PXE - Netzwerkbooten mit Linux

Michael Gebetsroither

Technische Universität Graz

20. Mai 2006

Übersicht - Teil I

1. Vorwort
   - Grml
   - Wieso?
2. Vorraussetzungen
3. Vorbereitung
   - Auf grml
   - Auf Debian SID
4. Konfiguration
   - Netzwerk
   - Grub
   - Optionen
   - Erfolgreicher Start

Übersicht - Teil II

5. Was passiert im Hintergrund
   - Hardware
   - Bootloader
   - Initrd
6. Services
   - DHCP
   - TFTP
   - NFS
7. Bootloader
   - Bootparameter
   - Grub
   - Pxelinux

Teil I

Praxis

Was ist grml?

grml ist eine Linux Live-CD mit Hardware-Erkennung für x86-Systeme. grml ist für Benutzer von Texttools und Systemadministratoren ausgelegt und basiert auf Knoppix und Debian unstable. grml verwendet als zentrale Shell die zsh und beinhaltet über 2 GiB an Software.
grml startet standardmäßig nicht mit in einer grafischen X-Oberfläche, sondern bringt dafür 12 Konsolen, unter anderem mit GNU screen und htop sowie natürlich der z-shell (zsh).

Wieso grml-terminalserver?

Wieso wurde grml-terminalserver entwickelt:

- Thema Netzwerkboot war bis dato schlecht dokumentiert
- Zeitaufwändig
- Wenig Ressourcen im Netz
- Interresse
- Testumgebung für grml[1]

[1] Derzeit nur intern verfügbar

Was wird benötigt

- Eine grml-CD
- Einen Server[2]
- Clients[3]
- Ein schnelles Netzwerk

[2] Bei sehr vielen Clients sind 1GB RAM empfehlenswert
[3] Bootbar per PXE oder grub und z.b Diskette

grml-terminalserver auf grml

1. grml-CD booten
2. IP konfigurieren
3. grml-terminalserver starten

grml-terminalserver auf Debian SID

1. Installieren des grml-terminalservers[4]
2. Mounten der grml-CD oder des Images nach /cdrom
3. mount -t squashfs[5] /cdrom/GRML/GRML to /mnt/tmp
4. Anpassen folgender Variablen in /etc/grml/terminalserver/config:
   - MODULES_PATH_ROOT_DIFF_="/mnt/tmp"
   - MODULES_PATH_="/mnt/tmp/lib/modules"
   - KERNEL_VERSION_="2.6.16-grml"[6]
5. Start des grml-terminalserver

