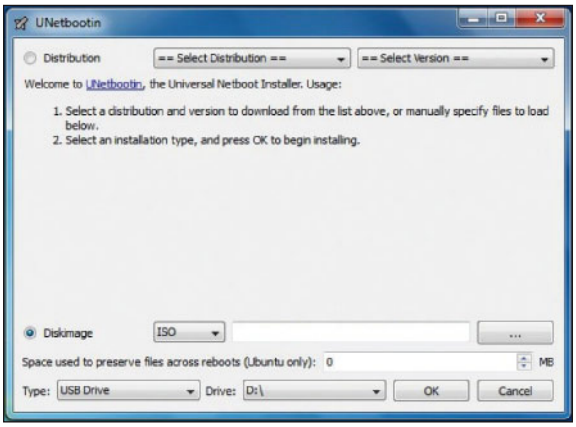
Recovery: Save

Think your precious files are lost forever? Think again. **Jonathan Roberts** explains how to recover missing data using two indispensable applications.



› ***Unetbootin*** **lets you make bootable live CDs in Windows, so even if your machine's broken you can use a friend's.**

Our expert

Super-secretive **Jonathan Roberts** has been tinkering with Linux since he was a teenager. Which actually wasn't that long ago...

In **LXF149**, Bob Moss walked us through some of the best backup applications for Linux, demonstrating how to keep your data safe in case of an accident. But even after reading Bob's words of wisdom, few of us back up as often as we should, and this almost always leads to the stomach-churning moment when you realise a crucial folder is gone forever.

Fortunately, forever doesn't always have to mean forever, and in this article we're going to show you what to do when disaster strikes. We'll be making use of the excellent *TestDisk* and *PhotoRec* applications, which between them can recover almost any file.

Before diving in, be sure to read the Where'd It Go box. It explains what to do as soon as you know data loss has occurred since, no matter how good *TestDisk* and *PhotoRec* are, if you don't respond quickly and sensibly, there'll be little they can do. With a bit of knowledge and a bit of luck, however, your next data loss shouldn't be such a terrible event.

The tools for the job

As with most Linux tasks, the first hurdle you must overcome is installing the tools you need to do the job. With *TestDisc* and *PhotoRec*, you'll want to give a little extra thought to what format you grab them in, since it will depend on where your lost files are located. If your lost files were on an external drive, then this won't pose much of a problem: you can just grab the **testdisk** package, which is found in most distribution's repositories. So, if you use Ubuntu, **sudo apt-get install testdisk** should do the trick, and if you use Fedora **su -c "yum install testdisk"** will get what you need.

If your lost files were stored on your own computer, the best thing to do is borrow a friend's system and download a live CD with **testdisk** on. There's lots listed on the *TestDisk* website, **http://www.cgsecurity.org/wiki/TestDisk_Livecd**, but we'd recommend the *GParted* live CD because it's part of a reliable project.

This is ideal because your friend doesn't need to have a Linux system, only a CD writer drive or a spare USB port. You can then use something like *ISO Recorder* or *Unetbootin* to turn the downloaded ISO file into a bootable system. It also means that you won't have to mount the hard disk to run the tools, decreasing the risk of overwriting lost data.

TestDisk to the rescue

OK, now we're ready to dive in and start rescuing some files. We'll begin by looking at how to use *TestDisk* to recover some files from a FAT32-formatted USB stick. Not only is this a common situation to find yourself in, learning this set of skills will be of most use to your friends and family as well, so it's a great place to start.

As we proceed, we'll be keeping a close eye on how to operate *TestDisk*'s interface, since it's far from consistent!

With the USB stick inserted into the computer, open a terminal window and run the *TestDisk* command. Running it as a normal user will give you access to external disks and non-system partitions, but if, later on, you want to use it on system partitions, you'll have to start it as root or with sudo.

» **Last month** We learned how to secure data using the power of encryption.

your lost files

Which tool to use?

Once you've got the tools installed, you're probably itching to get started and see how much of your data you can get back.

First, though, you need to know which tool to start with – *TestDisk* or *PhotoRec*.

