

Creating your own Linux Live CD

Author: Luc Parson
Email: luc@tinyghost.com
Revision: Draft 0.4
Date: March 22, 2007

Table of Contents

1.Introduction	3
1.1.Scope.....	3
1.2.Assumptions.....	3
1.3.References.....	3
1.4.Revisions.....	3
2.Obtaining Software Required	4
2.1.Obtaining SlackWare Linux.....	4
2.1.1.Accessing the SlackWare web site.....	4
2.1.2.Downloading the latest version of SlackWare.....	4
2.2.Obtaining the LZMA compression patch for the Squash File System.....	4
2.3.Obtaining the Squash File System code.....	4
2.3.1.Downloading SquashFS.....	4
2.3.2.Determining which Linux kernel that SquashFS can patch.....	5
2.4.Obtaining the latest Linux kernel that SquashFS can patch.....	5
2.5.Obtaining the Linux-Live scripts.....	5
2.6.Obtaining Another Union File System (AUFS).....	6
3.Installing SlackWare Linux for the LiveCD	6
3.1.Creating the LiveCD environment.....	6
3.2.Mounting the SLackWare CD.....	6
3.3.Installing the base SlackWare packages.....	7
3.4.Un-mounting the SlackWare CD.....	7
4.Compiling the kernel system wide	7
5.Compiling the Kernel for the Live CD	8
5.1.Setting up the temporary directory.....	8
5.2.Extracting all files needed for the kernel compilation and patching.....	8
5.3.Patching the kernel.....	8
5.4.Compiling the kernel.....	9
6.Compiling External Modules (AUFS and SquashFS + LZMA)	10
6.1.Compiling and installing AUFS.....	10
6.2.Compiling and installing SquashFS with LZMA.....	11
7.Cleaning-up the LiveCD environment	12
8.Installing the Kernel in the LiveCD Environment	12
9.Creating the LiveCD Image File	12

1. Introduction

1.1. Scope

The scope of this document is to provide a step by step installation manual, for creating a Linux live CD, based on Slackware Linux.

1.2. Assumptions

It is assumed that before proceeding with this document, you have access to a box that currently runs Linux, and that you are familiar with compiling a Linux kernel.

1.3. References

Linux-Live scripts from <http://www.linux-live.org/>

slackware from <http://www.slackware.com/>

Linux Kernel from <http://www.kernel.org/>

SquashFS from <http://squashfs.sourceforge.net/>

lzma patches from <http://www.squashfs-lzma.org/>

Aufs from <http://aufs.cvs.sourceforge.net/>

1.4. Revisions

Name	Changes	Date
Luc Parson	Creation of document (Draft 0.1)	March 10, 2007
Luc Parson	Fixed a few typos (Draft 0.2)	March 12, 2007
Luc Parson	Revised every command (Draft 0.3)	March 15, 2007
Luc Parson	Added the ldconfig command in section 9 (Draft 0.4) Fixed a typo in a command in section 5.	March 22, 2007

2. Obtaining Software Required

2.1. Obtaining SlackWare Linux

2.1.1. Accessing the SlackWare web site

The first step in obtaining SlackWare, is to find a web or ftp mirror site, which carries the latest version of the SlackWare Linux distribution.

- For a list of all SlackWare mirror sites, visit: <http://www.slackware.com/getslack/>

2.1.2. Downloading the latest version of SlackWare

From the mirror sites obtained from section 2.1.1, select your country or the closest country to you, and then select a mirror site.

- Click on the mirror address, and a list of folders should appear.
- Select the latest version available that finishes with "-iso". Example: slackware-11.0-iso/
- Within the ISO folder, select the first disk. Example: slackware-11.0-install-d1.iso
- Download this file, and burn the ISO to a CD.

2.2. Obtaining the LZMA compression patch for the Squash File System

- For the latest version of LZMA visit the following web site: <http://www.squashfs-lzma.org/>
- Scroll down near the bottom of the page where it states "So, where are the patches, finally?".
- Under this section click on "LZMA SDK", and download this file. At this current date the file name is "lzma443.tar.bz2".
- Also under this section click on "squasfs-lzma patches", and download this file. At this current date the file name is "sqlzma3.2-r2.tar.bz2". (*The file name will have the Squash FS (File System) version, that this patch will work with.*)

2.3. Obtaining the Squash File System code.

2.3.1. Downloading SquashFS

It is mandatory to download the SquashFS version that matches the LZMA patch version.

