

Securing the information of my working mate, my laptop

The simple, efficient and secure backup tool: duplicity

One of the biggest problems in the digital age is the increase of dependency from digital information.

I am more and more using my laptop to store and carry with me important informations. This because carrying a 1,3 kg laptop is less annoying than carrying a pile of paper. And the laptop enables me to do much more than a pile of paper I would be able to carry ;)

But working on a laptop and storing all information in it, to be able to access the information while I am anywhere on this planet, has also some security issues.

One of the worst thing which could happen to a laptop dependent guy like me, would be to loose the laptop, let's say it gets stolen.

In this case I would not just loose the value of the hardware (I use Free Software and restoring my system is a matter of a 20 minutes GNU/Linux installation, so I don't count that), but also the much higher value of the information stored on the hard disk.

The 2 main dangers are:

- 1) There is sensible information stored on my hard disk, I or better who gave it to me wouldn't like that someone else could gain access to it. I solved this problem, using an encrypted folder on my hard disk. I will increase the security by extending the encryption to a stronger encryption algorithm and to the whole partition or maybe to the whole hard disk.
- 2) Loosing the laptop means loosing the hard disk. So I have to create backups. This is getting more and more difficult, because the volume of the information is constantly increasing and the old fashioned backup strategies are getting difficult or at least annoying.

Possible solutions for the 2 problems:

Encryption of a hard disk content:

At the moment I use encfs, which allows me to encrypt a directory on my hard disk. encfs is very easy to use, it is a user space filesystem (FUSE) and using it is just a matter of mounting an encrypted folder into an empty folder, where the information gets accessible in unencrypted form. The mount command is password protected. encfs is very interesting for encrypting folders on external storages like USB sticks. The ease of use of encfs is a plus but also a minus, because it goes with the sake of high security. The single files and directories in the encfs encrypted directory get encrypted with a 128bit key, which would be fairly secure. But the problem is, that the key itself gets stored in the directory, too. The key gets encrypted, but for the sake of ease of use, the key is just encrypted with a simple password. This means a cracker has just to crack the keys password, and he will gain access to the key and through it to the whole encrypted data.

One increase of security I would like to check out, is to store the file, containing the key on an external storage, let's say on an USB key.

On the long run, I would like to encrypt the partitions on my laptop and probably I will encrypt the whole hard disk including the partition table itself. Such a system needs a possibility to store the decryption key externally. That could be a USB key, but preferable it is a device, which doesn't allow cloning of it's content. One possibility is a crypto card, like those I got from the FSFE when I have become a fellow of the FSFE.

Encrypting the hard disk will solve the problem of unwanted access to the data on it. But it will not

solve the problem of me, not being able to access the data on course of a system failure. And this risk is even much higher, because there are much more hardware failures than stolen systems.

Backup of a laptops content:

I want to minimize the risk of loosing data on course of a system failure or of the device gotten me stolen, by creating periodic backups. It's not easy to find the right tool, easy to use, efficient and secure to allow me to do periodic backups. Here some considerations:

The right storage for backups

One problem I was facing trying to find a good solution for creating the backup of the data on my laptop was the constantly increasing size of data I wanted to be backed up. My laptop has a DVD writer, but there is no DVD big enough to store a full backup and I am not willing to play disc jokey every time I create a backup. Also I don't trust those silver discs much more than my had disk. So I wouldn't like to end up with an unusable backup, just because the quality of the DVD discs wasn't high enough. So I looked for some kind of client server backup system. I tried different projects and at the end I discovered a very simple, easy to use tool. It's called duplicity.

Ease of use and support for various back ends

What I like of duplicity is it's ease of use. Making a backup with duplicity is like calling the cp command to make a copy of a folder. Very nice is the support for various back ends, like the local file system, ftp and ssh-scp. That means that the user can create a backup to a locally mounted file system, let's say an external hard disk, or to a remote server where he has an ftp or ssh account.

Securing the backup volumes

Duplicity also solves the problem of securing the backup volumes from evil access. That's a big problem with other backup solutions. Recurring questions of who creates backups are: Where can we lock away the backup volumes, what if someone cracks the backup server? duplicity solves this problem, just encrypting the backups using GPG (GNU Privacy Guard). This allows you to store the backup where ever you want, because the volumes are encrypted and the access to them will be useless without the pass phrase you gave while creating the backup. If you use the default symmetric encryption a pass phrase is enough to gain access and if you use the more secure asymmetric encryption to encrypt the volumes also your private key will be needed. This means that you don't have the need to lock away your backups any more and it enables you for example to use the free space on a gmail account to store the backup volumes. If you have a gmail account and if your backup doesn't exceed the 2.X GB of space available on a gmail-account, you could use gmailfs, which mounts your gmail account into your local file system and store the backup volumes there. Google will not be able to index them, because only you can decrypt the volumes with your pass phrase or better your private key. But be aware that google will have the right to remove all data on their server and therefore it's probably not the most safe place to put your backups.

Reducing bandwidth load and backup time:

Duplicity uses the rsync algorithm to make incremental backups. That means, that only the first time you make a backup of your files, they will be completely transferred to the backup volume, the consequent times only the changes you made on the local file system will be transferred to the backup volume. This reduces the volume of transferred data and enables duplicity to register even the changes made historically on your file system.