LADISLAV HAGARA[1]

SOURCE MAGE GNU/LINUX LIVE AND KICKING

Source Mage GNU/Linux is a source-based GNU/Linux distribution. It allows users the maximum amount of freedom and choice possible. Any user can build his/her own unique systems. Source Mage systems can be optimized for specific hardware and with specific options. It can contain only necessary software. Unlike common binary distributions Source Mage is really „up-to-date”.

Live System is an operating system (usually containing other software as well) stored on a bootable medium (CD, DVD or USB) that can be run directly from this medium, without installing into permanent memory, such as a hard drive. Users can work with their own Live Systems everywhere.

Source Mage GNU/Linux Live System Generator is a program (set of scripts) which provides a quick and easy way of creating Live Systems based on Source Mage GNU/Linux. Users can create and optimize their own Live Systems for own needs.

Live System

Live System is an operating system (usually containing other software as well) stored on a bootable medium (CD, DVD or USB Flash disk) that can be run directly from this medium, without installing into permanent memory, such as a hard drive. User does not need to solve problems with the installation, because completally functional system is located on the given medium. Live System does not alter the current operating system or files without a user's doing. The system returns to its previous OS state when the computer is rebooted and the medium is ejected. Live System does this by placing the files which typically would be stored on a hard drive, into temporary memory, such as a ram disk. In fact, a hard drive is not needed at all. Live System can be used as a rescue system or special system (firewall, antivirus, penetration testing ...).

[1] Department of Communication and Information Systems (K–209), Faculty of Military Technology, University of Defence, Kounicova 65, 612 00 Brno, Czech Republic. E-mail: ladislav.hagara@unob.cz.

Linux Live System

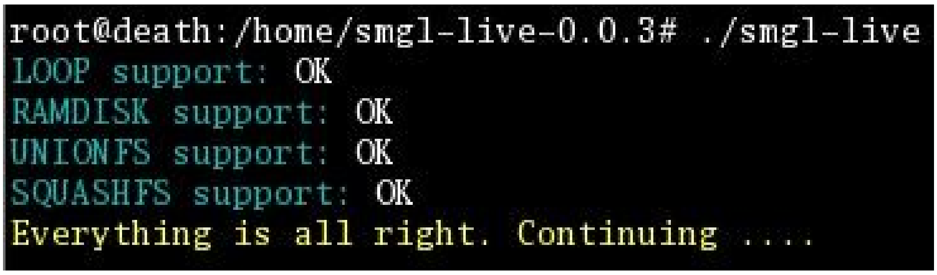
Some Linux Live Systems are designed to demonstrate the power of Linux to unsuspecting users of other operating systems, while others are highly specialised projects useful even to experienced Linux gurus. They range from emergency rescue and system diagnostics CDs to multimedia oriented projects that transform a diskless or OS-less computer into a full home theatre. Each of these projects can have a unique feature, a merit and a niche to fill.

The primary function of Linux Live System can be:

- Desktops: provides a working GUI desktop environment with a collection of desktop programs, such as browsers and text editors. Many also include utilities for other purposes, such as home entertainment, but are only listed here because the additional functions are not their primary focus;
- OS Replacement: provides an option to transfer the cd to the hard drive, or to install an OS in a different form;
- Education: provides a collection of educational programs, or was created to be used in the educational field;
- Rescue: provides tools needed for data recovery;
- Clustering: provides tools for making clusters;
- Security: contains network security tools;
- Home Entertainment: geared towards playing video and audio;
- Gaming: video games;
- Medical: contains medical programs;
- Diagnostics: contains utilities for testing hardware;
- Firewalls: distributions created to be used as firewalls;
- Forensics: distributions containing forensic tools;
- Servers: distributions used for various server functions.

Linux Live System internals

The most important software technologies used in Linux Live Systems are Loop device [7], Ramdisk [8], UnionFS [9] and SquashFS [11]. UnionFS and SquashFS are not included in the mainline Linux kernel (vanilla kernel).



```
root@death:/home/smgl-live-0.0.3# ./smgl-live
LOOP support: OK
RAMDISK support: OK
UNIONFS support: OK
SQUASHFS support: OK
Everything is all right. Continuing ....
```

Source Mage GNU/Linux Live System Generator: Primary test

Loop device

In Unix-like operating systems, a loop device or loopback device is a device node that represents a regular file. The same names are also used for the device drivers that control the devices. Loop devices are mostly used for being mounted on a directory, which produces the same effect of mounting a disk whose image is identical to the file associated to the device on that directory.

