

Installation

<https://doc.opensuse.org/documentation/leap/startup/html/book.opensuse.startup/cha.inst.html>

Boot Parameter für Problemfälle

Ab Kernel 2.6.28 wurde der Video-Modus der Grafikkarten vom [XServer](#) in den Kernel verlagert. Dadurch wurde es möglich, dass ein hochauflösender "Boot Splash" (siehe [Plymouth](#)) flackerfrei angezeigt werden kann. Bei einigen Karten bleibt dieser "Boot Splash" hängen oder endet in einem schwarzen Bildschirm. Dieses Verhalten tritt typischerweise bei Hardware mit Intel und nVidia-Karten auf. Mit `nomodeset` wird der grafische Boot aus dem Kernel umgangen:

`nomodeset`

F3 = Text mode (ncurses) oder Kernel Parameter

`Textmode=1`

Automatisches Update des Installers (Download)

`self_update=1`

UEFI

GRUB 2 for EFI on

openSUSE Leap does not support a boot prompt or function keys for adding boot parameters. By default, the installation will be started with American English and the boot media as the installation source. A DHCP lookup will be performed to configure the network. To change these defaults or to add additional boot parameters you need to edit the respective boot entry. Highlight it using the arrow keys and press E. See the on-screen help for editing hints (note that only an English keyboard is available now). The Installation entry will look similar to the following:

setparams 'Installation'

set gfxpayload=keep echo 'Loading kernel ...' linuxefi /boot/x86_64/loader/linux splash=silent echo 'Loading initial ramdisk ...' initrdefi /boot/x86_64/loader/initrd Add space-separated parameters to the end of the line starting with `linuxefi`. To boot the edited entry, press F10. If you access the machine via serial console, press Esc - 0. A complete list of parameters is available at <http://en.opensuse.org/Linuxrc>. The most important ones are:

Table 2.2: Installation Sources [Report Bug](#) <#>

CD/DVD (default)	<code>install=cd:/</code>
Hard disk	<code>install=hd:/?device=sda/PATH_TO_ISO</code>

SLP	install=slp:/
FTP	install= ftp:///ftp.example.com/PATH_TO_ISO//
HTTP	install= http:///www.example.com/PATH_TO_ISO//
NFS	install=nfs:/PATH_TO_ISO
SMB / CIFS	

Table 2.3: Network Configuration [Report Bug #](#)

DHCP (default)	netsetup=dhcp
Prompt for Parameters	netsetup=hostip,netmask,gateway,nameserver
Host IP address	hostip=192.168.2.100 hostip=192.168.2.100/24
Netmask	netmask=255.255.255.0
Gateway	gateway=192.168.5.1
Name Server	nameserver=192.168.1.116 nameserver=192.168.1.116,192.168.1.118
Domain Search Path	domain=example.com

Table 2.4: Miscellaneous [Report Bug #](#)

Driver Updates: Prompt	dud=1
Driver Updates: URL	dud= ftp:///ftp.example.com/PATH_TO_DRIVER// dud= http:///www.example.com/PATH_TO_DRIVER//
Installation Language	Language=LANGUAGE Supported values for <i>LANGUAGE</i> are, among others, cs_CZ, de_DE, es_ES, fr_FR, ja_JP, pt_BR, pt_PT, ru_RU, zh_CN, and zh_TW.
Kernel: No ACPI	acpi=off
Kernel: No Local APIC	noapic
Video: Disable KMS	nomodeset

Video: Start Installer in Text Mode	Textmode=1
-------------------------------------	------------

Warning: Custom Partitioning on UEFI Machines

A UEFI machine *requires* an EFI system partition that must be mounted to `/boot/efi`. This partition must be formatted with the FAT file system. If an EFI system partition is already present on your system (for example from a previous Windows installation) use it by mounting it to `/boot/efi` without formatting it.

Warning: Custom Partitioning and Snapper

openSUSE Leap can be configured to support snapshots which provide the ability to do rollbacks of system changes. openSUSE Leap uses Snapper in conjunction with Btrfs for this feature. Btrfs needs to be set up with snapshots enabled for the root partition. Refer to Book “*Reference*”, Chapter 3 “System Recovery and Snapshot Management with Snapper” for details on Snapper.

