



- ✓ Vendor Independent
- ✓ Community Supported
- ✓ Respected Worldwide
- ✓ Distribution Neutral



## 1.102.2 - Installation eines Bootmanagers

**Beschreibung:** Prüfungskandidaten sollten in der Lage sein, einen Bootmanager auszuwählen, zu installieren und zu konfigurieren. Dieses Lernziel beinhaltet das Bereitstellen alternativer Bootmöglichkeiten (z.B. booten mittels Bootdiskette / CDROM). Kenntnisse über grundlegende Interaktionen mit dem Bootloader.

### Die wichtigsten Dateien, Bezeichnungen und Anwendungen:

`/boot/grub/menu.lst`

`grub-install`

MBR

Superblock

`/etc/lilo.conf`

`lilo`



## Master Boot Record

```

Datei Bearbeiten Ansicht Terminal Hilfe
root@pem-lx:~#
root@pem-lx:~# hexdump -n 512 -C /dev/sda
00000000 eb 48 90 00 04 00 dc 02 00 00 00 00 00 00 4e 50 |.H.....NP|
00000010 fa 33 c0 bc 00 66 8e d0 50 07 50 1f fb fc bf 00 |.3...f..P.P....|
00000020 08 be 00 7c b9 00 01 f3 a5 50 bf 2f 08 57 cb bb |...|....P./..W..|
00000030 00 06 be 02 08 0f b6 0c b8 01 02 ba 80 00 03 02 |.....|
00000040 ff 00 00 20 01 00 00 00 00 02 fa 90 90 f6 c2 80 |pres-minimizer....|
00000050 75 02 b2 80 ea 59 7c 00 00 31 c0 8e d8 8e d0 bc |linux.Y|..1.....|
00000060 00 20 fb a0 40 7c 3c ff 74 02 88 c2 52 be 7f 7d |...@|<.t...R..}|
00000070 e8 34 01 f6 c2 80 74 54 b4 41 bb aa 55 cd 13 5a |.A...+T.A..|
00000080 52 72 49 81 fb 55 aa 75 43 a0 41 7c 84 00 00 00 |
00000090 83 e1 01 74 37 66 8b 4c 10 be 05 7c c6 00 00 00 |
000000a0 66 8b 1e 44 7c c7 04 10 00 c7 44 02 01 00 00 00 |
000000b0 5c 08 c7 44 06 00 70 66 31 c0 89 44 04 00 00 00 |
000000c0 0c b4 42 cd 13 72 05 bb 00 70 eb 7d b4 00 00 00 |
000000d0 73 0a f6 c2 80 0f 84 ea 00 e9 8d 00 be 00 00 00 |
000000e0 44 ff 00 66 31 c0 88 f0 40 66 89 44 04 00 00 00 |
000000f0 ca c1 e2 02 88 e8 88 f4 40 89 44 08 31 00 00 00 |
00000100 c0 e8 02 66 89 04 66 a1 44 7c 66 31 d2 00 00 00 |
00000110 88 54 0a 66 31 d2 66 f7 74 04 88 54 0b 00 00 00 |
00000120 3b 44 08 7d 3c 8a 54 0d c0 e2 06 8a 4c 00 00 00 |
00000130 08 d1 8a 6c 0c 5a 8a 74 0b bb 00 70 8e 00 00 00 |
00000140 b8 01 02 cd 13 72 2a 8c c3 8e 06 48 7c 00 00 00 |
00000150 00 01 8e db 31 f6 31 ff fc f3 a5 1f 61 00 00 00 |
00000160 7c be 85 7d e8 40 00 eb 0e be 8a 7d e8 38 00 eb ||.}.@.....}.8..|
00000170 06 be 94 7d e8 30 00 be 99 7d e8 2a 00 eb fe 47 |...}.0...}.*...G|
00000180 52 55 42 20 00 47 65 6f 6d 00 48 61 72 64 20 44 |RUB .Geom.Hard D|
00000190 69 73 6b 00 52 65 61 64 00 20 45 72 72 6f 72 00 |isk.Read. Error.|
000001a0 bb 01 00 b4 0e cd 10 ac 3c 00 75 f4 c3 00 00 00 |.....<.u.....|
000001b0 00 00 00 00 00 00 10 2c 43 19 00 00 80 01 00 00 |.....,C.....|
000001c0 01 00 83 fe ff ff 3f 00 00 00 00 3d fc 06 00 fe |.....?....=....|
000001d0 ff ff 05 fe ff ff 50 56 4b 0e fe 64 9c 00 00 fe |.....PVK..d....|
000001e0 ff ff 83 fe ff ff bf 3d fc 06 f0 62 a9 03 00 00 |.....=...b....|
000001f0 00 00 00 00 00 00 00 00 00 00 00 00 55 aa |.....U..|
00000200
root@pem-lx:~#

