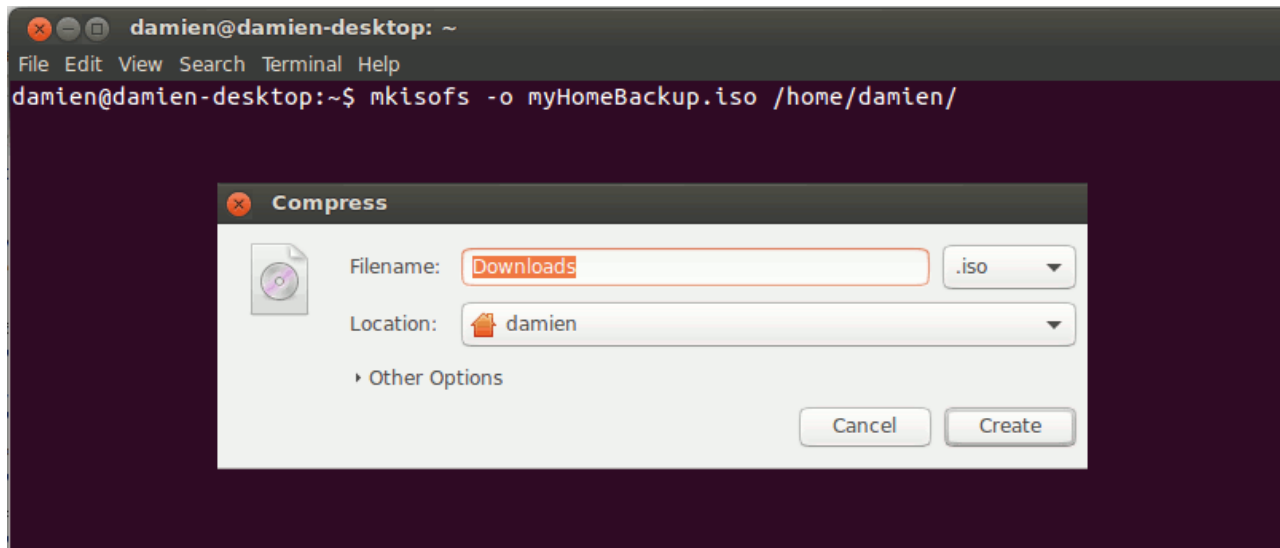


# How To Create ISO Files In Linux

MTE [maketecheasier.com/how-to-create-iso-files-in-linux](https://maketecheasier.com/how-to-create-iso-files-in-linux)



You probably know that an ISO file can be burned onto a CD/DVD or USB drive, but did you know that you can easily back up or archive your files and folders into an ISO file as well? With this ISO file you can then burn it onto a CD/DVD as a backup or simply mount it like an external drive and access the files from within.