- For the version of SquashFS that matches the LZMA compression patch, visit the SquashFS web site at the following URL: <http://squashfs.sourceforge.net/>
- Click on the latest version of SquashFS, or just a bit lower (forth paragraph) you can click on "download page"
- For this document the LZMA patch downloaded was "sqlzma3.2-r2.tar.bz2", therefore we will download "squashfs3.2-r2.tar.gz".

2.3.2. **Determining which Linux kernel that SquashFS can patch**

- From the console prompt, run the following command "cd ; tar zvft squashfs3.2-r2.tar.gz", assuming that all files are being downloaded to your home directory.
- Find the latest version of the Linux kernel that SquashFS can patch. Example:
drwxr-xr-x phillip/500 0 2007-01-15 21:29:51 squashfs3.2-r2/kernel-patches/linux-2.6.20/

This indicates that we can use Linux kernel 2.6.20.

2.4. **Obtaining the latest Linux kernel that SquashFS can patch**

- Using a web browser go to the following URL: <http://www.kernel.org/pub/linux/kernel/v2.6/>
- A list of files should show up on the screen. Scroll down to the latest kernel version which can be used with SquashFS. For the purpose of this document the file "linux-2.6.20.tar.bz2" was downloaded. (The file size was 41 Megs in size)

2.5. **Obtaining the Linux-Live scripts**

- Using a web browser go to the following URL: <http://www.linux-live.org/>
- Click on "download"
- Then on the first link under section 1, click on the latest Linux Live link, and save the file "linux-live-6.0.7.tar.gz", to your home directory. (Home directory should be the default location)
- Then scrolls down to the bottom of the page. The last link should be a kernel configuration file, which has many hardware features enabled by default. This would be ideal for making a LiveCD, as you want this CD to work in almost any PC. Click on the ".config" link, then save this file. If you have a GUI Web browser, then you can <right click> on the .config link, and select "save as".

2.6. Obtaining Another Union File System (AUFs)

AUFs doesn't come in a tar ball or rpm, so we must use CVS to obtain it.

- From the command prompt type the following:

```
"cvs -d:pserver:anonymous@auf
```

s.cvs.sourceforge.net:/cvsroot/aufs login"
Then press ENTER

- At the password prompt, simply press **ENTER** leaving it blank.

A cvs login warning may appear as a file in your home directory may not be present. IGNORE THIS WARNING.

- Once CVS returns you to the prompt, type in the following:

```
"cvs -z3 -d:pserver:anonymous@auf
```

s.cvs.sourceforge.net:/cvsroot/aufs co aufs"
Then press **ENTER**

- This will create a directory called aufs, and download all files related to this project into this directory.
- Once CVS completes and returns you to the prompt, we can proceed by creating an AUFs tar ball with the following command: "**tar zcf aufs.tgz aufs**" then pressing **ENTER**.
- The AUFs directory can now be deleted with the following command: "**rm -rf aufs**", as we now have a tar ball.

3. Installing SlackWare Linux for the LiveCD

3.1. Creating the LiveCD environment

- Any directory can be used to create your LiveCD into. For this document the directory /live will be used. Create this directory with the following command: "**mkdir /live**"

3.2. Mounting the SLackWare CD

- Many packages have dependencies to libraries, so expanding beyond what is documented could prove to be a trial and error practice.
- Insert the SlackWare CD1 into your CD-ROM drive, then mount the CD with the following command: "**mount -t iso9660 /dev/cdrom /mnt**"

If the above command does not work, then your CD-ROM isn't properly linked to the device cdrom. In this case you can run the following command to see what device your CD-ROM is associated with: "dmesg | grep -i cd"

Look for lines something like this:

hdc: SAMSUNG DVD-ROM SD-616E, ATAPI CD/DVD-ROM drive

hdd: HL-DT-ST GCE-8483B, ATAPI CD/DVD-ROM drive

Example: If the CD is in the SAMSUNG drive, then the following command would mount the CD: "mount -t iso9660 /dev/hdc /mnt" note that the device hdc was used.

