

# Research report on creating a bootable Live-CD

Bal Krishna Bal, Basanta Shrestha, Subir B. Pradhananag, Paras Pradhan

*Madan Puraskar Pustakalaya*

[bal@mpp.org.np](mailto:bal@mpp.org.np), [basanta@mpp.org.np](mailto:basanta@mpp.org.np), [subir@mpp.org.np](mailto:subir@mpp.org.np),

[paras@mpp.org.np](mailto:paras@mpp.org.np)

## Abstract

*This document is focused on the development and customization of the Linux based Live-CDs. It begins with the definition of a Live-CD followed by elaborated procedures and requirements for creating a Live-CD. It should be noted that this document has been released under GNU Free Documentation License (<http://www.gnu.org/copyleft/fdl.html>).*

## 1. Introduction

With the advancement of new technology, computer end-users have today become more and more demanding and sophisticated. The development of a bootable Live-CD, thus avails the end-users with various options, like for example, installing the operating system only if it is impressive and effective enough in the eyes of the end-users or running the operating system from the Live-CD itself, thus not necessarily installing on the hard disk.

There exist certain steps and procedures for creating a bootable Live-CD for several Linux Distributions. This report would basically deal with the procedures that need to be undertaken for creating one such CD for the Nepalinux, a localized version (GNU/Linux) of the Debian Distribution in Nepali.

## 2. Methods

The methodology employed for the purpose of the study is purely a research-oriented one. The standard procedures for creating a bootable Live-CD were studied from different sources. Later, the further requirements for the customization and development of such a CD for Nepalinux were explored and the findings implemented.

## 3. Discussion

### 3.1. What is a Live-CD?

A Live-CD is an operating system (usually containing other software as well) stored on a bootable CD-ROM or DVD-ROM that can execute the operating system, without the installation of the operating system on a hard drive. The most popular Live-CD Linux Distributions are: Knoppix, Morphix, Gnoppix, Ubuntu etc.

### 3.2. Working with the Base System

The foremost thing required for creating a bootable Live-CD would be downloading the Base system from [http://www.morphix.org/index.php?option=com\\_content&task=view&id=37&Itemid=59](http://www.morphix.org/index.php?option=com_content&task=view&id=37&Itemid=59). The file is an .iso archive and has the name similar to MorphixBase-0.5-pre5.iso. You then need to extract the iso to some folder say /iso:

- mkdir /iso
- -mount -o loop MorphixBase-0.5-pre5.iso /iso
- All the files and folders should be copied to some directory say /mods:
  
- mkdir /mods
- cp -Rp /iso/\* /mods

### 3.3. Installing Required Software

A few software need to be installed for working with compressed images like cloop utils, cloop modules and cowloop modules. These utilities and modules could be installed from any Debian repository using the following commands:

- apt-get install cloop-module
- apt-get install cloop-src
- apt-get install cloop-utils
- -apt-get install cowloop-module

### 3.4. Creating the Core Debian Image

The package `debootstrap` is used to create a Debian base system from scratch, without requiring the availability of `dpkg` or `apt`. It does this by downloading `.deb` files from a mirror site, and carefully unpacking them into a directory which can eventually be chrooted into.

For installing the tool `debootstrap`, the following command should be run:

- `apt-get install debootstrap`

Once you have installed `debootstrap`, download the Debian Core System using `debootstrap`

- `mkdir /sid-base`
- `debootstrap sid /sid-base`

You may use `sarge` (Stable), `etch` (Testing) and `sid` (Unstable) with `debootstrap`. If you want to work with the latest packages, you need to use `sid`. When `debootstrap` is complete, change the directory to `/sid-base`. It will look like installation of Linux on the hard disk. After that, you may add packages like: `Gnome`, `KDE`, and `Apache` whatever you would want in your Live-CD.

### 3.5. Packages for the Live-CD

After `debootstrap`ing the Debian Core System, you may now `chroot` to the folder and add packages.

For adding `Gnome Desktop`:

- `'chroot /sid-base'`
- Add `"deb ftp://ftp.debian.org/debian/ sid main contrib"` in `/etc/apt/sources.list` file
- `'apt-get update'`
- `'apt-get install gnome-desktop-environment'`
- `'updatedb'`

Opt 'Y'(Yes) for the dependencies.

Following the same procedures as above for the `gnome-desktop-environment`, you may install other packages.

### 3.6. Locales and Live-CD

Locale is needed in order to get the User interface in the native language.

For Locale installation:

- `'chroot /sid-base'`
- Copy your locale definition file to `/usr/share/i18n/locales`
- Add a line for your `lang_country` name (eg: `ne_NP UTF-8`) in `/etc/locale.gen` file
- Run `'locale-gen'`
- Verify using `'locale -a'`
- `exit`

### 3.7. Adding up Translations

Translated files in the form of `.mo` are needed for the strings to be displayed at the User interface in native language.

For adding Translation files of `Gnome`:

- `'chroot /sid-base'`
- Copy the translations in the form of `.mo` to `/usr/share/locale/ne/LC_MESSAGES`
- `'exit'`

### 3.8. Language

For modifying the base system of the Live-CD to default boot into your desired language:

Open the file `/mods/boot/grub/lang.lst` and add language (`lang`) anywhere at the kernel path line. For example:

```
kernel(cd)/boot/vmlinuz lang=ne ramdisk_size=100000
init=/etc/init noapic acpi=off apm=power-off vga=791
splash=silent          initrd=miniroot.gz          quiet
BOOT_IMAGE=morphix
```

Note `lang=ne` above. This will boot the Live-CD to the desired language, in the above case, `Nepali`.