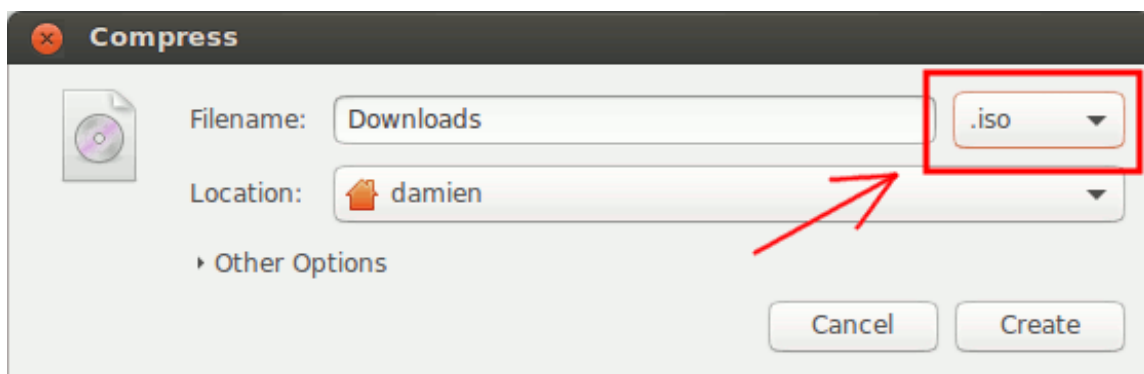
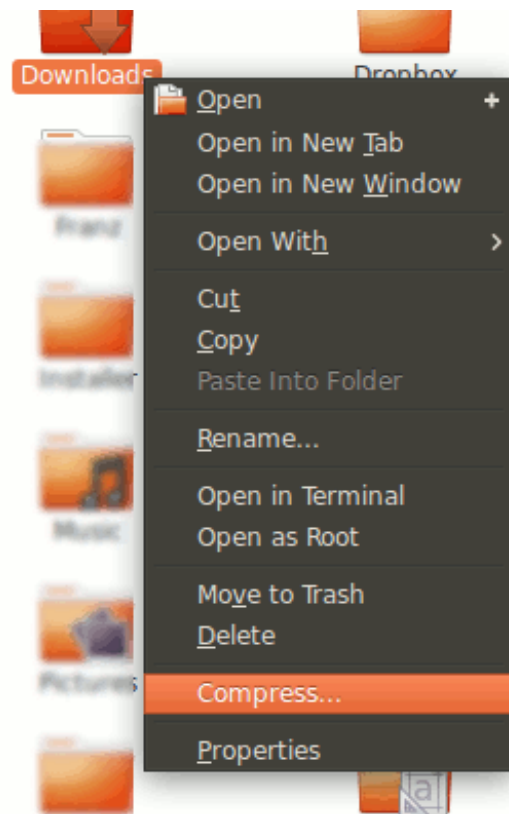
If you want to back up the content of a disc or if you have a bunch of files and folders that you want to back up and archive, here is how you can create ISO files in Linux.

## Via Archive Manager

If you are using Ubuntu (or Gnome desktop), the Archive Manager application (file-roller) allows you to easily create an ISO file.

1. Open the file manager. Select the files and folders that you want to back up. Right-click on them and select "Compress."

2. Select “.iso” option, and click “Create.” This will compress all the selected files and folders into an ISO file.



## Via Command Line

`dd` is a useful command that you can use to create ISO file. All you need to do is specify the source and destination, and it will do the necessary work of creating an ISO file.

The basic usage is as follow:

```
dd if=source of=destination
```

For example, if your CD-ROM drive is mounted at “/dev/hdc,” and you want to back up the content of your disc to a “my-cd-backup.iso” file, you can use the following command:

```
dd if=/dev/hdc of=/home/username/my-cd-backup.iso
```

The “source” doesn’t have to be a CD-ROM drive. It can be a hard drive partition, an external drive or a file path, though it won’t work on a folder.

**Note:** *there is much more to `dd` than described above, but we won’t go into the details here.*

Alternatively, you can also use the `mkisofs` command to create an ISO file. The advantage of `mkisofs` is that it gives you more options to customize how you want the ISO file to be created.

The basic usage is as follow:

```
mkisofs -o destination-filename source
```

For example, use the following command to backup your Home folder:

```
mkisofs -o myHomeBackup.iso /home/username
```

You can tell `mkisofs` to enable Rockridge extension by setting the `-R` option.

```
mkisofs -R -o myHomeBackup.iso /home/username
```

The Joliet extension is enabled by the `-J` flag.

```
mkisofs -J -o myHomeBackup.iso /home/username
```

You can also set a volume name (`-V` option) for the ISO file. (If you burn your ISO file to a CD, the volume name will be used as the CD’s name.)

```
mkisofs -V "Home Folder Backup" -o myHomeBackup.iso /home/username
```

You can also exclude certain files to add to the ISO file with the “-m” option. It supports wildcard (\*), so you can use it for something like the following

```
mkisofs -m ".*" -o destination source
```

to exclude all hidden files (filename with a “.” in front) from adding to the ISO file.

Check out all the options of `mkisofs` with the following command:

```
mkisofs --help
```

## Conclusion

---

While it may not be the best backup option, an ISO file can be rather useful in some instances. With the above instructions, you can now create ISO files with ease.