Each is best suited to different scenarios, so here we'll have a quick look at their strengths and weaknesses, so you know which to use when.

TESTDISK

TestDisk was originally designed as a partition recovery tool, which means that it's best to use if your problems arose after:

» Accidentally deleting a partition, maybe while installing a new distro.

» Your partition was corrupted by a virus or faulty software and made unreadable.

In these situations, *TestDisk* will be able to recognise the lost partitions, copy the data off them, and sometimes even restore the partition table to the way it was before.

It's also very good at recovering individual deleted files from FAT, NTFS and ext2 formatted drives.

This is really useful since most USB sticks will be formatted in either FAT or NTFS for compatibility with Windows.

Since this is the format used by Windows, it also gives you a chance to save the day for friends and family and show off just how great Linux is.

Unfortunately, there are few Linux distributions that use the old, but reliable, ext2 format for large data drives. Instead, they mostly use ext3 or ext4, which aren't supported by *TestDisk*.

PHOTOREC

This is where *PhotoRec* steps in. It was originally designed to recover lost photos, hence it's name, but it's since expanded to include almost any file format that's in common use, including:

» archives: ZIP, TAR, TAR.GZ

» media: FLV, MP3, MP4 and Ogg

» office documents: ODT, DOC and DOCX

For a full list, see **http://www.cgsecurity.org/wiki/File_Formats_Recovered_By_PhotoRec**.

The best thing about it is that it works by ignoring the filesystem. Instead it looks for unique signatures left by certain file types. Its website says that it definitely works with FAT/ext2/ext3, but in reality it should work with almost any block-based filesystem.

It's worth noting that *PhotoRec*, unlike *TestDisk*, is unable to recognise filenames, so if *TestDisk* will work, that should be your first port of call.

TestDisk operates very much like a wizard, walking you through a series of steps to get your files back.

After launching the application, your terminal will be taken over by *TestDisk*'s first screen, the log file screen. Here, you're given the option of asking *TestDisk* to create a log file of all its activity, which can be useful if you need to get further support from a forum or IRC. For our purposes, though, we're just going to select No Log.

> **"TestDisk operates like a wizard, walking you through a series of steps."**

You navigate your way around this screen, as with all of *TestDisk*'s screens, by using the cursor keys. Pressing Up or Down highlights a different option and pressing Return selects the item that you've currently got highlighted.

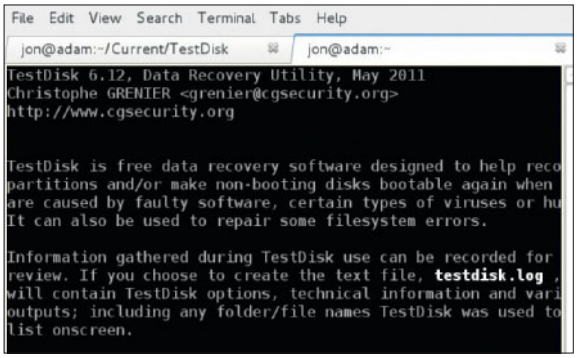
The next screen will ask you to select which device you want to work with. At the end of each device entry, you will see the device label, the name that's displayed when you open it in a file browser, which should make it easy for you to identify the correct one.

Navigation on this screen works slightly differently to before. You still use the cursor keys, and by default the Return key selects the currently highlighted device, but there's also an array of further options displayed at the bottom of the screen.

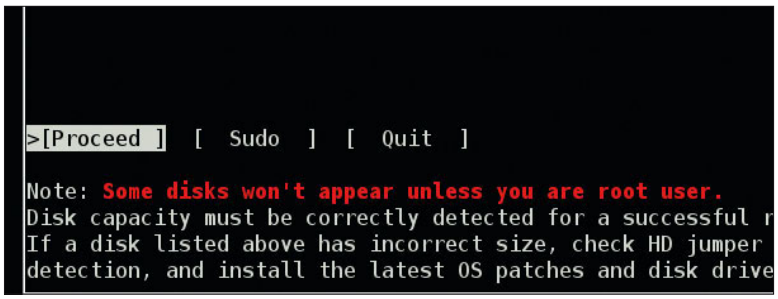
As you might imagine, you can navigate these using the Left and Right arrow keys. Bear in mind that the Return key actually operates on these options, not the currently highlighted device, so if you move away from Proceed, make »

Quick tip

In *TestDisk*, use a colon to add a deleted file to your recovery selection. The colour won't change, but it will have been selected.