```

### Partitionen

```

Datei Bearbeiten Ansicht Terminal Hilfe
root@pem-lx:~# fdisk -l
Platte /dev/sda: 128.0 GByte, 128035676160 Byte
255 Köpfe, 63 Sektoren/Spuren, 15566 Zylinder
Einheiten = Zylinder von 16065 x 512 = 8225280 Bytes
Disk identifizier: 0x19432c10

   Gerät boot.      Anfang       Ende   Blöcke  Id System
---
/dev/sda1 *          1         7295   58597056 83  Linux
/dev/sda2             14929      15566    5124735  5  Erweiterte
/dev/sda3             7296      11119   30716280 83  Linux
/dev/sda5             14929      15566    5124703+ 82  Linux Swap / Solaris

```

Magic Number oder  
Magic Word (55 aa)



## Master Boot Record (MBR)

Der MBR ist in der x86-Architektur der erste physikalische Datenblock (512 Byte) eines bootfähigen Mediums, wie beispielsweise einer Festplatte. Bei diesem Bootblock handelt es sich im eigentlichen Sinne um nichts anderes als einen Speicherbereich, welcher ausführbaren Code enthält. Dieser Code wird beim Bootvorgang geladen und ausgeführt.

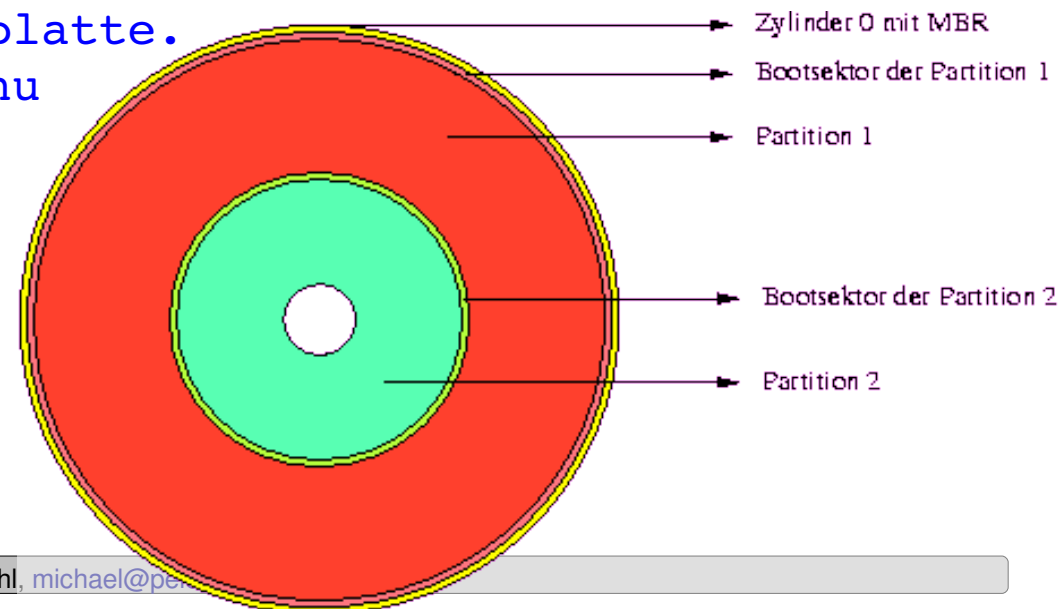
HD-Zylinder 0:

MBR(512 bytes) mit Informationen über Partitionstabelle & Partitionsgröße(n)

Stage 1: BIOS liest MBR und startet ausführbaren Code des Bootblocks von der Festplatte.