Mounting a file on a directory involves two steps:

1. The file is associated with a loop device node using a specific command (losetup in Linux, for example);
2. Then, the device node is mounted on the directory as for any other block device. For example, if example.img is a regular file and /home/you/dir is a directory on a Linux box, the root user can mount the file on the directory by executing the following two commands:
   - ➢ losetup /dev/loop0 example.img;
   - ➢ mount /dev/loop0 /home/you/dir.

The first command associates the loop device node /dev/loop0 with the regular file example.img. This association can be later destroyed by executing losetup -d /dev/loop0. The second command mounts the device on the directory /home/you/dir.

The global effect of executing these two commands is that the content of the file is used for storing the whole mounted directory.

Ramdisk

A ramdisk is a virtual solid state disk that uses a segment of active computer memory, RAM, as secondary storage, a role typically filled by hard drives. Access times are greatly improved, because RAM is approximately a hundred times faster than hard drives. However, the volatility of RAM means that data will be lost if power is lost, e.g. when the computer is turned off. Ramdisks can be used to store temporary data or hold uncompressed programs for short periods.

A proper disk cache in the operating system will usually obviate the performance motivation for a ramdisk; a disk cache fulfills a similar role (fast access to data that is notionally stored on a disk) without the various penalties (data loss in the event of power loss, static partitioning etc.). Ramdisks are, however, indispensable in situations in which a physical disk is not available, or where access to, or changing a physical disk is not desirable (such as in the case of Live CDs). They can also be used in a kiosk-style device where any changes made to a system are not committed and the original configuration is to be loaded each time the computer is turned on.

Another way to use RAM to store files is the temporary filesystem. The difference between temporary filesystem and a ramdisk is that the ramdisk (/dev/ram0 etc.) is fixed-sized and acts like a disk partition, whereas the temporary filesystem (/dev/shm; in Source Mage GNU/ Linux often /tmp) grows and shrinks to fit the files put on it.

Ramdisks have the advantage of being much faster than hard drives and only require special software (and of course the computer's RAM). Their disadvantage is that they are limited to main memory and data is lost on loss of power unless other measures (such as battery backup) are used.

UnionFS

UnionFS (A Stackable Unification File System) a stackable unification file system, which can appear to merge the contents of several directories (branches), while keeping their physical content separate. Unionfs is useful for unified source tree management, merged contents of split CD-ROM, merged separate software package directories, data grids, and more. Unionfs allows any mix of read-only and read-write branches, as well as insertion and deletion of branches anywhere in the fan-out. To maintain unix semantics, Unionfs handles elimination of duplicates, partial-error conditions, and more.

There are identified three primary use cases for UnionFS. The first and most prevalent use of UnionFS is in LiveCDs. The second is using

UnionFS to provide a common base for several NFS-mounted machines. The third is to use UnionFS for snapshotting.

1. LiveCDs

LiveCDs allow users to boot Linux without modifying any data on a hard disk. This has several advantages:

- Users can try Linux without committing to it.
- Special-purpose open-source software can be distributed to non-technical users (e.g., for music composition).
- System administrators can rescue machines more easily.
- Many similar machines can be set up without installing software (e.g., in a cluster environment, or at events that require certain software).

2. NFS-mounted machines

Another use of UnionFS is to simplify the administration of diskless machines. A set of machines can share a single read-only NFS root file system. This enables administrators to maintain a common image for all of the machines. This root file system is then unified with a higher-priority read-write branch so that users can customize the machine or save data. If persistence is not required, then a tmpfs file system can be used as the highest priority branch. If persistence is required, then a read-write NFS mount or a local disk could be used for the user's files.

3. Snapshotting

The previous usage scenarios all assumed that one or more components of the union were read-only by necessity (either enforced by hardware limitations or the NFS server). UnionFS can also provide copy-on-write semantics by logically marking a physically read-write branch as read-only. This enables UnionFS to be used for file system snapshots. To create a snapshot, the unionctl tool is used to invoke branch management ioctls that dynamically modify the union without unmounting and remounting UnionFS. First, unionctl is used to add a new high-priority branch. Next, unionctl is called for each existing branch to mark them as read-only. Any changes made to the file system take place only in the read-write branch. Because the read-write branch has a higher priority than all the other branches, users see the updated contents.

SquashFS

When creating tiny-sized and embedded Linux systems, every byte of the storage device (floppy, flash disk, etc.) is very important, so compression is used everywhere possible. Also, compressed file systems are frequently needed for archiving purposes. For huge public archives, as well as for personal media archives, this is essential.