› **Even though *TestDisk* is text-based, its wizard-based interface makes using it a breeze.**



› ***TestDisk* makes use of a number of different interaction methods, so be sure to keep an eye on the bottom of the screen.**

» **If you missed last issue** Call 0844 848 2852 or +44 1604 251045

» sure you return to it before pressing Return. This interface pattern is repeated elsewhere in the program, so be sure to keep an eye on the bottom of the screen for hidden goodies.

Next you're asked to choose a partition type and about 90% of the time, you'll need to select [Intel] Intel/PC partition. This is the basis of all Windows compatible devices, so, if you're in doubt, it's probably the option you're looking for.

Rescue files

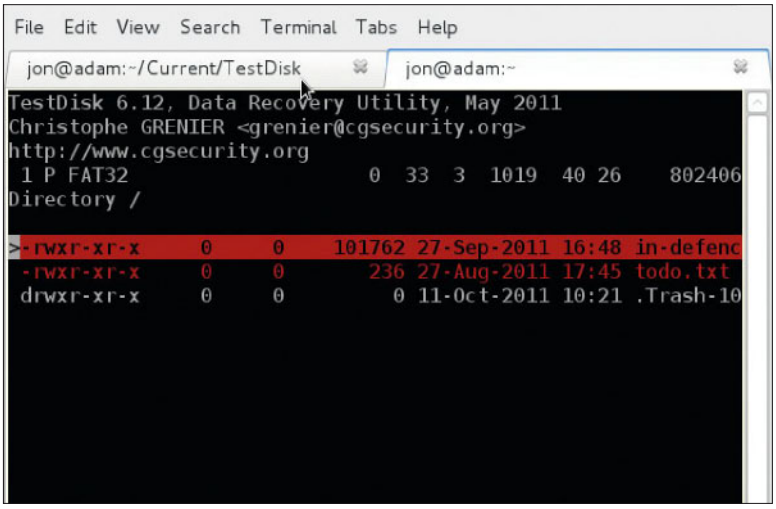
On the next screen, select the Advanced option (Return works fine here) and then use the Right arrow key to select Undelete. You'll then see a list of all the files on the device. Deleted files will be displayed in red, and will display information such as last modified time and date, and the filename at the very end of the line. If you hover over a directory, the Right arrow key will allow you to enter it and have a look what's inside, and then the Left arrow key will bring you back again.

As you can see by looking at the bottom of the screen, there are various options you can perform on the files and folders, and yet another new interface paradigm is introduced. No longer just restricted to arrow keys, there's all kinds of hotkeys listed here: typing a colon will select a file or folder, C will copy all selected files while c will copy the current file.

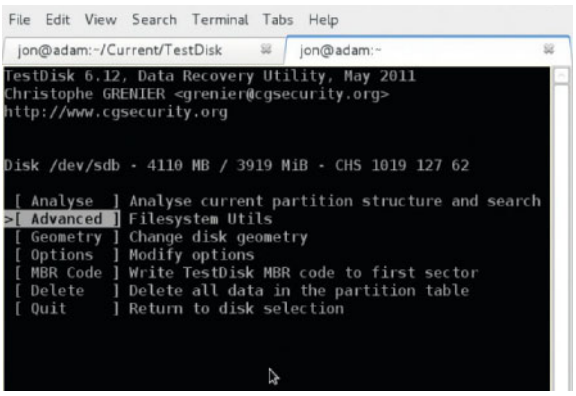
You might be surprised to only see copy options here, rather than anything to actually 'undelete' the file, but this is entirely intentional on the part of the *TestDisk* developers.