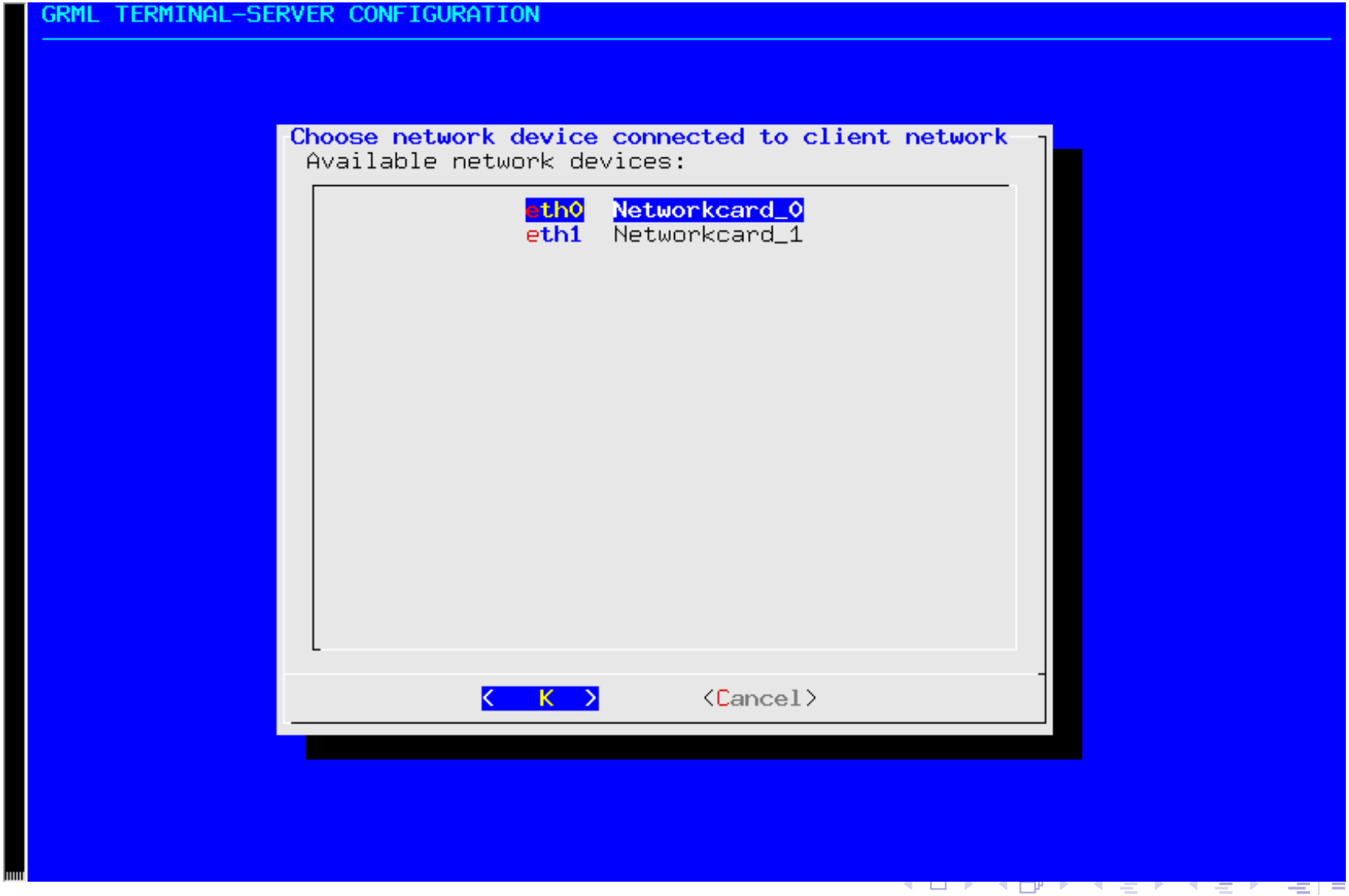
[4] Aktuellste Pakete `http://einsteinmg.dyndns.org/debian/`
[5] Das squashfs Kernelmodule wird benötigt
[6] Kernelversion der zu bootenden grml-CD

