

## How to use Rsync to exclude Files and Directories in Data Transfer

### Introduction

Rsync is a useful Linux command-line tool that syncs and copies files and directories. You can use the tool to synchronize data locally between directories and drives, or between two remote machines.

The basic [rsync commands in Linux](#) sync everything from a location you specify. In many backup scenarios, you may want to exclude specific files, directories, or file types.

**Follow the examples in this guide to learn how to exclude files or directories with rsync.** We will list the most common exclude use-cases to cover your day-to-day rsync usage.

### Prerequisites

- User with **sudo** or **root** privileges
- Access to a command line/terminal window
- Rsync installed on your system

## How rsync Exclude Option Works

The **--exclude** option with the **rsync** command uses relative paths to the source directory. Append the **--exclude** option to the **rsync** command, followed by the relative path to a directory or file(s).

**The basic syntax** for the rsync exclude option looks like this:

```
rsync [OPTIONS] --exclude 'file_or_directory' source/ destination/
```

Replace *source/* with the directory name you want to use as a source for data transfer to another location.

Replace *destination/* with the directory name rsync will use as the target location for your data. If the directory does not exist, rsync creates one for you and transfers the files to that directory.

To back up files to a remote location, follow our other guide to learn [how to copy or transfer files with rsync over SSH](#).

**Note: The trailing slash (/) on the source directory** modifies the behavior of the **rsync** command.

- If you do not use a trailing slash, the source directory is copied to the destination directory, and then the contents of the directory.
- When you do use the trailing slash, rsync only copies the content of the source **without** creating an additional directory level.

## Exclude a Specific File

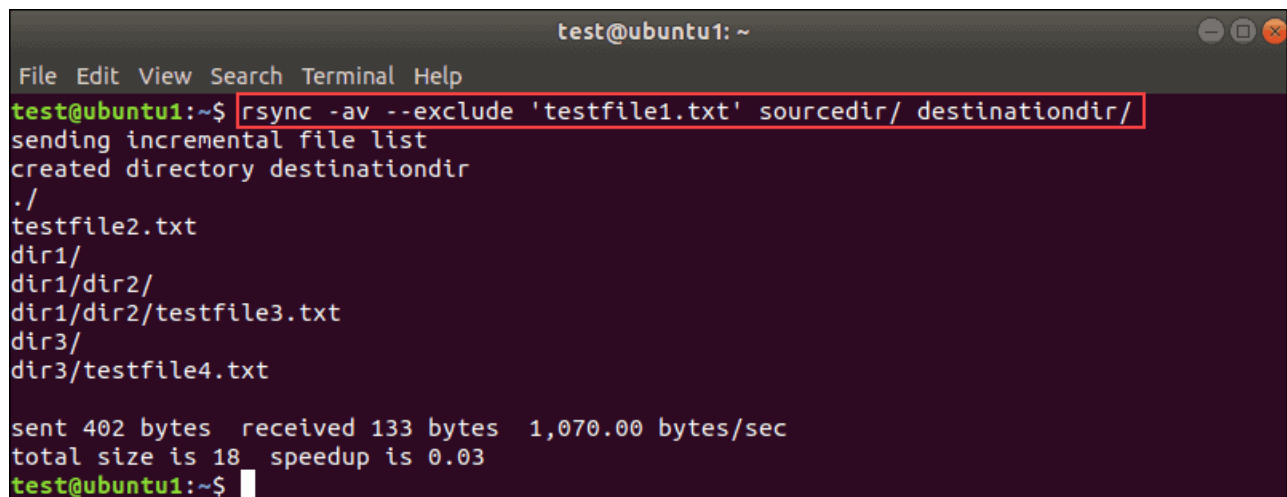
In the rsync exclude examples below, we will use the **-a** (**archive**) and **-v** (**verbose**) options (**-av**). The **-a** option syncs directories recursively while keeping permissions, symbolic links, ownership, and group settings. The **-v** flag is optional and prints the progress and status of the rsync command.