SquashFS brings all this to a new level. SquashFS is a squashed read-only filesystem for Linux. It lets you compress whole file systems or single directories, write them to other devices/partitions or to ordinary files, and then mount them directly (if a device) or using a loopback device (if it is a file). The modular, compact system design of SquashFS is bliss. For archiving purposes, SquashFS gives you a lot more flexibility and performance speed than a .tar.gz archive.

SquashFS is distributed as a Linux kernel source patch (which enables SquashFS read support in your kernel), and the mksquashfs tool, which creates squashed file systems (in a file or on a block device).

Linux Live CD/USB scripts

Linux Live CD/USB scripts [13] is a set of shell scripts which allows you to create own LiveCD from almost every Linux distribution. Just install your favourite distro, remove all unnecessary files (for example man pages and all other files which are not important for you) and then download and run these scripts to build your custom Live Linux.

It doesn't work with old 2.4 kernels, only with 2.6 ones. Unpack the archive to /tmp, login as root and run ./runme.sh. You need unionfs and squashfs support for your Linux Kernel (required kernel modules only for 2.6.16 are included in linux live scripts). If you don't like to setup it yourself, download precompiled Linux kernel 2.6.16 with unionfs and squashfs modules included, and with alsa sound drivers and madwifi drivers.

Linux Live Systems based on Source Mage GNU/Linux

Some users of Source Mage GNU/Linux want work with their own systems everywhere. Of course they can not install them on all boxes.

However they can use their own Live Systems. Live Systems based on Source Mage GNU/Linux can satisfy all their needs. Users can use their own Source Mage GNU/Linux everywhere. Of course these Live Systems can be used also by users of other Linux distributions as well as users of other operating systems.

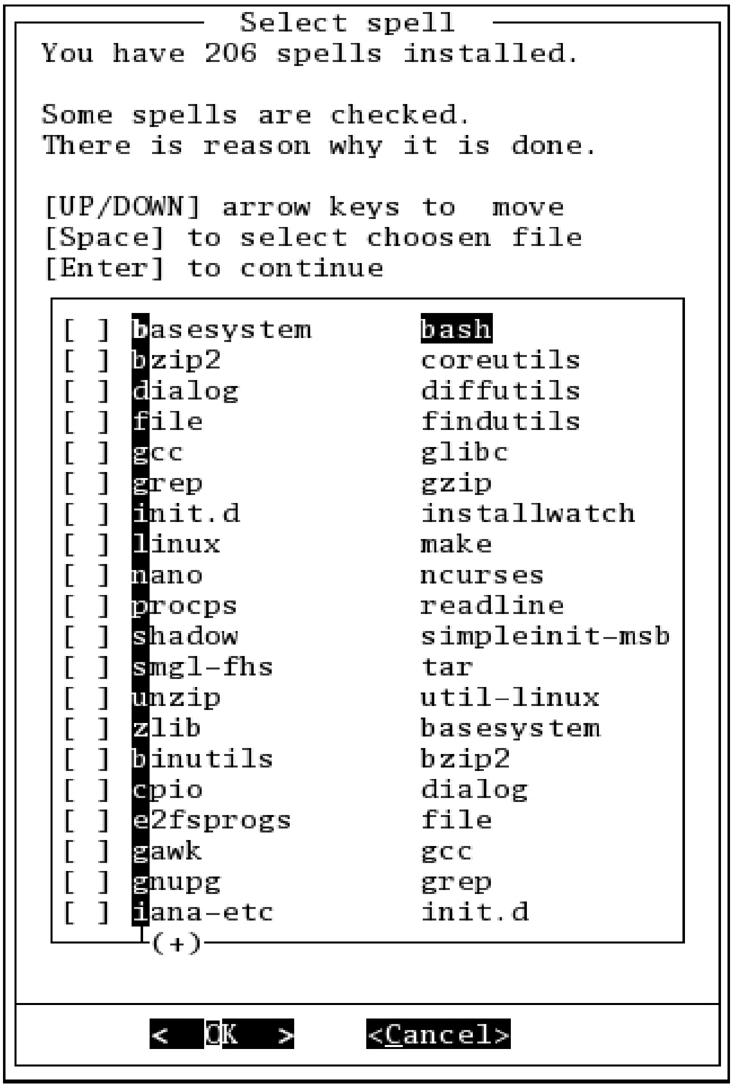
Source Mage GNU/Linux currently includes around 5000 applications (spells). Users can choose which spells will be located on their Live Systems. Their can create their own either general or single purpose systems, demo systems or daily used productive systems.