Vorteile

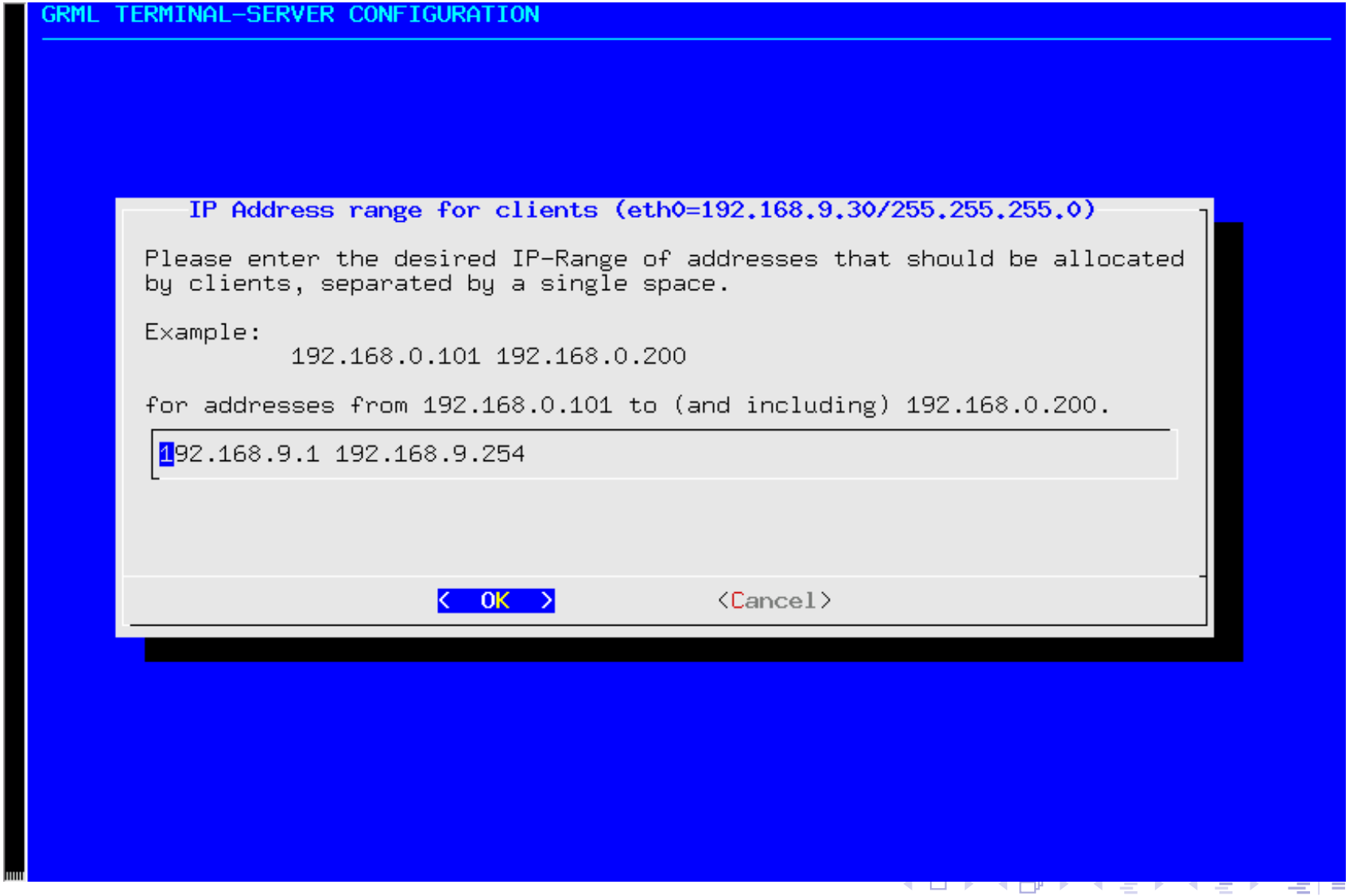
- Sehr einfache Konfiguration
- Kein langes Suchen nach Dokumentation
- Kein umständliches extrahieren von benötigten Dateien
- Sehr gute Testplattform[7]