Stage 2: Präsentiert das Boot-Menü und wartet...

Stage 3: Startet das selektierte Betriebssystem





### LILO

|                   |  |
|-------------------|--|
| /etc/lilo.conf    | Konfigurationsdatei  |
| /boot/boot.b      | Boot-Menu Datei  |
| /boot/boot-menu.b | Weitere mögliche Menu Elemente   |
| /boot/message     |  |
| /boot/boot-bmp.b  |  |
| /boot/map         | Physikalische Adresse und Größe der Kernel-Datei.  |
| /sbin/lilo        | Lesen der Konfigurationsdatei und schreiben in den MBR; nutzt BIOS Bios-Funktionen und erzeugt /boot/map |

**Wichtig: Nach jeder Änderung an /etc/lilo.conf oder irgend einer Datei im /boot directory muß lilo aufgerufen werden.**



### LILO boot:

Das Wort LILO wird Zeichen für Zeichen auf den Bildschirm geschrieben. Tritt ein Fehler auf gibt LILO damit den Fehler an...

**nichts passiert** lilo nicht installiert oder lilo ist auf einem Bootsektor, der nicht aktiviert ist.  
**LFehlernummer** Erster Teil des Bootloaders lokalisiert und geladen, aber er kann den zweiten Teil nicht laden. Die zweistellige Fehlernummer gibt genauere Hinweise auf den Grund an.

**LI** Der erste Teil des Bootloaders hat den zweiten Teil geladen, kann ihn aber nicht ausführen. Gründe: Inkompatibilität mit der Plattengeometrie, oder die Datei /boot/boot.b wurde von der Stelle bewegt, an der sie lag, als lilo installiert (aufgerufen) wurde.

**LIL** Der zweite Teil des Bootloaders wurde gestartet, kann aber die Beschreibungstabelle des map-files nicht laden. Möglicher Medien-Fehler (Oberflächenbeschädigung) oder unpassende Plattengeometrie.

**LIL?** Der zweite Teil des Bootloaders wurde an eine falsche Adresse geladen. Häufigste Ursache: die Datei /boot/boot.b wurde verändert oder bewegt.

**LIL-** Die Beschreibungstabelle ist beschädigt. Entweder ein Oberflächenfehler, oder die Datei /boot/map wurde verändert oder bewegt.

**LILO** Alle Teile von lilo wurden ordnungsgemäß geladen.



## Datei: /etc/lilo.conf (aus Linux ML)

```
boot    = /dev/sda
read-only
prompt
timeout = 30
lba32
append  = "apm=power-off"
default = LinuxMuster26

disk    = /dev/sda

image   = /boot/vmlinuz
label   = LinuxMusterSMP
root    = /dev/sda3
#initrd = /boot/initrd

#  Kernel 2.6
image   = /boot/vmlinuz_26
label   = LinuxMuster26
root    = /dev/sda3
```





## GRUB (GRand Unified Boot Loader)

Stage 1: MBR (Master Boot Record). Der Bootloader liest nun Stage 2 aus einer beliebigen Partition aus, wobei die genaue Position in einer Map-Datei steht.

Stage 2: normalerweise unter `/boot/grub/stage2`. Das Verzeichnis enthält auch die Dateisystemtreiber, die eigentliche Shell, den Code für das Menü und den Loader für den Kernel. Nach dem Laden dieser Datei wird die **Konfigurationsdatei** (`/boot/grub/menu.lst`) angezeigt

[Diese Lösung hat den Nachteil, dass der Bootloader nach Verschieben oder Ändern von Stage 2 nicht mehr bootfähig ist. Deswegen wurde zwischen Stage 1 und 2 eine Zwischenstufe, Stage 1.5, eingeführt. Diese liegt in den Datenblöcken zwischen dem MBR und dem ersten Block der ersten Partition. **Stage 1.5** hat nun die Fähigkeit, genau ein Dateisystem zu lesen. Zur Zeit gibt es Stage 1.5 für die Dateisysteme FAT, Minix, Ext2/3, JFS, ReiserFS sowie XFS.]