Software products can be updated almost daily. With Source Mage GNU/Linux distribution it is easy for users to ensure all software products (spells) are installed in their up-to-date versions. Users are able to create their own Live Systems with up-to-date version of their favourite applications. Spells are updated continuously so users can update their Live Systems shortly after the application is published. They do not need to wait several months as the users of binary distributions have to. Binary distributions are updated only 2 as or 3 times per year, Source Mage GNU/Linux even several times a day.

With security related problems it is the same. User does not need to wait as far as a new version of distribution will go out. Users can update their systems by command „sorcery system-update" and burn the new updated version of their Live Systems.

Source Mage GNU/Linux Live System Generator

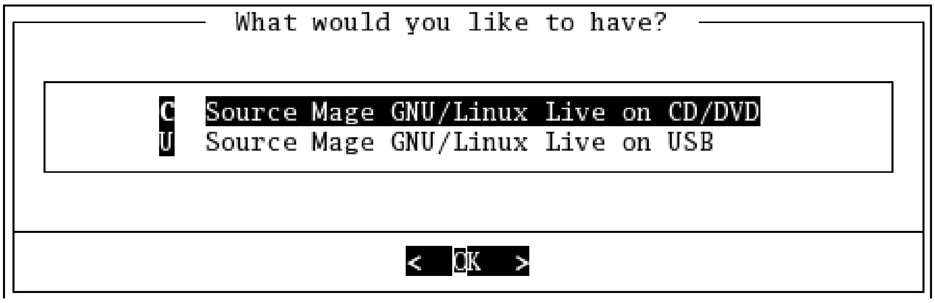
There are several documents on the internet describing the creation of Linux Live Systems. Their disadvantages are their usability only on a particular Linux distribution or their out datedness. It is possible to find also „universal ready-made scripts". Even they claim they allow you to create your own Live System from every Linux distribution it is not true. These scripts and documents alike are useful only for specific distribution or very similar distributions. Some scripts could be really universal, so they could be useful with all distributions. Their problem is that it is impossible to choose what applications (packages) will be located on Live System. Either they copy all Linux system or they do not support all kind of packages (rpm, deb, tgz …). The size of final images of Live Systems can overrun the size of the medium we want to use. For example DVD can hold up to only 4.7 GB of data. These scripts only have inspired and inspirited us to create our own scripts.



```
                    Select spell
 You have 206 spells installed.

 Some spells are checked.
 There is reason why it is done.

 [UP/DOWN] arrow keys to  move
 [Space] to select choosen file
 [Enter] to continue

  [ ] basesystem       bash
  [ ] bzip2            coreutils
  [ ] dialog           diffutils
  [ ] file             findutils
  [ ] gcc              glibc
  [ ] grep             gzip
  [ ] init.d           installwatch
  [ ] linux            make
  [ ] nano             ncurses
  [ ] procps           readline
  [ ] shadow           simpleinit-msb
  [ ] smgl-fhs         tar
  [ ] unzip            util-linux
  [ ] zlib             basesystem
  [ ] binutils         bzip2
  [ ] cpio             dialog
  [ ] e2fsprogs        file
  [ ] gawk             gcc
  [ ] gnupg            grep
  [ ] iana-etc         init.d
          (+)

           <  OK  >     <Cancel>
```

Source Mage GNU/Linux Live System Generator: Selection of spells

Source Mage GNU/Linux Live System Generator is a program (set of scripts) which provides a quick and easy way of creating Live System based on Source Mage GNU/Linux distribution. Users can define list of applications (spells) or only their parts which will be part of Live System. They can choose which files will be copied to Live System medium, specify unique logins and passwords, optimize Live System according their own needs. Source Mage GNU/Linux Live System Generator can produce either ISO image of Live System for CD or DVD or just directory structure used by USB Flash disk.



```
What would you like to have?

C  Source Mage GNU/Linux Live on CD/DVD
U  Source Mage GNU/Linux Live on USB

< OK >
```

Source Mage GNU/Linux Live System Generator: CD/DVD or USB?