To exclude a file while transferring the contents of a folder with rsync, specify the file and the relative path.

For example:

```
rsync -av --exclude 'testfile1.txt' sourcedir/ destinationdir/
```

This command allows you to copy the files to *destinationdir* from *sourcedir* but exclude *testfile1.txt*.

A terminal window titled 'test@ubuntu1: ~' with a menu bar (File, Edit, View, Search, Terminal, Help). The command 'rsync -av --exclude 'testfile1.txt' sourcedir/ destinationdir/' is entered and highlighted with a red box. The output shows 'sending incremental file list', 'created directory destinationdir', and a list of files: './', 'testfile2.txt', 'dir1/', 'dir1/dir2/', 'dir1/dir2/testfile3.txt', 'dir3/', and 'dir3/testfile4.txt'. At the bottom, it shows 'sent 402 bytes received 133 bytes 1,070.00 bytes/sec', 'total size is 18 speedup is 0.03', and the prompt 'test@ubuntu1:~\$' with a cursor.

```
test@ubuntu1:~$ rsync -av --exclude 'testfile1.txt' sourcedir/ destinationdir/
sending incremental file list
created directory destinationdir
./
testfile2.txt
dir1/
dir1/dir2/
dir1/dir2/testfile3.txt
dir3/
dir3/testfile4.txt

sent 402 bytes received 133 bytes 1,070.00 bytes/sec
total size is 18 speedup is 0.03
test@ubuntu1:~$
```

As you can see in the output, *testfile1.txt* is not on the list.

To do so, run this command:

```
sudo rm -rf destinationdir
```

## Exclude a Specific Directory

Now that you know how to use the **rsync --exclude** file command, use the tool in the same fashion to exclude a directory.

Specify a directory you want to exclude (instead of a file name):

```
rsync -av --exclude 'dir1' sourcedir/ destinationdir/
```

```
test@ubuntu1: ~  
File Edit View Search Terminal Help  
test@ubuntu1:~$ rsync -av --exclude 'dir1' sourcedir/ destinationdir/  
sending incremental file list  
created directory destinationdir  
./  
testfile1.txt  
testfile2.txt  
dir3/  
dir3/testfile4.txt  
  
sent 335 bytes  received 121 bytes  912.00 bytes/sec  
total size is 18  speedup is 0.04  
test@ubuntu1:~$
```

This command copied the contents of *sourcedir* into *destinationdir* and excluded *dir1*, as seen in the output.

## Exclude Files or Directories Based on a Pattern

Use an asterisk \* (wildcard) when defining a file or directory name to exclude everything that matches the pattern.

To **exclude files that start with** *test*, run this command:

```
rsync -av --exclude 'test*' sourcedir/ destinationdir/
```

Every file and directory that matches this pattern will be excluded from the transfer.

You can also use the wildcard in a similar matter **to exclude all directories that end with a specific pattern**.

Run this command to exclude all directories that end with number 3:

```
rsync -av --exclude '*3' sourcedir/ destinationdir/
```

You can use an asterisk before and after a pattern to additionally refine the **--exclude** criterion.

