

---

# SLES9 Network Install

NFS/FTP/HTTP Installation with SLES9

Anas Nashif, SUSE Linux AG

Copyright © 2004 SUSE Linux AG

Revision History

Revision 1.2

2004/05/25 17:20:46

Added section about SAMBA Setup. More information about CD Naming Product CD/ Core CD split

Revision 1.1.1.1

Date: 2004/05/20 02:46:18

Adapted instorder format to specification.

## Table of Contents

1. Introduction .....	2
2. The installer source .....	3
2.1. File Structure on Installation CDs .....	3
2.2. The order file .....	5
2.3. The instorder file .....	6
3. Setting it all up .....	7
4. Example using SLES 9 .....	9
5. SAMBA Sources .....	12

# 1. Introduction

The *multiple source* approach of YaST gives a solution as it allows installation from several (package) sources. So a customer can start installing from the product CD, proceed with the CORE CDs, and end with a patch CD containing the latest security patches.

The process described below can be done using a new YaST module for creating sources using either CDROM media or ISO images. Using the control center, run "*Installation Server*".

## Product and CD naming confusion

ISOs and Product CDs now all are called SLES and no longer SLES+CORE. Internally we still architect the product around a set of CORE CDs and one product specific CD but this is no longer visible to the user during installation. However, If you are creating a source tree, you will notice that and should have in mind that SLES CD1 is the product CD and all other CDs are the CORE CDs.

The following scheme applies:

LINUX	Enterprise	Server	9	CD1	=	SLES	CD1
LINUX	Enterprise	Server	9	CD2	=	CORE	CD1
LINUX	Enterprise	Server	9	CD3	=	CORE	CD2
LINUX	Enterprise	Server	9	CD4	=	CORE	CD3
LINUX	Enterprise	Server	9	CD5	=	CORE	CD4
LINUX	Enterprise	Server	9	CD6	=	CORE	CD5

## 2. The installer source

When starting the installation from a media (CD, DVD, NFS, ...), `linuxrc` looks for a `/boot/root` file containing a complete (cramfs based) root filesystem for YaST.

Usually, this file is on the boot CD, but other installation types are also supported by `linuxrc`. The information, where the `/boot/root` file came from, is passed to YaST and defines the installer source.

YaST then looks for various files defining the product and packages, when those are not provided in the root of the source, it searches for `/yast/order` file. If this file exists, YaST goes into the *multiple installation* mode.

### 2.1. File Structure on Installation CDs

The following illustration shows the CD structure of the SLES CD.

```
-- boot
|  |-- loader
|-- docu
|  |-- HOWTOs
|-- dosutils
|  |-- .....
|-- media.1
|-- suse
|  |-- i586
|  |-- setup
|  |-- src
|-- yast
|  |-- core9
```

The media and product description is splitted to a set of files, separating the media preparation data from the contents (product) and making it possible to combine several products on a single media.

At top-level of a media, a directory `/media.1/` should exist. If it does not exist, the presence of the correct media can not be verified. This may be convenient for an on-line media which just provides a set of packages.

In this case, no verification takes place. The numeric extension depends on the media number in case of splitted medias. The first media has `.1`, the second `.2`, and so on.

This allows merging all medias of a splitted set into a single directory. YaST always knows which media to expect and can thereby directly access the correct media directory.

#### 2.1.1. Files inside boot/

the directory `boot/` contains installation images and floppy images that can be used to create boot and installation floppies. The `root` files contains a complete (cramfs based) root filesystem for YaST. This file contains the actual installer. The `root` file is fetched by `linuxrc` from the installation source and is mounted in the installation environment. `Linuxrc` starts the installer from within this file.

The directory `boot/loader` contains the kernel and the initial ramdisk used for installation. It also contains the `isolinux` files used for creating the CD.

## 2.1.2. Files inside media.N/

Inside the media.N/ directory the files media and products must exist. Optionally, a file named patches may exist to denote a media with patches (e.g. for YaST online update).