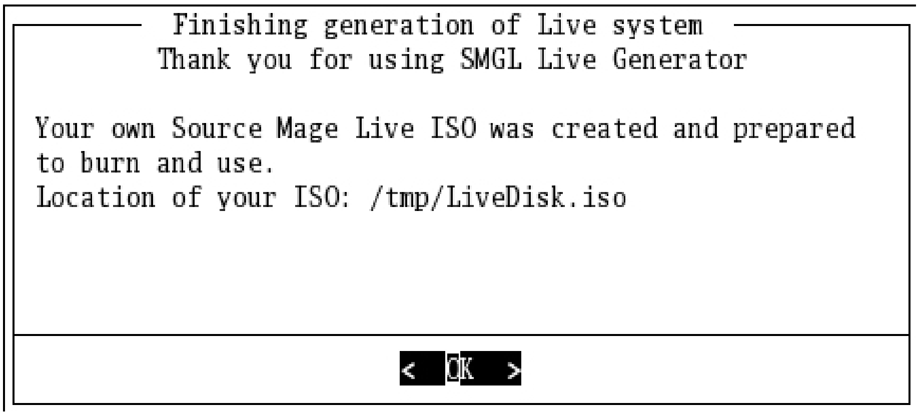
The size of final Live System depends on an amount and size of used applications (spells). It can vary from several MB (only text mode) to several GB (complete graphic environment (X Window System) with thousands of applications).

Live System can be used as a demo allowing users of other Linux distributions or even other operation systems to try out Source Mage GNU/Linux. Users can work with their own systems everywhere. Live System medium can contain crucial files and applications. Users can carry their own Live Systems with their own crypto keys (GnuPG, OpenSSH). Moreover almost all Live System medium can be encrypted. Live System can be used as a rescue system (not only for Source Mage GNU/Linux) or special system (firewall, antivirus, penetration testing ...).

Users can generate Live Systems from scratch or they can generate them according to prepared configurations. Of course users can save and modify their favourite configurations. They are able to choose the files from /root and /home directories which will be copied to Live Systems.

For more information you can visit home page of this project [14]. You can read about the reasons of the developing the Live Systems based on Source Mage GNU/Linux. There is a manual for the generator and additional information. Of course you can download there the latest version of the Source Mage GNU/Linux Live System Generator.

We are working on next version of this generator. Probably we will add the new www interface that makes generation of the Live System even easier. We plan to integrate our generator to standard grimoire of Source Mage GNU/Linux (spell smgl-live) so users can use it without downloading and installing it manually. We plan to integrate our Live System Generator and used Live Systems with our Source Mage Ledger [15]. Users could store their configuration files on Source Mage Ledger and they could be informed for example about security problems in their Live Systems.



```
Finishing generation of Live system
Thank you for using SMGL Live Generator

Your own Source Mage Live ISO was created and prepared
to burn and use.
Location of your ISO: /tmp/LiveDisk.iso
```

Source Mage GNU/Linux Live System Generator: Final report

BIBLIOGRAPHY/REFERENCES

[1] KADERKA, Josef–HAGARA, Ladislav: Introduction to Source Mage GNU/Linux. Proceedings of the conference „New Challenges In The Field Of Military Sciences 2005”. Budapest, 18–19. 10. 2005.

[2] CEPEK, Jan: Linux Live Systems based on Source Mage GNU/Linux. Proceedings of the Students Scientific Conference. University of Defence, Brno, 25–26. 4. 2006.

[3] HAGARA, Ladislav–CEPEK, Jan: Source Mage GNU/Linux Live System Generator. Proceedings of the 1st Military Communications and Information Systems Conference MCC 2006. Gdynia, Poland, 18–19. 9. 2006.

[4] HAGARA, Ladislav: Source Mage GNU/Linux. http://www.root.cz/ serialy/ source-mage-gnulinux/

[5] Source Mage GNU/Linux. http://www.sourcemage.org/

[6] FrozenTech's LiveCD List, http://www.livecdlist.com/
[7] Loop device, http://en.wikipedia.org/wiki/Loop_device
[8] RAM disk, http://en.wikipedia.org/wiki/RAM_drive
[9] UnionFS (A Stackable Unification File System). http://www.filesystems.org/ project-unionfs.html
[10] Unionfs: User- and Community-Oriented Development of a Unification File System. http://www.am-utils.org/docs/sipek-ols2006/ index.html
[11] SquashFS (A squashed read-only filesystem for Linux). http:// squashfs.sourceforge.net/
[12] SquashFS HOWTO, http://tldp.org/HOWTO/SquashFS-HOWTO/
[13] Linux Live CD/USB scripts, http://www.linux-live.org/
[14] Source Mage GNU/Linux Live Generator. http://dcs.unob.cz/~hgr/smgl-live/
[15] Source Mage Ledger. http://ledger.sourcemage.org/
[16] Linux live and kicking, Part 1. http://www.vnunet.com/2045915