[7] CD-RW Ausschuss ist ansonsten unerträglich hoch

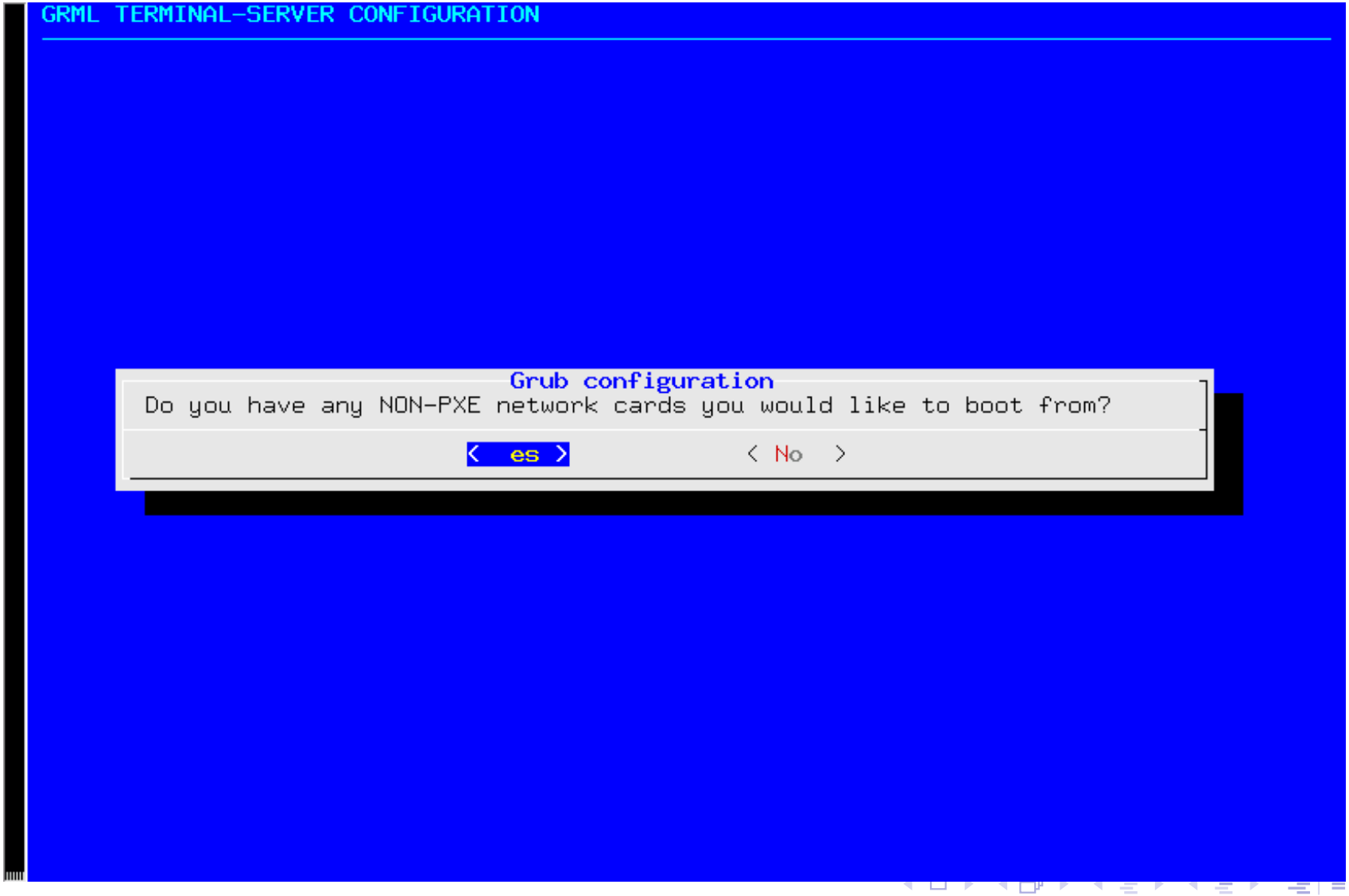
Auswahl der Netzwerkkarte



Konfigurieren des Addressbereiches



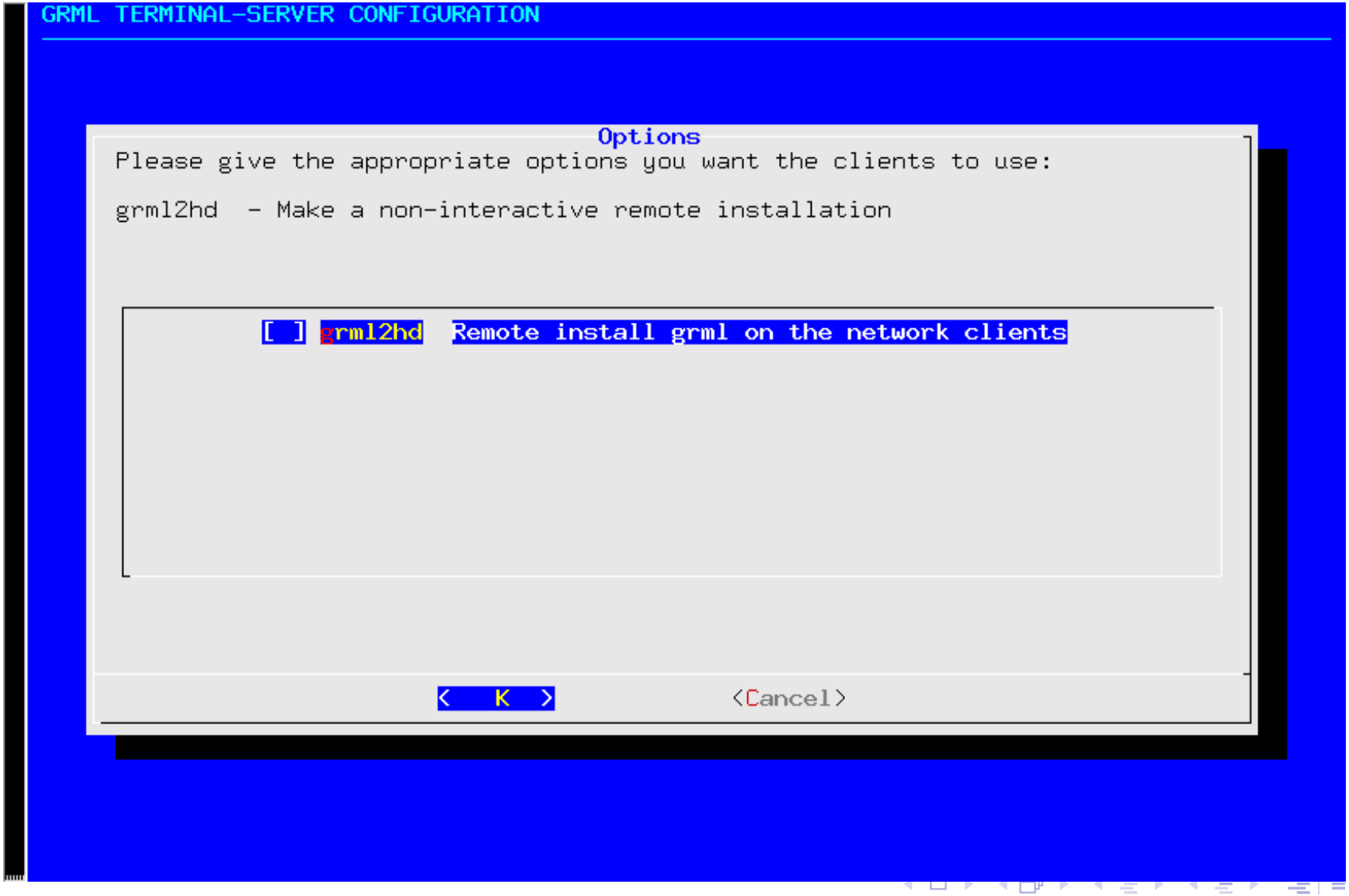
Soll grub zusätzlich verwendet werden



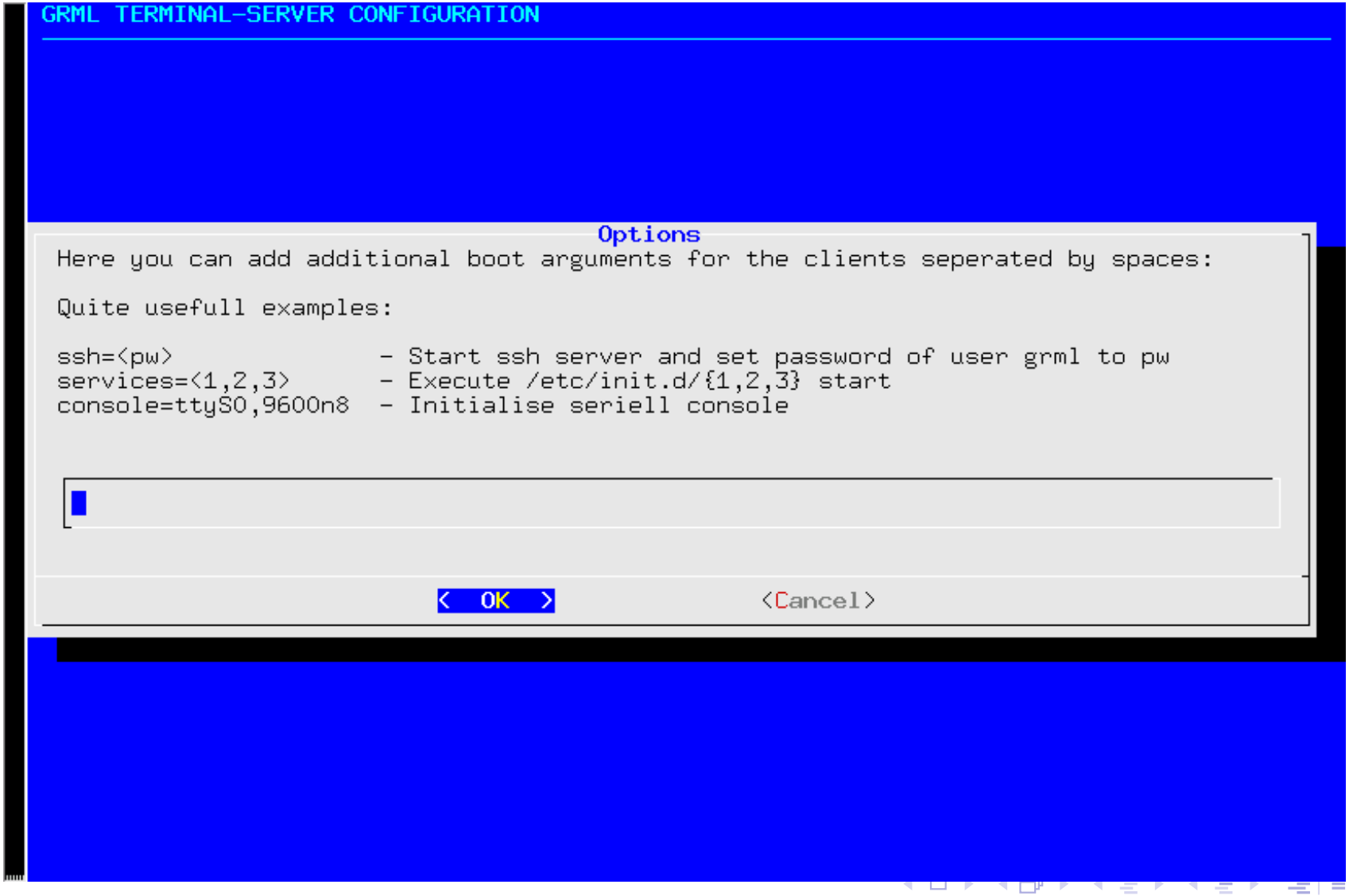
Konfiguration von grub

```
GRML TER                    Grub configuration

Give me the types of network cards you would want to boot from.
If you networkcard is allready listed press <OK>!