3.3. Installing the base SlackWare packages

Now that the directory is created and the CD is mounted, we can continue to install the base packages of the SlackWare distribution.

- Change directory to the slackware CD, by issuing the following command: **"cd /mnt"**

To look into this directory we can type the "ls" command.

- SlackWare stores all its packages in the directory called Slackware, so we will change directory in this directory with the following command: **"cd slackware"**

- The base packages are all stored in the "a" directory, which is what we will install. Change directory into this directory with the following command: **"cd a"**

- To install all these packages into our LiveCD environment, type the following command: **"installpkg -root /live *.tgz"**

If you are running a distribution that does not have the installpkg command you can always do the following command:

"cd /mnt/slackware/a ; for I in *.tgz; do tar zxvf \$I -C /live; done"

3.4. Un-mounting the SlackWare CD

- To un-mount the CD you must first leave the directory where the CD-ROM is mounted. Typing **"cd ~"** will return you to your home directory.

- Un-mount the CD with the following command: **"umount /mnt"**

4. Compiling the kernel system wide

As the Linux Live CD will utilise Aufs and SquashFS+lzma as modules, we have to boot into the kernel version that the Linux Live CD will use, in order to compile these modules properly.

Check your current Linux kernel version by typing **"uname -r"**. For this document Linux kernel 2.6.20 was downloaded (section 2.4) and will be used for our Linux Live CD.

If "uname -r" reports a different kernel version, then you will have to compile the 2.6.20 kernel system wide.

It is critical at this time to ensure that you will be booting in the same kernel version as the Linux Live CD will be running.

The steps for compiling a new kernel will not be addressed with this documentation, as some Linux distributions vary in steps. There are many "Kernel Compiling HOWTOs" on the internet for many different Linux distributions.

5. Compiling the Kernel for the Live CD

5.1. Setting up the temporary directory

- Create a directory where we can build the kernel that the LiveCD will use. For this document the directory `/tmp/livecd` will be used. To create this directory type in the following command **"mkdir /tmp/livecd"**.

- Then copy all the files downloaded in section 2 into this newly created directory. Example:

```
cp ~/aufs.tgz /tmp/livecd
cp ~/linux--2.6.20.tar.bz2 /tmp/livecd
cp ~/lzma443.tar.bz2 /tmp/livecd
cp ~/sqlzma3.2-r2.tar.bz2 /tmp/livecd
cp ~/squashfs3.2-r2.tar.gz /tmp/livecd
```

- Once all files are copied we can change directory to our temporary build livecd directory. Type **"cd /tmp/livecd"** to change to this directory.

5.2. Extracting all files needed for the kernel compilation and patching

- Decompress the LZMA SDK by first creating a directory for it to decompress into. This can be done by typing **"mkdir lzma443"**. Then by decompressing it with the following command: **"tar -C lzma443 -jxf lzma443.tar.bz2"**

- Decompress the AUFS package by typing the following: **"tar zxf aufs.tgz"**

- Decompress the SquashFS package by typing the following:
"tar zxf squashfs3.2-r2.tar.gz"

- Decompress the LZMA SquasFS patches by first creating a directory for it to decompress into. This can be done by typing **"mkdir sqlzma-patch"**. Then by decompressing it with the following command: **"tar -C sqlzma-patch -jxf sqlzma3.2-r2.tar.bz2"**

- Then finally decompressing the kernel with the following command:
"tar jxf linux-2.6.20.tar.bz2"

5.3. Patching the kernel

- Change into the directory of the kernel by typing the following command:
"cd linux-2.6.20"

- Patch the kernels AUFS by typing the following two commands:

```
"patch -p0 < ../aufs/ksize.patch"
```

```
"patch -p0 < ../aufs/lhash.patch"
```

- Patch the kernels SquashFS utilizing the patch which is located in the directory of the kernel version used by typing the following command:

```
"patch -p1 < ../squashfs3.2-r2/kernel-patches/linux-2.6.20/squashfs3.2-patch"
```

For this document, the Linux kernel 2.6.20 was used, therefore the patch within the directory of 2.6.20 was used.