- The /media.N/media file

The media file contains the media identifier needed to determine correctness when using removable media. In case of splitted media, it must be present on every media of the set.

This should contain a key to quickly identify correctness of the inserted media. It is not shown to the user but should contain human-readable data for debugging purposes.

- The products file

The products file contains directory specifications for each product. In case of splitted media, it must be present of the first media of the set.

- The patches file

The patches file contains a single directory specification for the root of the patches directory tree. In case of splitted media, it must be present of the first media of the set.

- The info.txt file

This file contains any licensing information that should be presented to the user when YaST is started. If not present, no license screen is shown. On the SLES CD, it contains the EULA.

The identification data is stored inside the file on purpose. It could also be encoded in the file name (older SuSE media did it). But this requires a directory scan for file access and should be avoided.

By storing the vendor name in the file, the timestamp can be kept vendor specific. (It's the name of the vendor who created the media which might be different from a product vendor! It's just used as an additional key for media identification.)

The media count is needed if the installation data is split up on multiple (identical) medias. The media file on the first media should contain the total count of medias.

The media count is a numerical value (all characters are digits). It is an optional information on the other media.

A forth line may contain flags further describing the media. In order to distinguish the flags from the count line, the flags values start with a letter.

All media of a media-set must have identical vendor and data information in the media file.

### Example 1. media directory

```
# ls -l media.1
info.txt
media
products
```

```
# cat media.1/media
```

```
SuSE Linux AG  
20040408195328  
1
```

```
# cat media.1/products  
/ SUSE-Linux-SLES-i386 9
```

### 2.1.3. The contents files

This file resides inside the product directory as specified by the products file on the media. It contains all product-specific data to describe and identify the contents of the product. If there was no products file, the product directory defaults to the root directory on the media.

The content file might be missing if the media does not contain a product (but just a set of packages).

#### Example 2. The contents file

```
# cat content  
PRODUCT SUSE SLES  
VERSION 9  
DISTPRODUCT SUSE-Linux-SLES-i386  
DISTVERSION 9-0  
BASEPRODUCT SUSE CORE  
BASEVERSION 9  
YOUTYPE business  
YOUPATH i386/update/SUSE-SLES/9  
YOUURL http://www.suse.de/cgi-bin/suseservers.cgi  
VENDOR SUSE LINUX AG  
RELNOTESURL http://www.suse.com/relnotes/i386/SUSE-SLES/9/release-notes.rpm  
ARCH.i686 i686 i586 i486 i386 noarch  
ARCH.i586 i586 i486 i386 noarch  
ARCH.i486 i486 i386 noarch  
ARCH.i386 i386 noarch  
DEFAULTBASE i386  
REQUIRES distribution-release  
LINGUAS de en  
LABEL SUSE SLES Version 9  
LABEL.de SUSE SLES in der Version 9  
DESCRDIR suse/setup/descr  
DATADIR suse  
FLAGS update  
LANGUAGE en_US  
TIMEZONE America/Los_Angeles
```

## 2.2. The order file

The order file defines which sources are needed for installation and their priority for installation. A typical example for an order file is given here:

```
# yast/order: priority of installation sources
#
# source-      product
# description  path
/
/yast/core9
```

The order file is ASCII, line based, '#' at the beginning of a line denote a comment line.

Each non-comment line should contain two (tab separated) directory specifications. If the second one is omitted, it defaults to "/".

The first directory specifies the location of the description. The second directory specifies the location of the product as defined here. For CD/DVD bases products, this is usually "/". But for network installations, or if several product are placed on a single CD, each product needs a separate path.

You can even specify a full URL (e.g. "nfs://my-nfs-server.domain.org/product/CD1") in the second column. During installation, you can't switch the type of media. If your installer source is CD, every other source must be CD based. If your installer source is NFS, every other source is expected to be on the same NFS server.

