

Linux.com

Everything Linux and Open Source

How to recover lost files after you accidentally wipe your hard drive

August 28, 2006 (8:00:00 AM) - 1 year, 11 months ago

By: **Shawn Hermans**

Recently I wanted to make sure I had enough space to back up my home digital videos and pictures, so I purchased a new hard drive to add to my home Linux server. I moved all the files I wanted to save onto a single hard drive and repartitioned the old hard drive so I could upgrade to a newer version of Linux. After going through the process of reinstalling the operating system, I mounted the backup hard drive and discovered that it was empty. I had some how mixed up the hard drive I used to back up all the data with a hard drive that I wanted to wipe. Because I had done such a poor job of retaining backups on external media, I did not have any backups of my pictures and videos.

After the overwhelming feeling of dread passed, I started to look into file recovery options. I demoed a variety of commercial products to see if any of them could find my lost files or partitions. Nothing seemed to work. Finally, I discovered **TestDisk** and **PhotoRec**, and was able to use the latter to recover my lost files.

TestDisk can recover lost partitions of virtually any filesystem. PhotoRec can recover files of most types, including most picture and video formats. PhotoRec can be used on existing partitions, or can be used to recover files on deleted partitions without having to recover the underlying partitions. Both PhotoRec and TestDisk can be run on DOS, Windows (9x, NT, 2000, XP, 2003), Linux, FreeBSD, NetBSD, OpenBSD, Sun Solaris, and Mac OS X, and, their developers claim, can be compiled and run on most Unix systems.

The recovery

I began my attempt at recovery by using TestDisk run from a Knoppix CD. Unfortunately, I had already overwritten the partition table, and an exhaustive search of the hard drive for lost partitions yielded too many results. I decided to use PhotoRec instead to recover my lost files.

PhotoRec recovers files by finding deleted files and copying them to disk. This means that files should not be recovered to the same disk partition on which the deleted files reside (unless you're recovering from a disk image file), because that could lead to the deleted data being permanently overwritten.

Another important thing to remember is that PhotoRec will most likely recover a lot of files. This means that the partition on which the recovered files are to be stored should have at least as much free space as the size of the partition on which PhotoRec is searching for recovered files.

Possible setups for recovery include:

1. Recover the files to a separate hard drive.
2. Recover the files to a networked storage drive.
3. Recover the files to a separate partition on the same hard drive.
4. Image the hard drive using a tool like **ddrescue** and recover files using only one partition.

As I had completed erasing my partitions, I could not use the third option. The second option introduces problems associated with network speed and latency. The fourth option is worth considering in the case of an incident response where the image of the hard drive is used as evidence.

I chose the first option, and installed two hard drives in a single computer. I divided the hard drive used to recover files into two major partitions; the first partition held the operating system (CentOS 4), while the second partition was set up to hold the recovered files. Partitioning in this manner is an extra precaution to prevent PhotoRec from halting the system by writing more files than the storage space allows. Another option is to run the operating system off a live CD such as **Knoppix**, which contains the TestDisk and PhotoRec utilities.

You can **download** both PhotoRec and TestDisk in a single archive file. The files `photorec_static` and `testdisk_static` are the executable files, and can be executed from the command line.

Make sure that the recovery partition is mounted (I mounted it at `/var/recovery`). Don't mount the hard drive that contains the deleted files; if the partition remains unmounted, you can't overwrite the data it contains.

Recovery steps

PhotoRec recovers files to the directory from which it is run. Therefore, I changed into the `/var/recovery` directory and ran `photorec_static`. If the PhotoRec executable does not run with this command, make sure that you either copy the executable to the `/usr/bin` directory or type in the full path where the program resides.

The PhotoRec interface is easy to understand. At the initial screen, you select the hard drive you wish to recover. In my case, it was `/dev/hdb`.

```
Select a media (use Arrow keys, then press Enter):  
Disk /dev/hda - 200 GB / 186 GiB (RO)  
Disk /dev/hdb - 160 GB / 149 GiB (RO)  
Disk /dev/hdc - 120 GB / 111 GiB (RO)  
Disk /dev/hdd - 296 MB / 282 MiB (RO)
```

```
[Proceed ] [ Quit ]
```

Next, you select the partition type. In my case, I selected an Intel/PC partition.

```
Disk /dev/hdb - 160 GB / 149 GiB (RO)
```

Please select the partition table type, press Enter when done.

```
[Intel ] Intel/PC partition
[Mac   ] Apple partition map
[None  ] Non partitioned media
[Sun   ] Sun Solaris partition
[XBox  ] Xbox partition
[Return] Return to disk selection
```

Note: Do NOT select 'None' for media with only a single partition. It's very rare for a drive to be 'Non-partitioned'.