- Since the LZMA patches affect both LZMA and the kernel, it is time to get out of the kernels directory by typing the following to go back one directory: `"cd .."`
- Now for the LZMA patches, which are the last set of patches to apply. At the prompt type in the following 3 patch commands one at a time:

```
"patch -p0 < sqlzma-patch/sqlzma1-443.patch"  
"patch -p0 < sqlzma-patch/sqlzma2k-3.2-r2.patch"  
"patch -p0 < sqlzma-patch/sqlzma2u-3.2-r2.patch"
```

5.4. Compiling the kernel

- Enter the Linux kernel directory by typing the following command: `"cd linux-2.6.20"`. If you used a different kernel version, ensure you change the 2.6.20 to your kernel version.

IF you will be configuring your own kernel configuration, ensure that SQUASHFS is set to NO!

- To use the .config file that was downloaded in section 2.5, simply copy this .config file into the current directory. If the file was downloaded in your home directory, which is the default location, type the following command:

```
"cp ~/.config ."
```

- Then run the following command to configure the kernel to use an old configuration file:

```
"make oldconfig"
```

- Then to make the kernel prepare some scripts with the following command:

```
"make scripts"
```

- Now to prepare the kernel before compiling, with the following command:

```
"make prepare"
```

The kernel is now ready to be compiled.

- We can start by compiling the modules with the following command: `"make modules"`
- Then install the modules in the LiveCD environment with the following command:

```
"INSTALL_MOD_PATH=/live make modules_install"
```
- Now finally to compile the kernel, with the following command: `"make bzImage"`

6. Compiling External Modules (AUFS and SquashFS + LZMA)

- Time to go back to the temporary directory where all these modules were extracted in section 5.1. Type the following command: **"cd /tmp/livecd"**

6.1. Compiling and installing AUFS

- Enter the AUFS directory with the following command: **"cd aufs"**.

Two methods of modifying the Makefile will be documented here, as not everyone is familiar with long Linux commands. It is best to understand what the changes are.

1. One command method

- At the prompt, type this long winded command in one line:

```
cat local.mk | grep -v "^CONFIG_AUFS_DEBUG =" | grep -v  
"^CONFIG_AUFS_COMPAT =" | sed 's/KDIR = /KDIR = \live/' > Makefile
```

(Note that this command does NOT start with " and does not end with ", as this may confuse people, as this command has many of these characters already.)

2. The manual method

- Copy the provided makefile to the default makefile name with the following command: **"cp local.mk Makefile"**

With your favourite text editor, open this file, then do the following:

- At line 19, there will be a line that looks like this: **"CONFIG_AUFS_DEBUG = y"**. DELETE this entire line.
- At line 21, there will be a line that looks like this: **"CONFIG_AUFS_COMPAT ="**. DELETE this entire line.
- At line 54, there will be a line that looks like this:
"DIR = /lib/modules/\$(shell uname -r)/build"
Change this line to look like this:
"DIR = /live/lib/modules/\$(shell uname -r)/build"

Notice that /live was simply inserted at the beginning of the path, to indicate the LiveCD environment.

- Compile AUFS by typing the following command: **"make"**
- Finally copy the module to the current system environment by first creating the AUFS module directory by typing **"mkdir -p /lib/modules/2.6.20/kernel/fs/aufs"**, then copying the module in this directory with the following command:
"cp aufs.ko /lib/modules/2.6.20/kernel/fs/aufs".

6.2. Compiling and installing SquashFS with LZMA

• The sqlzma patches will compile everything needed. Since this will be accessing lzma, squashfs, and the Linux kernel, we will first start by moving the patch to the temporary environment. This can be done with the following commands:

- First to ensure we are at the right place type "**cd /tmp/livecd/sqlzma-patch**".
- Then copy all files to the temporary environment with: "**cp * ..**".
- Then change directory to the temporary environment with "**cd ..**".

There are again two methods in modifying the Makefile.

1. The one liner method

At the prompt type the following two commands:

(Note that the first and last quotes have been removed from this command to not confuse anyone with extra quotes)

At the prompt, type the following in one line:

```
cat Makefile | sed 's/^SqFs = ./SqFs = fs\squashfs/' | sed 's/^KDir = /KDir = \live/' > Makefile2
```

Followed by this command:

```
mv Makefile2 Makefile
```

2. The Manual method

Using your favourite text editor, open the file "Makefile".