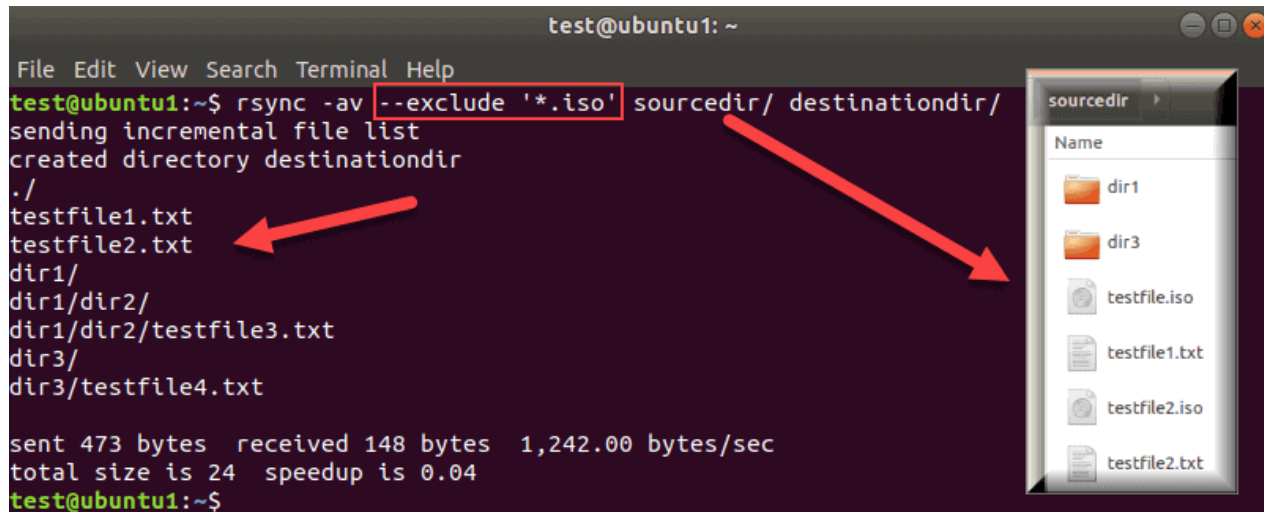
## Exclude a Specific File Type

The rsync tool allows you to **exclude certain file types** when synchronizing data. Use an asterisk \* followed by the extension of the file type you want to exclude.

For example, you may want to back up a directory that contains many *.iso* files that you do not need to back up.

To exclude a specific file type, in this case *.iso*, run this command:

```
rsync -av --exclude '*.iso' sourcedir/ destinationdir/
```



The screenshot shows a terminal window titled 'test@ubuntu1: ~' with a menu bar (File, Edit, View, Search, Terminal, Help). The terminal output shows the execution of the command `rsync -av --exclude '*.iso' sourcedir/ destinationdir/`. The output includes: 'sending incremental file list', 'created directory destinationdir', a list of files and directories being transferred (./, testfile1.txt, testfile2.txt, dir1/, dir1/dir2/, dir1/dir2/testfile3.txt, dir3/, dir3/testfile4.txt), and statistics: 'sent 473 bytes received 148 bytes 1,242.00 bytes/sec', 'total size is 24 speedup is 0.04'. A file explorer window titled 'sourcedir' is open on the right, showing a list of files and directories: dir1, dir3, testfile.iso, testfile1.txt, testfile2.iso, and testfile2.txt. Two red arrows point from the `--exclude '*.iso'` part of the command to the file explorer, highlighting that the .iso files (testfile.iso and testfile2.iso) are excluded from the transfer.

```
test@ubuntu1: ~
File Edit View Search Terminal Help
test@ubuntu1:~$ rsync -av --exclude '*.iso' sourcedir/ destinationdir/
sending incremental file list
created directory destinationdir
./
testfile1.txt
testfile2.txt
dir1/
dir1/dir2/
dir1/dir2/testfile3.txt
dir3/
dir3/testfile4.txt

sent 473 bytes received 148 bytes 1,242.00 bytes/sec
total size is 24 speedup is 0.04
test@ubuntu1:~$
```

The output shows that rsync did not transfer `.iso` files. We added the contents of the source directory in the image for comparison.

## Exclude Files by Size

During file transfers, you can specify the minimum or the maximum size of files you want to exclude. If your source directory contains many large files that you do not want to back up, use the **`--max-size=size_in_mb_or_gb`** option. Replace the `size_in_mb_or_gb` with the desired size.

For example, to exclude all files larger than 500MB, run this command:


```
rsync -av --max-size=500m sourcedir/ destinationdir/
```

On the other hand, to exclude files smaller than a specific size, use the **`--min-size=size_in_mb_or_gb`** option.

For example, you want to transfer a directory with images, but there are many thumbnail files. If all your images are larger than 1MB, then you can exclude all files smaller than that size.