WARNING: DON't add all NIC's, grub could possibly stop working!!

[*] 3c503          3Com_Etherlink2
[ ] 3c503-shmem    3Com_Etherlink2
[ ] 3c503-aui      3Com_Etherlink2_AUI
[ ] 3c507          3Com
[*] 3c509          3Com
[ ] 3c529          3Com
[ ] 3c595          3Com
[*] 3c90x          3Com
[ ] cs89x0         Crystal_Semiconductor
[ ] davicom        DM9102/DM9009
[ ] depca          DE100/DE200
[ ] eepro          Intel_Ethernetexpress_Pro/10
[*] eepro100       Intel_Ethernetexpress_Pro/100
[*] epic100        SMC_83c170
[ ] exos205        EXOS205
[ ] ni5210         Racal-Interlan
[ ] lance          Lance_PCNet/AMD_HomePNA
[ ] ne2100         Novell_NE2100/NE1500
[ ] ni6510         Racal-Interlan
[ ] ni5010         Racal-Interlan
[ ] ne             NE1000/2000_ISA
[*] ns8390         NE2000_PCI_RTL8029
[ ] wd             WD8003/8013_SMC8216/8416
[ ] otulip         Old_tulip_clones
[*] rtl8139        Realtek8139/SMC1211/D-Link_DFE530TX/DFE538TX
    v(+)

                   <  K  >          <Cancel>
```

