

The du (Disk Usage) commands are most handy in Linux

du is a command line tool provided with Linux, which reports the amount of disk space used by directories and files. du stands for 'disk usage'.

du is a command line tool provided with Linux, which reports the amount of disk space used by directories and files. du stands for '**disk usage**'. This is the main tool for analyzing disk space in the command line.

How to use du command in Linux

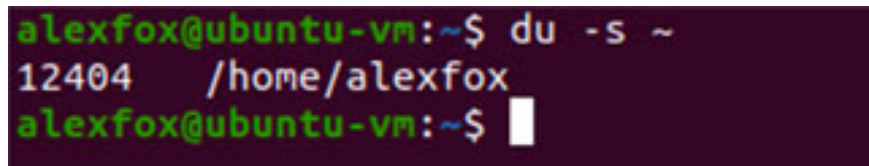
1. Basic usage
2. Control output
3. Change the capacity display
4. Include files and links
5. Exclude files, directories and links

Basic usage

```
du /path/to/directory
```

Run du with a directory to see the full analysis of the disk space that each folder holds. Each directory will be listed in turn in the standard output, so large directories or full system scans can generate tens of thousands of result streams and take a considerable amount of time. As a result, basic du commands are often added with flags (which will be shown below). Any of these flags can be combined with another flag, although some combinations are mutually exclusive because they are contradictory.

Control output



```
alexfox@ubuntu-vm:~$ du -s ~
12404 /home/alexfox
alexfox@ubuntu-vm:~$
```

```
du -c
```

Display a line at the bottom of the output to indicate the total amount of disk space used by the scanned folders.

```
du -s
```

Show only a summary of the total amount of disk space used by the specified directory. No output is displayed until the calculation process is complete.

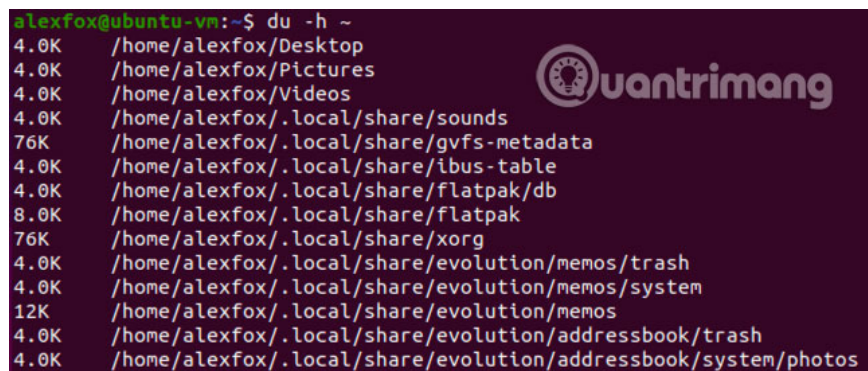
```
du > du-report.txt
```

People who use the command line will recognize this command, but not everyone. This command outputs the result of the command to a text file in the current directory, named '**du-report.txt**'.

```
du | less
```

This command produces cleaner and more readable results. You can browse content with fewer navigation shortcuts.

Change the capacity display



```
alexfox@ubuntu-vm:~$ du -h ~
4.0K  /home/alexfox/Desktop
4.0K  /home/alexfox/Pictures
4.0K  /home/alexfox/Videos
4.0K  /home/alexfox/.local/share/sounds
76K   /home/alexfox/.local/share/gvfs-metadata
4.0K  /home/alexfox/.local/share/ibus-table
4.0K  /home/alexfox/.local/share/flatpak/db
8.0K  /home/alexfox/.local/share/flatpak
76K   /home/alexfox/.local/share/xorg
4.0K  /home/alexfox/.local/share/evolution/memos/trash
4.0K  /home/alexfox/.local/share/evolution/memos/system
12K   /home/alexfox/.local/share/evolution/memos
4.0K  /home/alexfox/.local/share/evolution/addressbook/trash
4.0K  /home/alexfox/.local/share/evolution/addressbook/system/photos
```

Note that the file size will usually be rounded when specific units are specified. For example, the **4KB** file will be displayed as **0MB** if the **-m** flag is called, while the **750KB** file will be displayed as **1MB**.

```
du -h
```

Displayed in a readable format with the appropriate volumes listed, such as kilobytes, megabytes and gigabytes, instead of the standard block (block) size.

```
du -k
```

Displays the number of blocks in 1024-byte blocks (1 kilobyte).

```
du -m
```

Displays the number of blocks in blocks of 1,048,576 bytes (1 megabyte).

```
du -g
```

Displays the number of blocks in blocks 1,073,741,824 bytes (1 gigabyte).