To do so, run this command:

```
rsync -av --min-size=1m sourcedir/ destinationdir/
```

```
test@ubuntu1: ~  
File Edit View Search Terminal Help  
test@ubuntu1:~$ rsync -av --min-size=1m sourcedir/ destinationdir/  
sending incremental file list  
created directory destinationdir  
./  
dir1/  
dir1/dir2/   
dir3/  
  
sent 326 bytes  received 68 bytes  788.00 bytes/sec  
total size is 999,429  speedup is 2,536.62  
test@ubuntu1:~$
```

Do not use the **--exclude** rsync option with when defining the minimum or maximum file size.

## Exclude Multiple Files or Directories

Add multiple **--exclude** options to exclude multiple files or directories. You can combine any **rsync --exclude folder** and **rsync --exclude file(s)** commands to transfer only the data you need. Any of the commands we previously talked about can be used in one line.

The following command will exclude all files with the *.txt* extension, as well as the *dir3* and *dir4* directories:

```
rsync -av --exclude '*.txt' --exclude 'dir3' --exclude 'dir4' sourcedir/  
destinationdir/
```

You can add as many **--exclude** entries as you need. To keep the command tidy, you can specify files and directories in the **--exclude** option using curly brackets. Separate the patterns using a comma.

For example, we can shorten the above command in this fashion:

```
rsync -av --exclude={'*.txt','dir3','dir4'} sourcedir/ destinationdir/
```

```
test@ubuntu1: ~  
File Edit View Search Terminal Help  
test@ubuntu1:~$ rsync -av --exclude={'*.txt','dir3','dir4'} sourcedir/ destinationdir/  
sending incremental file list  
created directory destinationdir  
./  
testfile.iso  
testfile2.iso  
testfile3.iso  
dir1/  
dir1/dir2/  
  
sent 999,967 bytes  received 125 bytes  2,000,184.00 bytes/sec  
total size is 999,405  speedup is 1.00  
test@ubuntu1:~$
```

The output shows that the listed files and directories are excluded from the transfer.

## Exclude Files and Directories from a List

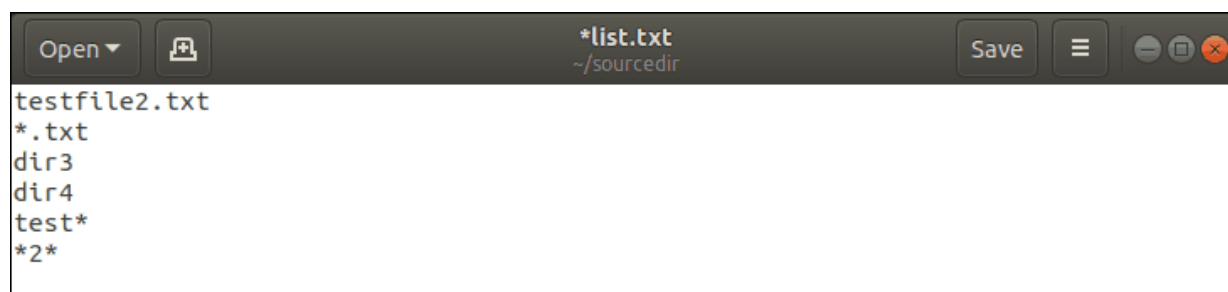
When you need to exclude a large number of different files and directories, you can use the `rsync --exclude-from` flag. To do so, create a text file with the name of the files and directories you want to exclude. Then, pass the name of the file to the `--exclude-from` option.

The command looks like this:

```
rsync -av --exclude-from={'list.txt'} sourcedir/ destinationdir/
```

The `rsync` tool skips all files and directories you list in the file. You can add any pattern we used in this guide.

The example file we used in the above command contains the following patterns:



## Conclusion

This guide showed you all `rsync` exclude examples you need when transferring data with this tool. You can combine the patterns to customize the command and exclude multiple files and directories with `rsync`.