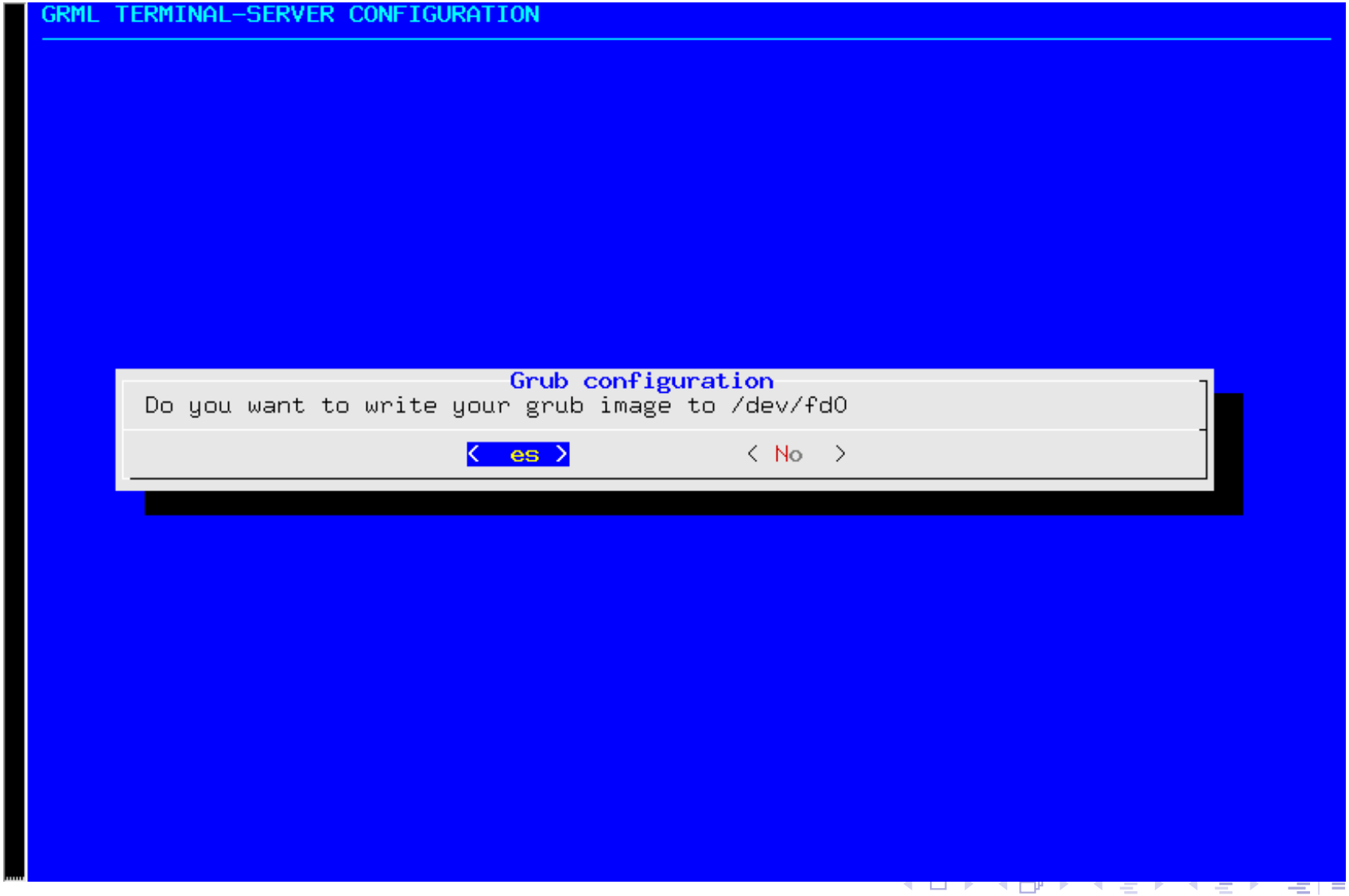
Vordefinierte Optionen



Zusätzliche Optionen



Soll grub auf eine Diskette geschrieben werden



Erfolgreicher Start

```
Creating initrd: /etc/grml/terminalserver/minirt26.gz
done
Starting tftpd...done
Starting dhcpd...done
Starting nfs...Starting portmap daemon....
Starting NFS common utilities: statd.
Exporting directories for NFS kernel daemon...done.
Starting NFS kernel daemon: nfsd mountd.
done
grml-terminalserver  15.01s user 2.38s system 49% cpu 34.928 total
root@top /home/einstein #
```

Fragen

Fragen?

Teil II

Theorie

Was passiert vor dem Bootloader

1. Client wird gestartet
2. Client fordert IP vom DHCP-Server an
3. Client bekommt vom DHCP-Server die ihm zugewiesene IP und die des TFTP-Servers[8]
4. Client lädt den Bootloader z.b pxelinux vom TFTP-Server

[8] Diese IP wird von einigen Clients leider ignoriert und der TFTP Request einfach an die IP des DHCP-Servers geschickt

Was passiert im Bootloader

1. *(Grub)* - Lädt die menu.lst durch eine fix eincompilierte Konfiguration vom TFTP-Server
2. Lädt den Kernel und die Initrd vom TFTP-Server und führt den Kernel aus

Was passiert in der Initrd

1. Versuch die Netzwerkkarte zu erkennen
2. Das richtige Kernelmodule für diese Netzwerkkarte laden
3. Eine IP Addresse per DHCP anfordern[9]
4. Die per NFS exportierte CD vom Server mounten
5. Das komprimierte Root Filesystem[10] mounten

[9] Die IP vom Bootloader kann nicht wiederverwendet werden
[10] Squashfs

Welche Services werden benötigt

- *DHCP*: Um die IP-Addressen der Netzwerkclients zu verwalten
- *TFTP*: Damit pxelinux/grub den Kernel und die Initrd auf den Client laden können
- *NFS*: Um den Clients den schnellen Zugriff auf das root Dateisystem zu gewährleisten