Quick tip

In both applications, it's good to check you've got the right disk by looking at its size. But note that sizes are in MB, so divide by 1,024 to get GB.



› ***TestDisk*** **highlights deleted files that it's found in red.**



› **When recovering partitions instead of files, you'll need to select the Analyse rather than the Advanced option.**

Rather than risk overwriting other deleted files, they only ever copy deleted files off the disk and on to another.

When you press one of the copy keys, you'll need to select a destination for the files to be copied to. By default, *TestDisk* will put you in your home directory, from where you can navigate with the arrow keys as before. If you want to put the files in the home directory, or whichever directory is currently displayed, highlight the entry whose name is just a single dot '.' and press C. And that's all there is to it. Your files will now be safe in whichever destination folder you selected, ready for you to back up or do what you want with.

Rescue a partition

Excellent, but what happens if you managed to delete an entire partition? Well, *TestDisk* can help with that, too.

As before, we'll demonstrate using a FAT-formatted USB stick, but the principles will be the same for any other disk. Of course, if the disk happens to be on your main system and is no longer bootable, you'll have to get hold of a live CD from which to run *TestDisk*.

The first thing to do is insert the USB stick and launch a new terminal window to run *TestDisk* in. Then you can proceed as above, making your way through the menus until you reach the list that includes Analyse, Advanced etc. Here, instead of selecting Advanced, choose Analyse and then Quick Search. *TestDisk* will then examine your disk, looking for evidence of old partitions that have since disappeared. Very quickly it'll respond and show a list of partitions that it managed to discover.

If you're unsure whether it's found the correct partition or not, there's a few ways you can check. First, at the end of

Where'd it go?

When a file gets deleted, even if you remove it from the trash, it's not really gone. What actually happens is that the operating system marks it as deleted, and informs other applications that the space taken up by this file is now available for other uses. Until another application comes along and uses that space, however, the file and its contents are still there.

This means that if you ever delete a file by accident, the first thing you must do is stop using the disk that it was stored on. If it was on a USB stick or an SD memory card, for instance, remove it immediately; if it was on your hard drive, then you'll want to shut down the computer and not touch it again until you're ready to do the file recovery.

This will minimise the chances that another application will write over the old data, and increase your chance of being able to recover the data.

What happens if you didn't just delete a single file, but accidentally corrupted an entire partition? Well, the steps are the same, really: as soon as you realise what's happened, stop using the device. In this situation, however, you may also find that since computers won't be able to recognise the contents of the device at all, they may prompt you to reformat it. Make sure you don't agree to this, as it's only going to complicate matters.

» **Never miss another issue** Subscribe to the #1 source for Linux on page 66.

each partition's entry in the list, you should see a label. If you recognise this, either as something you explicitly set or as something that appears when you plug in your USB stick, then it's probably the right partition.

The other thing you can do is press P to see a list of files that can be found on that partition. If you do this and you see your files, with appropriate time stamps and names, then you've definitely found the right partition.

> **"All that's left to do is breathe a sigh of relief... and back up your data!"**

Make a copy

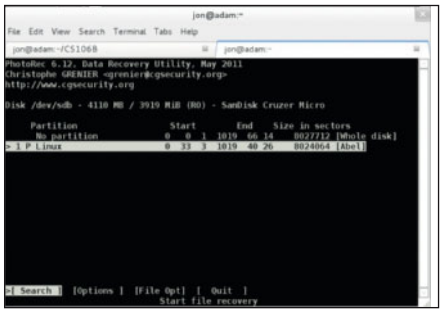
Once you're sure, we'd recommend pressing P to list the files anyway, and then copying them all off the device. This way, if anything else goes wrong, such as *TestDisk* failing to recover the partition, at least your files will be safe (assuming *TestDisk* supports this kind of operation for the disk partition format).

With your data a little safer, let's return your disk to the way it was before the accident, saving lots of unnecessary copying. From the same screen at which you pressed P to list the files, simply press Return to proceed further, and then select Write to restore the partition to the way it was. As soon as you've done that, you should be able to reboot and see your disk appearing exactly as it did before. All that's left to do is breathe a sigh of relief... and back up your data!