Being able to create system snapshots that enable rollbacks requires most of the system directories to be mounted on a single partition. Refer to Book “*Reference*”, Chapter 3 “System Recovery and Snapshot Management with Snapper”, Section 3.1 “Default Setup” for more information. This also includes `/usr` and `/var`. Only directories that are excluded from snapshots (see Book “*Reference*”, Chapter 3 “System Recovery and Snapshot Management with Snapper”, Section 3.1.2 “Directories That Are Excluded from Snapshots” for a list) may reside on separate partitions. Among others, this list includes `/usr/local`, `/var/log`, and `/tmp`.

If you do not plan to use Snapper for system rollbacks, the partitioning restrictions mentioned above do not apply.

Zypper

Repos hinzufügen

```
zypper ar https://ftp.fau.de/packman/ packman
```

Liste aller installierten Pakete exportieren

```
zypper search --installed-only > locutus_packages.txt
```

Alle Pakete von einem bestimmten Repo zeigen (Repo Liste mit zypper lr)

```
zypper pa -ir [Repo-ID]
```

```
zypper search -i -r <repo alias|#|URI>
```

Andersrum: von welchem Repo stammt ein Paket:

```
zypper --no-refresh info atom
```

Welches RPM liefert eine bestimmte Datei:

```
zypper se --provides --match-exact libqconnmanbearer.so
```

Welche Abhängigkeiten hat ein Paket?

```
zypper info --requires libkde4-devel
```

Repo aktivieren

```
zypper mr -e 20
```

Repo Autorefresh aktivieren

```
zypper mr -rk 20
```

Vendor change in general

You can define a list of repositories having different "vendors" as equivalent by creating a file in the `/etc/zypp/vendors.d/` directory with a similar content:

```
[main]

vendors =
suse,opensuse,obs://build.suse.de,Packman,http://packman.links2linux.de
```

You can use any file name, but each file in this directory reflects a group of equivalent vendors. Libzypp makes a string comparison (like `strncmp`, case-insensitive) whereas the beginning of the strings are compared only. e.g. vendor "opensuse11.0" is compatible to "openSUSE".

Paketlisten ex/importieren

Export

```
rpm -qai | grep "^Name          :" | awk '{print $3}' | xargs > rpm_pkgs_list
```

Import

```
zypper in $(cat rpm_pkgs_list)
```

Alternativ per rpm:

My own approach is now to run `'rpm -qa -queryformat '%{NAME}\n'` and with `'zypper lr -pu'` prior to my regular backups to maintain a up-to-date list of both, installed packages and repositories which will give me enough information to easily get the old system back just in case anything goes really wrong... Best regards, Kallistos

Multiversion Kernel

Ältere Kernel nicht löschen: /etc/zypp/zypp.conf

Letzte zwei Kernel werden erhalten, außerdem ausdrücklich 5.7.11-1.:

```
# Default value:
##          empty
##
## multiversion = provides:multiversion(kernel)

## keep last 3 kernels and 5.7.11-1
multiversion.kernels = latest,latest-1,5.7.11-1,running
```

Tumbleweed: ältere Kernel gibt es unter <http://download.opensuse.org/history/> Zum Installieren wird -oldpackage gebraucht:

```
zypper in --oldpackage kernel-default-5.7.11-1.2.x86_64
```

kernel-syms, kernel-default-devel und kernel-devel nicht vergessen!

Aktuellste KDE Pakete auf LEAP

Standard ist KDE 5.12 LTS, mit diesen Repos kommt 5.15 aufs System

```
zypper ar -p 75
http://download.opensuse.org/repositories/KDE:/Qt5/openSUSE_Leap_15.0 KQ
zypper ar -p 75
http://download.opensuse.org/repositories/KDE:/Frameworks5/openSUSE_Leap_15.0 KF
zypper ar -p 75
https://download.opensuse.org/repositories/KDE:/Applications/KDE_Frameworks5
_openSUSE_Leap_15.0 KA
zypper -v dup --allow-vendor-change
```