DHCP-Server

Verwendete Software: ISC[11] DHCP-Server Version 3

dhcpd.conf

```
if substring(option vendor-class-identifier,0,9) \
    = ``Etherboot''
{
    filename ``etherboot.nbi'';
} else {
    filename ``pxelinux.0'';
}
```

[11] Internet Software Consortium

TFTP-Server

Verwendete Software: HPA TFPT Server

TFTP-Server Startkommando

```
in.tftpd -l -s /var/lib/tftpboot
```

- -l : Starte in stand-alone Modus
- -s : Wechsle root Verzeichniss

NFS-Server

Verwendete Software: NFS Kernel server

NFS-Server Konfiguration

```
exportfs -o ro,no_root_squash,async,nohide \
    $NETWORK_/$NETMASK_:$MOUNT_POINT_
```

- *ro* : Read-only exportieren
- *async* : Antworten werden verschickt bevor die Daten die Festplatte erreicht haben
- *no_root_squash* : Rootzugriffe erlauben[12]
- *nohide* : Exportiert auch bereits gemountete Verzeichnisse

[12] Normal wird uid/gid 0 zur anonymen uid/gid gemappt

Allgemeines

Zwei unterstützte Bootloader bringen nur Vorteile:

- *(pxelinux)* Große Kompatibilität
- *(grub)* Netzwerkbooten ohne PXE Unterstützung möglich
- *(grub)* Grub kann auf der lokalen Festplatte installiert werden
- *(grub)* Theoretisch auch booten über Wlan möglich

Erklärung der von grml benutzten Bootparameter

- *nfsdir* : IP-Addresse und Pfad des Inhaltes einer grml-CD
- *nodhcp* : Keinen erneuten DHCP-Request starten
- *noprompt* : Keine Frage ob die CD ausgeworfen werden soll
- *noeject* : Die CD soll nach dem Herunterfahren nicht ausgeworfen werden

Konfiguration von grub

Grub wird mit einer einkompilierten Konfiguration erstellt welche eine weitere Konfiguration vom TFTP-Server lädt.

preset-menu.lst

```
dhcp
default 0
timeout 0
title Load config from Tftp server
  configfile (nd)/menu.lst
```

Beispielkonfiguration

menu.lst

```
title GRML
  root (nd)
  kernel /linux26 ramdisk_size=16384 \
    root=/dev/ram0 rw init=/etc/init \
    nfsdir=$IP_:$MOUNT_POINT_ nodhcp \
    noprompt noeject apm=power-off nomce
  initrd /minirt26.gz
...
title Reload config
  configfile (nd)/menu.lst
```

Beispielkonfiguration

pxelinux.cfg/default

```
LABEL grml
KERNEL linux26
APPEND ramdisk_size=16384 root=/dev/ram0 rw \
    init=/etc/init nfsdir=$IP_:$MOUNT_POINT_ \
    nodhcp noprompt noeject apm=power-off nomce \
    initrd=minirt26.gz vga=791
```

Fragen

Fragen?

Ressourcen im Web

- http://grml.org
- http://wiki.grml.org
- http://grml.org/terminalserver/
- http://wiki.grml.org/doku.php?id=terminalserver
- http://hg.einsteinmg.dyndns.org/ - Mercurial trees
- http://www.fefe.de/netboot/how-to-netboot-installer.html

Anhang

8 Ohne grml-terminalserver
- Vorraussetzungen
- Anleitung

Booten von grml ohne grml-terminalserver

Vorraussetzungen:

- grml-CD oder ISO
- Das für die zu bootende grml Version vorgesehene Netzwerkbootpaket (z.B. `http://grml.org/terminalserver/grml_netboot_package_0.7.tar.bz2`)
- DHCP-Server
- TFTP-Server
- NFS-Server

Nötige Schritte um grml in eine bestehende Netzwerkbootumgebung zu integrieren

- Downloaden und entpacken des Netzwerkbootpaket
- Kopieren von linux26 und minirt26.gz in das tftpboot Verzeichniss
- Den Inhalt des grml ISO oder der CD per NFS freigeben
- Grml in die pxelinux Konfiguration eintragen[13]

[13] siehe Beispielkonfiguration Seite 32