### 3.9. Input Systems

For installing Input method system and adding input tables:

There are many input method systems for Linux as: `scim`, `iiimf`, `.xkb` etc. Below are the steps if you would

be using scim input method. Refer to <http://scim-im.org> for more details.

- 'chroot /sid-base'
- 'apt-get install scim'
- 'apt-get install scim-gtk2-immodule'
- Copy the generated input tables at /usr/share/scim/tables/
- Copy the icon to be shown for input system at /usr/share/scim/icons' in the form of png
- exit

### 3.10. Fonts

In order to display the translated string at the Gnome Desktop you need to have OpenType fonts installed in your operating system..

For adding up Fonts:

- Copy your open type fonts at /usr/share/fonts/truetype
- Run 'fc-cache -f -v'
- Verify using 'fc-list'

### 3.11. Grub Splash Screen Modification

For customizing GRUB startup splash screen:

- Create a new directory
- 'mkdir /message'
- 'cd /message'
- Extract the file message using 'cpio -i < /mods/boot/grub/message'
- Edit background.pcx file using some image editor. Gimp is recommended over other applications for this. Apart from this file, files like "help-en", "lang", "language" and "texts.en" can also be customized.
- Archive it back using command 'ls . | cpio -o > message'
- Replace the old "/mods/boot/grub/message" file with this newly created one using: 'cp /message/message /mods/boot/grub/'

Note: The size of the background.pcx file should only be approx of 35KB.

### 3.12. Bootsplash Images

For customization of the boot splash images of the Live-CD seen during the boot process:

- Copy /mods/boot/miniroot.gz file to a temporary directory using 'cp /mods/boot/miniroot.gz /tmp'
- cd /tmp
- Create another temporary directory using 'mkdir /isomount'
- Uncompress miniroot.gz file using 'gunzip miniroot.gz '
- Mount the file to /isomount directory using 'mount -o loop miniroot /isomount'
- cd /isomount
- Edit the splash images of the folder /isomount/bootsplash/images. Gimp Image Editor is also recommended for this as well.
- Be careful of the images size. The approximate size is around 35 KB only.
- Unmount the folder using 'umount /isomount'
- Compress the miniroot file using gzip /tmp/miniroot
- Finally replace the old miniroot.gz with the new one using 'cp /tmp/miniroot.gz /mods/boot/'

### 3.13. Necessary Configurations

For cleaning unnecessary packages from the base directory to save disc space:

- 'apt-get clean'
- For making necessary directories:
- 'mkdir MorphixCd cdrom1 morphix floppy cdrom'

Then download the following files

- a) loadmod.sh
- b) init.sh
- c) main\_module

from <http://www.nepalinux.org/livecd/> and copy them to morphix folder.

After adding packages and doing necessary configuration exit the chroot environment

- 'exit'

### 3.14. Main Module Generation

Main module is the main file image which will be loaded during the Live-CD boot up. This contains the Core Debian System and other packages.

For generating the main module:

We will now need to convert the base system i.e /sid-base into a compress ext3 main module named gnome210.mod. The package used is morphix-modulebuilder. Install it following the below steps

- Edit /etc/apt/sources.list and add 'deb <http://www.morphix.org/debian> .' at the last line
- 'apt-get update'
- 'apt-get install morphing-tools'
- 'apt-get install morphix-modulebuilder'
- Press 'Y' to install the dependencies

Generate the main module by the following command. This may take a lot of time depending on the size of the base system and on the processor of the system

- `module-builder -t ext3 /sid-base /mods/mainmod/gnome210.mod`

### 3.15. Live-CD ISO

For creating the iso for your Live-CD:

- `'mkisofs -pad -l -r -J -v -V "CD name" -b boot/grub/iso9660_stage1_5 -c base/boot.cat -no-emul-boot -boot-load-size 4 -boot-info-table -hide -rr -moved -o /tmp/mycd.iso /mods'`

### 3.16. Burning CD Image

For burning mycd.iso in blank CD or CDRW:

- `'cdrecord -dev=0,0,0 mycd.iso'`
- `' cdrecord -dev=0,0,0 blank=fast' ( Use this to blank the CDRW )`

Here dev=0, 0, 0 is used using the output of the command of 'cdrecord --scanbus'. You can use any CD recording tool to burn the iso like: k3b, nautilus CD burner etc.

## 4. Results

Following the above procedures and steps as mentioned in the discussion section, a bootable Live-CD was created for Nepalinux. The created Live-CD was tested in several machines both Desktop and Laptop PCs and the result was a success.

## 5. Conclusion

While the objective of creating a bootable Live-CD has been met, further testing of the product in a number of machines of different hardware configurations resulting into a list of minimum hardware and system specification could be a major task in future for the bootable Live-CD package.

This document is believed to serve as a resource paper for the researchers and other interested people in creating a bootable Live-CD for their own versions of GNU/Linux.

## 6. References

For preparing this document the following links and documents were referred to.

[1] <http://www.wikipedia.org>

[2] "The World of Morphix" <http://www.morphix.org>

[3] "Debian - The Universal Operating System" <http://www.debian.org>

[4] Linux Man pages