Um mit grub einzurichten und damit zu arbeiten genügt es einmal den Befehl einzugeben :

```
/sbin/grub-install /dev/hda
```

Danach kann das Bootmenü jederzeit verändert werden und grub liest das veränderte Menü beim nächsten Bootvorgang ein.





## Datei: /boot/grub/menu.lst (boot/grub/grub.conf)

### Beispiel für Festplatteneinteilung:

```
/dev/hda1 /boot-Partition von Linux  
/dev/hda2 WindowsXP-Partition  
/dev/hda3 Rootpartition von Linux  
/dev/hda4 swap
```

```
default=0  
timeout=10  
splashimage=(hd0,0)/boot/grub/splash.xpm.gz  
title Linux (2.6.28)  
    root (hd0,0)  
    kernel /bzImage-2.6.28 ro root=/dev/hda3  
    initrd /initrd-2.6.28.img  
title Linuxalt (2.4.18)  
    root (hd0,0)  
    kernel /bzImage-2.4.18 ro root=/dev/hda3  
    initrd /initrd-2.4.18.img  
title Windows XP  
    title Microsoft Windows XP  
        root (hd0,1)  
        savedefault  
        makeactive  
        chainloader +1
```



### Übung: 1.102.2

- (1) Benutzen Sie die Befehle „apropos“ „whatis“ „man“ und „info“ immer in Kombination mit einem Programmnamen, um möglichst viele Informationen über die genannten Programme und Konfigurations-Dateien herauszufinden. Beispiel: `apropos grub`.
- (2) Welche Unterschiede bestehen zwischen `apropos` und `whatis`?
- (3) Beim Starten von `lilo` und `grub`, kann man dem jeweiligen Kernel Parameter übergeben. Was bewirkt folgendes: `[lilo] linux init=/bin/bash` (ähnlich in `grub` über die `grub-shell`)?
- (4) Welche Wirkung besitzt die Zeile in `/etc/lilo.conf` ?  
`append="reboot=warm"`
- (5) Wann müssen Sie das Programm `grub-install` aufrufen?
- (6) Was dient als Mountpoint für das Einhängen von Dateisystemen?
- (7) Wann müssen Sie das `mkswap` Programm aufrufen?
- (8) Wie lautet der korrekte Gerätenamen der 2. Logischen Partition der Slave-Platte am 1. IDE-Controller?



### Lösung zu Übung: 1.102.2

- (1)
- (2) Welche Unterschiede bestehen zwischen apropos und whatis?
  - apropos durchsucht die Manual-Kurzbeschreibungen – whatis durchsucht die Indexdatenbank nach Kurzbeschreibungen
- (3) Beim Starten von lilo und grub, kann man dem jeweiligen Kernel Parameter übergeben. Was bewirkt folgendes: [lilo] linux init=/bin/bash (ähnlich in grub über die grub-shell)?
  - man wird root ohne Passwort
- (4) Welche Wirkung besitzt die Zeile in /etc/lilo.conf ? append="reboot=warm"
  - (Kein Speichertest nach Reboot)
- (5) Wann müssen Sie das Programm grub-install aufrufen?
  - (Neuinstallation von Grub oder MBR gelöscht)
- (6) Was dient als Mountpoint für das Einhängen von Dateisystemen?
  - Verzeichnisse
- (7) Wann müssen Sie das Programm mkswap aufrufen?
  - Nach dem Neuanlegen einer Swap-Partition oder Swapdatei
- (8) Wie lautet der Gerätenamen der 2. Logischen Partition der Slave-Platte am 1. IDE-Controller?
  - /dev/hdb6