## 2.3. The instorder file

As stated above, the `/yast/order` file lists the sources in priority order. So a package from a source listed earlier in this file overrides a package (of the same name) from a later source.

But the `/yast/order` also defines the installation order, which might not be wanted when the 'base' packages are on a lower-priority source.

This is where `/yast/instorder` comes into play. This defines the installation order by repeating the entries of the first column of `/yast/order` in the right order.

A typical example for an instorder file is given here:

```
# yast/instorder: order of installation sources
# this example is used on SLES CDs.
#
/
/yast/core9
```

### 3. Setting it all up

Combining multiple sources requires that all sources comply to the UnitedLinux media description standard.

In the source directory, copy all SLES and CORE into separate directories (CDs can also be copied into one single directory, but to keep a clean structure, CDs can be copied into separate directories, i.e. using the following scheme: CORE9/CD1, CORE9/CD2, CORE9/CD3 . CORE9/CD4, CORE9/CD5 and SLES/CD1.)

Upon boot, Linuxrc requires the presence of the boot directory (which contains the `root` file), the `media.N` directory the `control.xml` file and the `content` file. Having those files in the root of the source will insure that linuxrc can start the YaST installer.

Once YaST has started, it will search the root of source for installation files using values from the `content` file name. However, to detect if multiple sources are involved, YaST first search for the `order` and `instorder` in the `yast` directory. The files in the `yast` directory will reroute YaST and direct it to search for the installation files and media description in different locations than root of the source.

On the installer source create a `yast/order` as described above and enter the location of the description data and packages. Alternatively, you can copy the description data of all sources to separate directories in the `yast` directory (This is the case on the SLES CD which contains the description data for CORE to support multiple sources using CDs).

In the 'order' example above, the 'order' directory structure looks as follows:

```

yast
|-- core9
|   |-- content
|   |-- media.1
|       |-- info.txt
|       |-- media
|       |-- products
|   |-- suse
|       |-- setup
|           |-- descr
|               |-- MD5SUMS
|               |-- common.war
|               |-- directory.yast
|               |-- packages
|               |-- packages.DU
|               |-- packages.cs
|               |-- packages.de
|               |-- packages.en
|               |-- packages.es
|               |-- packages.fr
|               |-- packages.hu
|               |-- packages.sk
|               |-- selections
|-- instorder
|-- order

```

YaST now takes this description instead of the one from the *real* media for the installation. So when asking for a specific media, this media must match the `media.1/media` file from the `order` directory.

All dependencies (package or selection) are fulfilled by looking at all installation sources. If multiple

sources provide a dependency, the source listed first in the 'order' file wins. (The installation order equals the installation priority currently).

## 4. Example using SLES 9

You should at least copy the first CD from both CORE 9 and SLES 9 sets, depending on what selection of software you want to install.

On the server side, create a directory which will serve as the installation source and make sure it is exported via NFS when creating an NFS source or make it available using the other supported protocols. This directory will be used later to specify the installation source using **linuxrc**.

