Partition–Rescue–mini–HOWTO

Table of Contents

Partition Rescue mini HOWTO....1
Jean−Daniel Dodin 1
1.What's in....1
2.What to do right now ?....1
3.Legal stuff....1
4.What do I must know right now ?....1
5.Disks....1
6.Partitions....1
7.Why is there a problem ?....1
8.Solving the problem....1
9.Authors....2
10.Most recent version....2
1.What's in....2
2.What to do right now ?....2
3.Legal stuff....3
4.What do I must know right now ?....3
5.Disks....3
6.Partitions....4
7.Why is there a problem ?....5
8.Solving the problem....5
8.1 The simpler case....5
8.2 A not so simple case....6
By hand....6
gpart 6
8.3 The rich man case....8
9.Authors....8
10.Most recent version....8

Partition Rescue mini HOWTO

Jean–Daniel Dodin

Jan 2000

Whow ! my disk is empty ! My Linux is gone ! If you have or fear to have one day or an other such a problem, read this...

1.What's in

2.What to do right now ?

3.Legal stuff

4.What do I must know right now ?

5.Disks

6.Partitions

7.Why is there a problem ?

8.Solving the problem

- 8.1 The simpler case
- 8.2 A not so simple case
- 8.3 The rich man case

9.Authors

10.Most recent version

1.What's in

This mini−HOWTO addresses only the "lost partition table" problem. This can be when :

- you have no more access to your computer, with the "no operating system" message,
- you have installed a new system (ie windows) and you see no more Linux, and windows take up all the capacity of the disk.

Here you will learn that if you know the right thing and do it, Linux comes usually safe from such things. Windows can, but it's luckier.

We will first see what you can do *before* the problem to easy the future recovery and what you must do *after* to recover. There is little to do to prevent from erasing a disk, usually this is done by automatic windows or Linux−install ill behaved programs or users mistakes − nothing can be done to prevent this except care, but you are already careful, isn't it ?

It can also be done by the use of MSDOS/Windows fdisk. avoid it as most as you can, but you probably can't.

I have done this many times, on my computer and on others' guy computers and restored Linux most of the time and windows sometimes. I wish you luck !

2.What to do right now ?

If you read this by curiosity or just seeking information and you are on a running Linux system, do immediately the following :

- open a root terminal or xterm,
- key in "/sbin/fdisk −l" (that last character being l for Lima).

you will be gratified from a list of all current partitions on all disks presents on your computer.

- Write back this on paper (or do "/sbin/fdisk −l | lpr" to print it) and save it in a safe place for future use.

If you are not the system administrator, you should not be concerned by the problem and can stop reading this.

3.Legal stuff

This mini−HOWTO is Copyright (c) 2000 by Jean−Daniel Dodin. All rights reserved. You may do (almost) whatever you want with it. I don't care. Just be sure to keep my name intact. I just hope it becomes useful for any person in the Linux community.

I am not responsible of any damage on any computer as a result of anyone reading this HOWTO. If you do any damage, _it is YOUR fault, NOT MINE !_ Be careful when partitioning disks, and don't make any mistakes, because it can be fatal ! Backup all your important data and check that everything you do is correct ! What is described here worked on my computer, but it may or may not work on your computer. Although it should work for everyone, I can't guarantee anything. This is the last warning you get : _BACKUP IMPORTANT DATA !_ Or, to put it short: Use at your own risk !

4.What do I must know right now ?

You *must* know that in case of any major problem with your hard disk, you *must* stop using it at all in *write* mode, at least the time necessary to understand what happens. Information there is very volatile...

If ever, one morning, awaking, you computer say "can't load, no system installed", you *must not begin reinstalling all the stuff.*

If you have windows installed, I can't promise you can recover your data, but it's likely you will recover all your Linux stuff, provided it's not located too low in the disk structure. This is because some windows viruses erases the very first disk cylinder, whatever is on. However I didn't ever experiment such virus and can't say for sure. Try recovering anyway.

You must also know that I give you all this information only for this −information purpose. Neither I nor any other people but you can be held responsible for any problem your data can have using this info. There are too many different systems on the world for anybody being able to promise anything. I can only wish you luck and hope you, like me, will be happy recovering data.

5.Disks

A hard disk is made of sectors numbered from 0 to the max.

dmesg gives, for example :

hdb: ST34321A, 4103MB w/128kB Cache, CHS=523/255/63

CHS means Cylinders, Heads, Sectors.

523*255*63=8401995 sectors of 512 bytes, thus the 4103Mbytes. This is only a logical map, it's not necessary what is written on the disk cover (except for the total size).