Snapper

Doku: https://de.opensuse.org/openSUSE:Snapper_Tutorial
<https://doc.opensuse.org/documentation/leap/reference/html/book.opensuse.reference/cha.snapper.html>

Snapshots zeigen

```
snapper list
```

Snapshots löschen

```
snapper delete [] []
```

Snapshot Unterschiede

```
snapper status 41..39
```

Snapshot Diff

```
snapper diff 71..72 /etc/zypp/zypp.conf
```

Snapshot anlegen

```
snapper create --description "Snapshot für Woche 2 2014"
```

Säuberungs-Algorithmus

Wenn Sie keinen guten Grund haben es anders zu machen, sollten Sie immer den Säuberungs-Algorithmus mit aufrufen, wenn Sie Snapshots erstellen. Sonst wird der Snapshot nie gelöscht, außer Sie machen das manuell. Sie machen das durch das Hinzufügen von Folgendem in Ihrem Snapper-Befehl

```
--cleanup-algorithm <Nummer|Zeitleiste|leeres-vor-nachher>
```

Rollback

Setzt den aktuell aktiven **read only** snapshot auf **read-write** und erzeugt einen weiteren Snapshot als read-only Kopie. Beim nächsten boot wird der read-write Snapshot benutzt.

```
locutus:~ # snapper rollback
Ambit is classic.
Creating read-only snapshot of default subvolume. (Snapshot 1946.)
Creating read-write snapshot of current subvolume. (Snapshot 1947.)
Setting default subvolume to snapshot 1947.
```

Für den Rollback Snapshot kann man eine Beschreibung hinzufügen, die im Grub gezeigt wird:

```
snapper modify --userdata "bootloader=rollback from [source snapshot]
[comment]" [snapshot ID]
```

Build Umgebung

Samba

Ab **Samba 4.7.0** / (Ubuntu: Bionic Beaver) versucht **smbclient** die Verbindung im Dialekt SMB3_11 aufzubauen. In der Folge funktioniert das Browsing (`smbclient -L <server>`) nicht mehr. Lösung ist es den Dialekt SMB1 mittels Parameter `-m` zu erzwingen (`smbclient -mNT1 -L <server>`). Weitere Infos siehe [Samba 4.7.0](#)

Shares auf einen Samba Server abscannen

Scannen von Samba Servern in der eigenen Broadcast domain

```
nmblookup __SAMBA__
```

```
'thommie@locutus:~> smbclient -N -L //nas2 Anonymous login successful
Sharename Type Comment --- -- --- video Disk video photo Disk photo music
Disk music admin Disk admin buch Disk buch downloads Disk downloads backup
Disk backup datanas1 Disk datanas1 public Disk public data1 Disk data1 IPC$
IPC IPC Service (nas2 server) Reconnecting with SMB1 for workgroup listing.
Anonymous login successful Server Comment // ''
```

mit `smbclient -L` wird ein Passwort für den aktuellen Shell User abgefragt

Samba Share fest mounten

Mit User mounten

```
mount -t cifs -o vers=3.0,username=thommie, password=xxx
nas2.netzwissen.loc/photo /mnt/nas_photo/
```

Anonym mounten für bestimmten User

```
===== Rescue System =====
```

- Boot Tumbleweed von USB.
- Kaputtes System mounten und changeroot:

```
**Root Partition mounten**
```

```
<code> mount /dev/<device name> /mnt
```

Virtuelle Dateisysteme

```
for i in $(ls /proc/sys/dev); do mount --rbind /$i /mnt/$i ; done
```

oder manuell

```
mount --rbind /proc /mnt/proc
```

```
mount --rbind /sys /mnt/sys  
mount --rbind /dev /mnt/dev
```

Danach chroot und alles lokale mounten

```
chroot /mnt  
mount -a
```

Mit exit wieder raus aus dem chroot

Build Umgebung

```
sudo zypper install --type pattern devel_basis
```

From:

<https://wiki.netzwissen.de/> - **netzwissen.de Wiki**

Permanent link:

<https://wiki.netzwissen.de/doku.php?id=opensuse&rev=1724929747>

Last update: **29/08/2024 - 11:09**