The next screen listed the partitions on the hard drive. I wanted to recover partitions on the whole hard drive, so I selected the first option. However, before selecting this option, I needed to go to the [File Opt] menu to select which type of files I wanted to recover.

```
Disk /dev/hdb - 160 GB / 149 GiB (RO)
```

Partition	Start	End	Size in sectors
D empty	0 0 1 19456 254 63	312576705	[Whole disk]
1 * Linux LVM	0 0 2 19457 80 63	312581807	

```
[ Search ] [Options] [File Opt] [ Quit ]
                Start file recovery
```

PhotoRec can recover a variety of files, but I only wanted to recovery Word documents, AVI video files, JPG picture files, and MPEG video files. I selected the appropriate boxes.

```
PhotoRec will try to locate the following files
```

```
[ ] dbf  DBase 3, prone to false positive
[X] FAT  FAT subdirectory
[X] doc  Microsoft Office Document (doc/xls/ppt/vis/...)
[X] dsc  Nikon dsc
[X] eps  Encapsulated PostScript
[ ] exe  MS executable
[X] EXT2/EXT3 Superblock
[X] gif  Graphic Interchange Format
[X] gz   gzip compressed data
[X] jpg  JPG picture
[X] mdb  Access Data Base
[X] mov  MOV video
[X] mp3  MP3 audio (MPEG ADTS, layer III, v1)
[X] mpg  Moving Picture Experts Group video
[X] mrw  Minolta Raw picture
```

```
[ Quit ]
                Return to main menu
```

After you select the file types, go back to the previous screen and begin the scan of the hard drive.

The scanning process is automated; on my machine it took a few hours to complete. Once PhotoRec is finished, the recovered files will be in multiple directories of the form `recup_dir.x` where `x` is the number of the directory. The files within these directories will not contain the names of the original files; instead, they are numbered to indicate the order in which the file was recovered, and an extension that indicates the file type. For example, `f89.avi` is the 89th file recovered and is an AVI file.

Post-recovery cleanup

While all of my files were recovered, I had many files on my hard drive. Manually examining each file would be time-consuming and tiresome. I created three folders within the `/var/recovery` directory named `VID/`, `DOC/`, and `JPG/`, into which I sorted the files using the commands:

```
find /var/recovery/ -name "*.avi" | xargs -i mv {} /var/recovery/VID/
find /var/recovery/ -name "*.mpg" | xargs -i mv {} /var/recovery/VID/
find /var/recovery/ -name "*.jpg" | xargs -i mv {} /var/recovery/JPG/
```

Although all the files are sorted into folders of the same type, the sorting was far from over. Before my accident, my hard drive contained more than 10,000 pictures, each around 2MB in size. During the recovery process, PhotoRec recovered all the pictures it could find -- including picture files from the Web browser cache. This meant it brought back a lot of unwanted files. To eliminate most of the picture files from miscellaneous sources, I moved files smaller than 1MB to a folder called `SMALL`, which I kept until I was satisfied that none were of interest. I moved the files to the folder using the command:

```
find /var/recovery/JPG/ -name "*.jpg" -size -1024k | xargs -i mv {} /var/recovery/SMALL/
```

PhotoRec does not recover the file names of recovered files, but luckily my recovered picture files contained EXIF metadata such as the time and date the picture was taken and the camera make and model. I used the **Jhead** command-line utility to extract this metadata. In the `JPG` folder I ran the command:

```
jhead -n%Y%m%d-%H%M%S *.jpg
```

This command renames all files with the `jpg` extension with its time/date stamp in the format `YYYYMMDD-HHMMSS.jpg`. Any files with the same time and date stamp are named in the format of `YYYYMMDD-HHMMSSx.jpg`, where `x` is a lower-case letter that increments for each duplicate time/date stamp found. Given that these pictures were all taken on the same digital camera, any pictures with the same time/date stamp should be the same picture. I moved duplicates to a folder called `DUPS` using the command:

```
find /var/recovery/JPG/ -name "*.a.jpg" | xargs -i mv {} /var/recovery/JPG/DUPS/
```

Once I had the files labeled with the time/date stamp, I could sort them into folders according to their year and month.

If I had included keywords or comments in the picture files, I could have used **libextractor** to extract keywords from the JPEG files and sort the files into folders using those keywords. Alas, this was not the case, so I had to spend hours sorting the pictures manually into folders after the recovery. I did however use libextractor on my AVI files to determine information regarding the codec, frame-rate, and resolution of those videos.

How to prevent recovery

I was glad that I was able to recovery my files with such "ease," but I also realized how easy it would be for my sensitive data to be recovered if I ever got rid of an old computer or hard drive. Luckily, you can wipe data from a hard drive in such a way as to prevent files from ever being recovered.

Whitedust Security gives the following as options for secure data removal.

1. Writing over existing data with "junk" data.
2. Giving the hard drive an acid bath.
3. Degaussing the hard drive with a degausser.
4. Damaging the disk with fire.

Assuming you do not want to render the hard drive unusable, the best options are either writing over existing data or degaussing the hard drive. If you don't have access to degaussing equipment, use a program like **Wipe** that writes over data with patterns known to cause data to become unrecoverable. Based upon your paranoia level, you can wipe your data with as many passes as you see fit. Some recommend 22 passes, while others say 99 is needed for absolute security. If you are only worried about the casual snooper who only has access to tools like PhotoRec, then three or four passes should suffice.

Read in the original layout at: <http://www.linux.com/articles/56588>