The true size of the sectors is of no interest for us given we don't want to modify anything but restore a previous state. For us, the default size given by fdisk is alright.

The size seen by the system is directly dependent of the work of the BIOS (Basic Input/Output System – the PC's ROM). The mode of the hard disk indicated in the BIOS is essential. On a new disk, it's better to use BIOS automatic hard disk recognition and say "yes". Anyway, any modification at this level may destroy all the data of the disk, so don't play with this without essential reason.

This is probably what your disk use, so don't be afraid.

6.Partitions

Disk are now huge, 13 Gb are not rare, so that it's not really handy to have all this stuff packed in only one part. Only windows do so, and if you use Linux, may be it's because you are aware of how inefficient the other is.

So a hard disk is usually cut in some pieces called "partitions" (see the "partition mini HOWTO" for details, also read /usr/doc/package/util/README.fdisk)

Let's get a look at (part of) my own print of fdisk –l :

Disk /dev/hdb: 255 heads, 63 sectors, 523 cylinders Units = cylinders of 16065 * 512 bytes

Device Boot Start End Blocks Id System

/dev/hdb1 1 153 1228941 83 Linux

/dev/hdb2 154 166 104422+ 82 Linux swap

/dev/hdb3 * 167 291 1004062+ 83 Linux

/dev/hdb4 295 523 1839442+ 5 Extended

/dev/hdb5 295 422 1028128+ 83 Linux

/dev/hdb6 423 523 811251 6 FAT16

This is my second hard disk, tied to guesses and tries (the first is too simple to be interesting).

/dev/hdb is my second ide disk (slave on the primary interface),

/dev/hdb1 is the first primary partition, running from the first (1) block to the block 153.

There can be four of such primary partitions. If one wants more than 4, one of them must be an "extended" one (not necessarily the fourth) and all other partitions are named "logical" and are located *inside* the extended one. Notice that partition number 5 and partition number three have the same beginning. Number five is logical, number 4 extended. Logicals begin always at 5, even if there are not 4 primary ones.

7.Why is there a problem ?

Problem is all the installed Operating Systems must share the disks and as at start the bios only scan the first one, there must be a so called "partition table" in the very beginning of this disk. This partition table is located in the Master Boot Record (MBR) side by side with the boot loader.

Any misuse of the MBR by any of the OS's leads to problems. When trying to install any system, yes answer at a question like "automatic partitioning ?" is likely to give problems... This is specially true with windows, especially with custom windows installation made by special makes PC's (when no true "windows" cd is included). But it's also true with new "smart" (not so smart !) Linux installation programs given with most recent distributions.

8.Solving the problem

Please, beware ! following the explanations given here will lead you to turn back to a previous system, loosing all your recent installed one, if any ! You must choose...

8.1 The simpler case

All is simple if you have at hand :

- a disk (floppy or cd) able to start Linux by itself with fdisk available − most rescue disks of any distribution can do that,
- a paper with the fdisk −l content written down.

It's enough to

1. start Linux,
2. start fdisk /dev/hda (or whatever is the disk to rescue),
3. use fdisk to delete (d option) all the existing partitions on the damaged disk,
4. use fdisk to create all the primary (1−4) partition mentioned on the paper,
5. give them the appropriate tag (t option) : 82 is for Linux swap, 83 for Linux main (L gives you the list), 5 is extended and must be done before creating logical partitions.
6. create any logical partition.

fdisk is a small and very smart programs. There are many other makes of fdisk, but I always prefer the bare bone one (I speak of Linux ones, of course, not the others...).

Be aware that fdisk doesn't write anything to disk before you hit w and return. In case you fear a mistake, hit q (quit) or Ctrl C (\^{ }C) to quit safe.

When your new partition table is written, start your Linux. Chance is you can't do that as usual : lilo can have been damaged also and you will need a boot floppy or booting from a cd (choose the option "booting the installed partition").

If you use to boot with lilo, as soon as you are logged in as root, key in "lilo" and hit return to reinstall you favourite boot loader.

Your Linux should be all here, test it. Try also to start windows if applicable. If you can't, there is a (very little) chance you can read your data from Linux, may be with a raw sector by sector read. If you can identify the disk sectors you data is on, using dd can copy them on a file. This is wise for text only. This recovery is NOT in the scope of this mini−HOWTO.

8.2 A not so simple case

By hand

This is when the previous case can't be used, for lack of fdisk paper or if it won't run for use of an out of date one.

First, be aware that as soon as you don't write to the disk (except with fdisk), you can't erase your data, so that you can use a block by block try. That is you need to know the beginning of the partition to start it. If, say a 153 don't fit, try a 154, and so on.