Include files and links

```
alexfox@ubuntu-vm:~$ du -a ~
4      /home/alexfox/Desktop
4      /home/alexfox/.bash_logout
0      /home/alexfox/.sudo_as_admin_successful
4      /home/alexfox/Pictures
4      /home/alexfox/Videos
4      /home/alexfox/.local/share/sounds
4      /home/alexfox/.local/share/gvfs-metadata/home
4      /home/alexfox/.local/share/gvfs-metadata/trash:
32     /home/alexfox/.local/share/gvfs-metadata/trash:-33f6ece1.log
32     /home/alexfox/.local/share/gvfs-metadata/home-9710ce64.log
76     /home/alexfox/.local/share/gvfs-metadata
4      /home/alexfox/.local/share/ibus-table
4      /home/alexfox/.local/share/session_migration-ubuntu
```

`du -L`

Follow the symbolic links in the command line and file hierarchy. On the other hand, the space occupied by the symbolic link itself will be reported (usually the minimum in the file system) rather than the directory tree that the symbolic link points to.

`du -a`

Displays the disk usage of all files, not just folders.

`du /path/to/file.txt`

Displays the disk usage of a specific file, as named in the command.

Exclude files, directories and links

```
alexfox@ubuntu-vm:~$ du -d 3
4      ./Desktop
4      ./Pictures
4      ./Videos
4      ./local/share/sounds
76     ./local/share/gvfs-metadata
4      ./local/share/ibus-table
8      ./local/share/flatpak
76     ./local/share/xorg
156    ./local/share/evolution
4      ./local/share/applications
8      ./local/share/gnome-shell
8      ./local/share/icc
972    ./local/share/app-info
4      ./local/share/keyrings
4      ./local/share/gnome-settings-daemon
288    ./local/share/tracker
1620   ./local/share
1624   ./local
4      ./Public
4      ./Documents
4      ./Music
```

`du -X FILE`

Exclude files that match any pattern in the specified string.

```
du --exclude="*.o"
```

Exclude files and subdirectories according to the pattern. In this example, all folders including the `'*.o'` string will be ignored. Note that these are pattern shells, not regular expressions. Thus, the control characters are limited to `*`, matching any string with 0 or more characters and `?`, matches any one character. The above command will exclude these files from the directory capacity calculation process. If flag `-a` is used, the excluded file will be ignored in the output.

```
du --threshold=SIZE
```

Exclude items that are less than **SIZE** if it is a positive value or items greater than **SIZE** if it is a negative value. **SIZE** is an integer and has one unit (optional). For example, the `--threshold=1MB` command will ignore all files smaller than 1 megabyte (1000 ^ 2 bytes). Units including K, M, G, T, P, E, Z, Y represent kibibyte, mebibyte, gibibyte, etc. `.` or KB, MB, GB, TB, represent kilobytes, megabytes, gigabytes, etc. This can be a useful tool to find the largest files on the system with a command like `du --threshold=1GB`.

```
du -d N
```

Set the maximum depth for N folders. This flag can take any positive integer. With this setting, `du` will scan up to 2 subdirectories in the specified directory. If other subdirectories exist, they will not be scanned individually. Instead, their value will be included in the reported directories. Note that `-d 0` will report the same result as the `-s` flag.

For example, consider the directory path `'dir1 / dir2 / dir3 / dir4'`, containing 1 parent directory and 3 subdirectories. The command `du -d 2 dir1` will scan to **dir3**. **Dir3** will include files in **dir4**, although **dir4** is not listed separately.

The `du` command is useful when paired with other utilities, such as the `du -a / | sort -n -r | head -n 10` command `du -a / | sort -n -r | head -n 10` `du -a / | sort -n -r | head -n 10`. This command will search your entire file system (`du -a /`), sort results by size (`sort -n -r`), and then display only the top 10 results (`head -n 10`). Basically, this is a shortcut to the 10 largest files on your computer. Combine `du` with other commands via pipes to produce even more useful results.

Hope you are successful.

You finished reading the article "**The du (Disk Usage) commands are most handy in Linux**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.