There's one other option that we should bring to your attention here, and that's to perform a Deep Search. If the Quick Search failed, this option can sometimes find older or more badly damaged partitions. You may not have as much luck recovering files from disks that are this damaged, but it's well worth a try. LXF

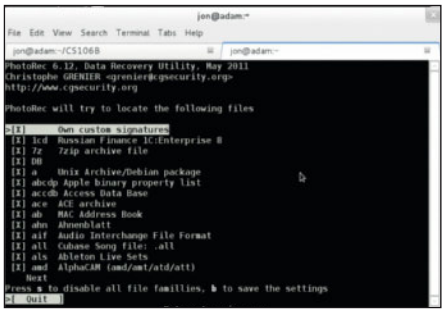
Step-by-step: Recovering files with PhotoRec

OK, so we've showed you how to rescue files from Windows-compatible partitions, and how to rescue entire partitions, but you're a Linux user: what if you want to recover files from an ext3 or ext4 partition, or maybe even ReiserFS or Btrfs? Here, we're going to walk you through *PhotoRec*, the tool that's designed specifically for this situation. It works in a similar way to *TestDisk*, so it should be familiar.



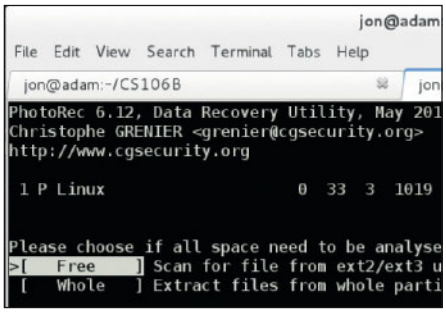
1 Launch PhotoRec

Launch the **photorec** command from a terminal, and then proceed as before until you reach the partition selection screen.



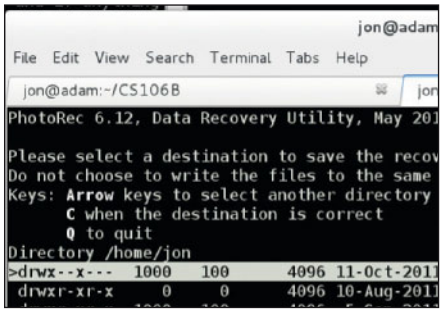
2 Specify filetypes

On the partition selection screen, first select File Opt to specify which filetypes you're searching for – you don't want to end up with 20,000 files to hunt through.



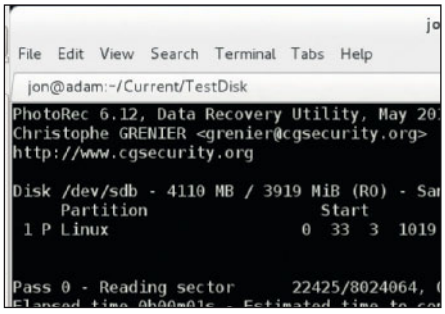
3 Search the drive

Return to the partition selection screen, and then select Search. On the screen that follows, be extra thorough by opting to scan the entire drive.



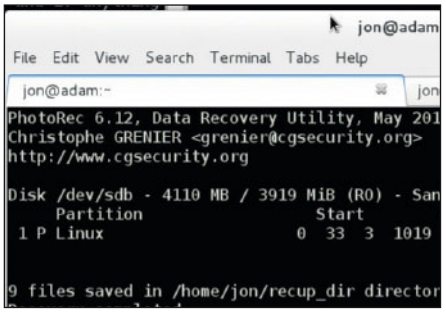
4 Begin recovery

Choose a destination for the recovered files, and press C to begin recovery. Be warned, it can find a lot of files, so it's best to select a sub-directory you don't mind getting messy.



5 Twiddle your thumbs...

Go off and make a cup of tea or something...



6 Rename files

Success! Now, check your file manager and see if it's found the correct files. Make sure you rename them, however, as *PhotoRec* doesn't recover filenames.

» **Next month** Discover how to manage your music collection with Banshee.