The following script illustrates what directories need to be created and where you should copy the data from the CDs. In this script, it is assumed that you have ISO images. In most cases you will have CDROMs instead. Do not use this script as provided. If you wish to use it, please make sure you have changed the directory names to something suitable to your environment.

```
#!/bin/sh

instsource="/exports/sles9"
#
# Create CD directories
#
cd $instsource
mkdir -p sles9-i386/CD1
mkdir -p core9-i386/CD{1,2,3,4,5}

#
# Copy SLES CD
#
cd sles9-i386/CD1

mount -oloop /iso/SLES-9-i386-CD1.iso /mnt
cp -a /mnt/* .
umount /mnt

#
# back to root of source
#
cd ../../

#
# Copy UL CDs
#

# 1
cd core9-i386/CD1
mount -oloop /iso/CORE9-i386-CD1.iso /mnt
cp -a /mnt/* .
umount /mnt

# 2
cd ../CD2
mount -oloop /iso/CORE9-i386-CD2.iso /mnt
cp -a /mnt/* .
umount /mnt

# 3
cd ../CD3
mount -oloop /iso/CORE9-i386-CD3.iso /mnt
cp -a /mnt/* .
umount /mnt
```

```
#
# Copy all other CDs...
#

#
# back to root of source
#
cd ../../..

#
# Now we have to symlink some data to make this
# a multiple source directory
#

# Then the boot directory with the installation image
#
ln -s sles9-i386/CD1/boot

# The content file and the media directory
#
ln -s sles9-i386/CD1/content
ln -s sles9-i386/CD1/control.xml
ln -s sles9-i386/CD1/media.1

# We are almost done..
# Now we'll adapt the order files.
# Fields of the order files are TAB delimited

mkdir -p yast

printf "/sles9-i386/CD1\t/sles9-i386/CD1\n" > yast/order
printf "/core9-i386/CD1\t/core9-i386/CD1\n" >> yast/order

printf "/sles9-i386/CD1\n" > yast/instorder
printf "/core9-i386/CD1\n" >> yast/instorder

#
# End
#
```

After you have gone through all steps, you will have a structure similar to the one shown below.

```
.
|-- sles9-i386
|   |-- CD1
|       |-- boot
|       |-- docu
|       |-- media.1
|       |-- suse
|       |-- yast
|-- core9-i386
|   |-- CD1
|   |-- CD2
|   |-- CD3
|-- boot -> sles9-i386/CD1/boot
|-- media.1 -> sles9-i386/CD1/media.1
|-- content -> sles9-i386/CD1/content
```

```
|-- control.xml -> sles9-i386/CD1/control.xml  
|-- yast
```

And the root of the source will have the following content:

```
# ls -l  
  
sles9-i386  
core9-i386  
boot  
content  
control.xml  
media.1  
yast
```

## 5. SAMBA Sources

SAMBA source are a special case because it is not possible to specify a directory below a share.

To create a SAMBA share, first you need to create a directory which will host the source tree which will should be exported as a share which will be used to specify the source in linuxrc or on the kernel command line.

Copy the CDs as specified above. Create two directories, i.e. *SLES* and *CORE* under the top directory which we will call *SLES9* in this example. Copy SLES CD1 into *SLES/CD1* and SLES CDs 2-6 (CORE 1-5) into *CORE/CD{1,2,3,4,5}*

Next you will need to add all CDs as shares, using the following scheme: SLES CDs (1 CD in this case) will be exported as a share using *SLESCD1* and the CORE CDs (5 CDs) will be exported using *CORECD1* to *CORECD5*

Then Copy the following files and directories from *SLES/CD1* to the root of the tree: *content*, *media.1*, *control.xml* and *boot*

Next you need to create the order files in the directory *yast*. Create the directory *yast* and using an editor create the file *order* and add the following lines:

```
/SLES/CD1 smb://user:password@hostname/SLESCD1
/CORE/CD1 smb://user:password@hostname/CORECD1
```

The *instorder* file has the following content:

```
/SLES/CD1
/CORE/CD1
```

When installing you only need to enter the following URI to access the source tree:

```
smb://user:password@hostname/SLES9
```

When done, you should be able to see the exported tree with **smbtree**. More or less, the host with the source tree should have the following shares:

```
root@avicenna:~ # smbtree
Password:
NASHIF
  \\MASTER
    \\MASTER\C$           Default share
    \\MASTER\ADMIN$      Remote Admin
    \\MASTER\F$          Default share
```

```
\\MASTER\SLES  
\\MASTER\CORECD1  
\\MASTER\CORECD2  
\\MASTER\CORECD3  
\\MASTER\CORECD4  
\\MASTER\CORECD5  
\\MASTER\SLESCD1  
.....
```