At line number 18, there will be a line that looks like this:

```
"SqFs = ${Squashfs}/kernel-patches/${KVer}/fs/squashfs"
```

Change this line to this:

```
"SqFs = fs/squashfs"
```

At line number 19, there will be a line that looks like this:

```
"KDir = /lib/modules/$(shell uname -r)/build"
```

Change this line to this:

```
"KDir = /live/lib/modules/$(shell uname -r)/build"
```

Then save the file.

• We can now proceed to compile SquashFS with LZMA compression, by typing the following command: "**make**"

• Now to copy the commands mksquashfs and unsquashfs to the LiveCD environment, by typing: "**cp squashfs3.2-r2/squashfs-tools/{mk,un}squashfs /live/usr/bin**"

• Create the directory where the squashfs module will reside on the current system with the following command: "**mkdir -p /lib/modules/2.6.20/kernel/fs/squashfs**"

- Copy the LZMA modules in the directory that was just created by doing the following one line command:

```
cp lzma443/C/7zip/Compress/LZMA_C/kmod/{sq,un}lzma.ko /lib/modules/2.6.20/kernel/fs/squashfs
```

- Then copy the squashfs module in the same directory with the following:
cp linux-2.6.20/fs/squashfs/squashfs.ko /lib/modules/2.6.20/kernel/fs/squashfs

7. Cleaning-up the LiveCD environment

- Since we are using a newer kernel than what Slackware came with by default, we can remove the old modules that was installed in section 3, with the following command:

```
"rm -rf /live/lib/modules/2.4.33"
```

- Then remove all the boot files that were installed by Slackware, as these are for the 2.4.33 kernel as well. Remove these files with the following command:

```
"rm -rf /live/boot/*"
```

- Then remove the two symbolic links in /live/lib/modules/2.6.20, as they will be broken once the LiveCD is booted.

```
"rm /live/lib/modules/2.6.20/build"
```

```
"rm /live/lib/modules/2.6.20/source"
```

8. Installing the Kernel in the LiveCD Environment

- Time to go back to the temporary directory where the Linux kernel code resides, with the following command: **"cd /tm/livecd/linux-2.6.20"**

- Copy the kernel config file to the LiveCD Environment with the following command:
"cp .config /live/boot"

- Copy the system map file to the LiveCD environment with the following:
"cp System.map /live/boot"

- Copy the compiled kernel in the LiveCD environment with the following:
"cp arch/i386/boot/bzImage /live/boot/vmlinuz"

9. Creating the LiveCD Image File

- Return to your home directory with the following command: **"cd ~"**.

- It is always a good practice to link all the shared libraries in the LiveCD environment before every build. This can be done by executing the following command:
"chroot /live /sbin/ldconfig".

- Extract the Linux-Live scripts, which was downloaded in section 2.5 in the home directory with the following command: **"tar xzf linux-live-6.0.7.tar.gz"**

- Enter the directory where the Linux Live scripts are located by typing:
"cd linux-live-6.0.7"

- Open the file ".config" with your favorite text editor.

At line 9, there will be a line that starts with:

LIVECDNAME=

Change this line to what ever name you want to call your LiveCD. example:

LIVECDNAME="foobar"

At line 29, there will be a line that starts with:

ROOT=

Change this line to:

ROOT=/live

- Then save the file
- To build your LiveCD run the command **"/build"**
- This will now create a directory `live_data_####` in your `/tmp` directory. The exact number will be displayed when the build script completed. Example: `"live_data_6870"`
- Change directory to this folder. example: **"cd /tmp/live_data_????"**
- Type `"ls"` and you will notice two directories. Change directory in the directory of the LIVECDNAME that was specified earlier. Example: `"cd foobar"`
- Then to create the .ISO file, run the command: **"/make_iso.sh"**
- The ISO image file will be located in `/tmp` with the name of the LIVECDNAME specified earlier. Example: `/tmp/foobar.iso`
- Use your favourite CD Writting/Recording software and burn this ISO to a CD.
- Boot up any system with this CD in it's drive, and play around with your very own Linux Live CD.

Play around, add more packages, make a LiveCD that is really cool that no one else has done, and share it with everyone in the Linux Community.