This can be tiresome, but if you remember approximately the size of the Linux partition, there is a chance to win.

gpart

But there is a better way if you can still access the net or have "gpart" at hand.

"gpart − guess PC−type hard disk partitions" is the first line of the man page of gpart (man gpart).

"gpart tries to guess which partitions are on a hard disk. If the primary partition table has been lost, overwritten or destroyed the partitions still exist on the disk but the operating system cannot access them.". This is exactly what we need.

gpart is still beta, but is already a very good tool.

The problem is the following : the first block of any partition is marked. But it's never "unmarked" if not overwritten. So many "first partition block" are existing on an old disk and gpart tries to do it's best guessing what is the good one. In fact it's not too difficult to try, nothing is written on the disk by gpart.

Here is the result of gpart on the previously seen disk hdb :

root@charles:/home/jdd > gpart /dev/hdb

Begin scan...

Possible partition(Linux ext2), size(1200Mb), offset(0Mb)

Possible partition(Windows NTFS), size(1200Mb), offset(1200Mb)

Possible partition(Linux ext2), size(1004Mb), offset(2402Mb)

Possible partition(Windows NTFS), size(1600Mb), offset(4102Mb)

End scan.

Checking partitions...

* Warning: partition(OS/2 HPFS, NTFS, QNX or Advanced UNIX) ends beyond disk end .

Partition(Linux ext2 filesystem): primary

Partition(OS/2 HPFS, NTFS, QNX or Advanced UNIX): primary

Partition(Linux ext2 filesystem): primary

Partition(OS/2 HPFS, NTFS, QNX or Advanced UNIX): invalid primary

Ok.

Guessed primary partition table:

Primary partition(1)

type: 131(0x83)(Linux ext2 filesystem)

size: 1200mb #s(2457880) s(63−2457942)

chs: (0/1/1)−(152/254/61)d (0/1/1)−(152/254/61)r

Primary partition(2)

type: 007(0x07)(OS/2 HPFS, NTFS, QNX or Advanced UNIX)

size: 1200mb #s(2457880) s(2457944−4915823)

chs: (152/254/63)−(305/253/60)d (152/254/63)−(305/253/60)r

Primary partition(3)

type: 131(0x83)(Linux ext2 filesystem)

size: 1004mb #s(2056256) s(4919781−6976036)

chs: (306/61/49)−(434/60/47)d (306/61/49)−(434/60/47)r

Primary partition(4)

type: 000(0x00)(unused) size: 0mb #s(0) s(0−0) chs: (0/0/0)−(0/0/0)d (0/0/0)−(0/0/0)r

As you see, primary partition can be recovered, but for extende ones it's still to be done.

Dos partitions are labelled "windows NTFS" because they were created while trying to install Windows 2000 (a very awful experience !). The "invalid" one is, in fact the extended partition.

With this, one can use fdisk and try re−creating the partition table (remember, this is risk−free given the original one is already lost).

8.3 The rich man case

Partition Magic is a commercial product, not so cheap given the little use one can have (approx a hundred bucks in France) but with a very high reputation all around there. However I never use it and will not rate it. It's said to be able to do anything with partitions, including restoring them. It's a windows tools, however, and I don't know if there is a dos part. If one needs to install windows to use Partition magic, the interest is low.

Original Ralf partition−rescue mini HOWTO was essentially based around the use of Partition Magic, so I presume it's a very good solution if you have valuable data on your Linux partition and little Linux capability. However there are now very recent makes of Partition Magic and I think it's better for you to read the manual.

9.Authors

The author of this HOWTO is Jean−Daniel Dodin. I can be joined at jdanield@free.fr or at jdanield@linux−france.org or jdanield@club−internet.fr.

My web site is now (January 20, 2000) at http://perso.club−internet.fr/jdanield, but may be soon at jdanield.free.fr or any personal address given by the www.free.fr IAP. Linux pages, who are of most interest for you are found at www.linux−france.fr/article/jdanield.

I want to thank Rolf Klausen (E−mail: rolfk@romsdal.vgs.no http://www2.romsdal.vgs.no/~rolfk/) who write the previous partition−rescue mini HOWTO even if I rewrite it almost entirely, he had first the good idea.

Every other member of the Linux community and everybody who supports Linux and writes documentation and programs for Linux and all the authors of the LDP and virtually any person involved in anything which has to do with Linux. Particularly Linus B. Thorvalds − he is _The King_ !!!

I want also to thank Michail Brzitwa <mb@ichabod.han.de> for writing gpart !

10.Most recent version

The most recent version of this mini−HOWTO shall be found on my Linux web site, at www.linux−france/article/jdanield/