protocol)]
- **PXE Netzwerk-Boot mit Ubuntu Client & Debian Server** [<http://lars-schenk.com/pxe-netzwerk-boot-mit-ubuntu-client-und-debian-server/60>]  - deutsche Anleitung von Lars Schenk, 09/2006
- SYSLinux / PXELinux:

- **Anleitung zum Erstellen eines Bootmenüs** [<http://syslinux.zytor.com/wiki/index.php/Menu>] 
- **Weiterführende Anleitung zum Erstellen eines Bootmenüs** [<http://syslinux.zytor.com/wiki/index.php/Comboot/menu.c32>] 
- **Beschreibung zu PXELINUX im SYSLINUX-Wiki** [<http://syslinux.zytor.com/wiki/index.php/PXELINUX>] 
- **Memtest:**
  - **Memtest\_86** [[http://de.wikipedia.org/wiki/Memtest\\_86](http://de.wikipedia.org/wiki/Memtest_86)]
  - **Memtest86** [<http://www.memtest86.com/>] 
  - **Memtest86+** [<http://www.memtest.org/>] 
- **Diskless Ubuntu Howto** [<https://help.ubuntu.com/community/DisklessUbuntuHowto>] 
- **CloneZilla:**
  - **Clonezilla** [<http://de.wikipedia.org/wiki/Clonezilla>]
  - **Clonezilla.org** [<http://clonezilla.org/>] 
  - **Beschreibung zum Erstellen einer PXE-Bootversion von Clonezilla auf Clonezilla.org** [<http://clonezilla.org/clonezilla-live/livepxe.php>] 
  - **Bootparameter für Clonezilla** [[http://clonezilla.org/clonezilla-live/doc/showcontent.php?topic=99\\_Misc#topic=99\\_Misc&section=00\\_live-initramfs-manual.doc](http://clonezilla.org/clonezilla-live/doc/showcontent.php?topic=99_Misc#topic=99_Misc&section=00_live-initramfs-manual.doc)] 

---

**Diese Revision** [<https://wiki.ubuntuusers.de/PXE-Boot/a/revision/870463/>] wurde am 25. Januar 2016 19:43 von **noisefloor** erstellt.

Die folgenden Schlagworte wurden dem Artikel zugewiesen: **Installation** [<https://wiki.ubuntuusers.de/wiki/tags/Installation/>], **Netzwerk** [<https://wiki.ubuntuusers.de/wiki/tags/Netzwerk/>], **Server** [<https://wiki.ubuntuusers.de/wiki/tags/Server/>], **System** [<https://wiki.ubuntuusers.de/wiki/tags/System/>]

Inhalte von ubuntuusers.de lizenziert unter Creative Commons, siehe <https://ubuntuusers.de/lizenz/>.