

How to check and manage disk space in Linux

A common problem is that updates fail due to a lack of disk space, but there are some simple tests that users can perform when they encounter an error message, to keep critical systems smooth operation.

An essential skill required by system administrators is to maintain the 'health' of both online and offline systems. This is especially important on production servers. Downtime or incidents can cause data loss on these servers. A common problem is that updates fail due to a lack of disk space, but there are some simple tests that users can perform when they encounter an error message, to keep critical systems smooth operation.

There are two main commands that can be used:

1. **df** - This command reports the amount of disk space on the system.
2. **du** - This command shows the capacity used by specific files.

Each of the above 2 commands is a different way of checking and can combine both commands at the same time if necessary. Here are some examples to illustrate the use of these commands.

How to control disk space on Linux systems

1. Use the df command
2. Use du command
3. ncd - Alternative to du
4. Good 'cleanup' tools
 1. Autoremove
 2. Delete APT cache
5. GUI-based options
 1. Bleachbit

Use the df command

Open **Terminal** and type **df**, then press **Enter**. An output like the following will be displayed:

```
ubuntu@ubuntu: ~  
File Edit View Search Terminal Help  
ubuntu@ubuntu:~$ df  
Filesystem            1K-blocks    Used Available Use% Mounted on  
udev                  3997356      0  3997356   0% /dev  
tmpfs                 805820       1904   803916   1% /run  
/dev/mapper/ubuntu--vg-root 958622848 39144012 870713704   5% /  
tmpfs                 4029084      28624  4000460   1% /dev/shm  
tmpfs                  5120         4       5116   1% /run/lock  
tmpfs                 4029084      0  4029084   0% /sys/fs/cgroup  
/dev/loop10           144384      144384      0 100% /snap/gnome-3-26-1  
604/70  
/dev/loop11           35712       35712      0 100% /snap/core/6405  
/dev/loop9            35712       35712      0 100% /snap/gtk-common-t  
hemes/1122  
/dev/loop12           178176      178176      0 100% /snap/spotify/34  
/dev/loop1            35584       35584      0 100% /snap/gtk-common-t  
hemes/319  
/dev/loop13           14848       14848      0 100% /snap/gnome-logs/3  
7  
/dev/loop14           13312       13312      0 100% /snap/gnome-charac  
ters/103  
/dev/loop0            2432        2432      0 100% /snap/gnome-calcul  
ator/180  
/dev/loop8            3840        3840      0 100% /snap/gnome-system  
-monitor/51
```

As can be seen, the results show all drives connected to the system. This result may be somewhat lengthy, so users can collapse the results by specifying the drive that is working. In the examples in this article, the main drive is reported as '**dev / sda**' - The author has also included specific partitions with '**dev / sda1**' and '**dev / sda2**'.

```
ubuntu@ubuntu: ~  
File Edit View Search Terminal Help  
ubuntu@ubuntu:~$ df /dev/sda  
Filesystem            1K-blocks    Used Available Use% Mounted on  
udev                  3997356      0  3997356   0% /dev  
ubuntu@ubuntu:~$ df /dev/sda1  
Filesystem            1K-blocks    Used Available Use% Mounted on  
/dev/sda1             720368 149604  518372  23% /boot  
ubuntu@ubuntu:~$ df /dev/sda2  
Filesystem            1K-blocks    Used Available Use% Mounted on  
udev                  3997356      0  3997356   0% /dev  
ubuntu@ubuntu:~$
```

Can make **df** a little easier to read by typing:

```
df -h
```

```

ubuntu@ubuntu: ~
File Edit View Search Terminal Help
ubuntu@ubuntu:~$ df -h
Filesystem      Size  Used Avail Use% Mounted on
udev            3.9G   0 3.9G   0% /dev
tmpfs           787M   1.9M 786M   1% /run
/dev/mapper/ubun-vg-root 915G  38G 831G   5% /
tmpfs           3.9G   35M 3.9G   1% /dev/shm
tmpfs           5.0M   4.0K 5.0M   1% /run/lock
tmpfs           3.9G   0 3.9G   0% /sys/fs/cgroup
/dev/loop10    141M  141M   0 100% /snap/gnome-3-26-1604/70
/dev/loop11    91M   91M   0 100% /snap/core/6405
/dev/loop9     35M   35M   0 100% /snap/gtk-common-themes/1122
/dev/loop12    174M  174M   0 100% /snap/spotify/34
/dev/loop1     35M   35M   0 100% /snap/gtk-common-themes/319
/dev/loop13    15M   15M   0 100% /snap/gnome-logs/37
/dev/loop14    13M   13M   0 100% /snap/gnome-characters/103
/dev/loop0     2.4M   2.4M   0 100% /snap/gnome-calculator/180
/dev/loop8     3.8M   3.8M   0 100% /snap/gnome-system-monitor/51
/dev/loop2     87M   87M   0 100% /snap/core/4917
/dev/loop3     141M  141M   0 100% /snap/gnome-3-26-1604/78
/dev/loop4     2.3M   2.3M   0 100% /snap/gnome-calculator/260
/dev/loop5     3.8M   3.8M   0 100% /snap/gnome-system-monitor/57
/dev/loop6     15M   15M   0 100% /snap/gnome-logs/45
/dev/loop7     13M   13M   0 100% /snap/gnome-characters/139
/dev/sda1      704M  147M  507M  23% /boot

```

Users can collapse everything more by adding the **--output** flag . The parameters for this command are:

1. **source** - The **source** of the device mount point
2. **size** - Total blocks
3. **used** - Total number of blocks used
4. **avail** - Total number of blocks available
5. **Percent** - Percent of used disk space
6. **target** - **Mount** point for the device

In this example, the author uses only two parameters.

```
df --output=source,used,avail
```

```

ubuntu@ubuntu: ~
File Edit View Search Terminal Help
ubuntu@ubuntu:~$ df --output=source,used,avail
Filesystem      Used      Avail
udev            0      3997356
tmpfs           1908      803912
/dev/mapper/ubun-vg-root 39178876 870678840
tmpfs           38588     3990496
tmpfs           4         5116
tmpfs           0      4029084
/dev/loop10    144384     0
/dev/loop11    93184     0
/dev/loop9     35712     0
/dev/loop12    178176     0
/dev/loop1     35584     0
/dev/loop13    14848     0
/dev/loop14    13312     0
/dev/loop0     2432     0
/dev/loop8     3840     0
/dev/loop2     89088     0
/dev/loop3     144128     0
/dev/loop4     2304     0
/dev/loop5     3840     0
/dev/loop6     14976     0
/dev/loop7     13312     0
/dev/sda1     140608     510372

```

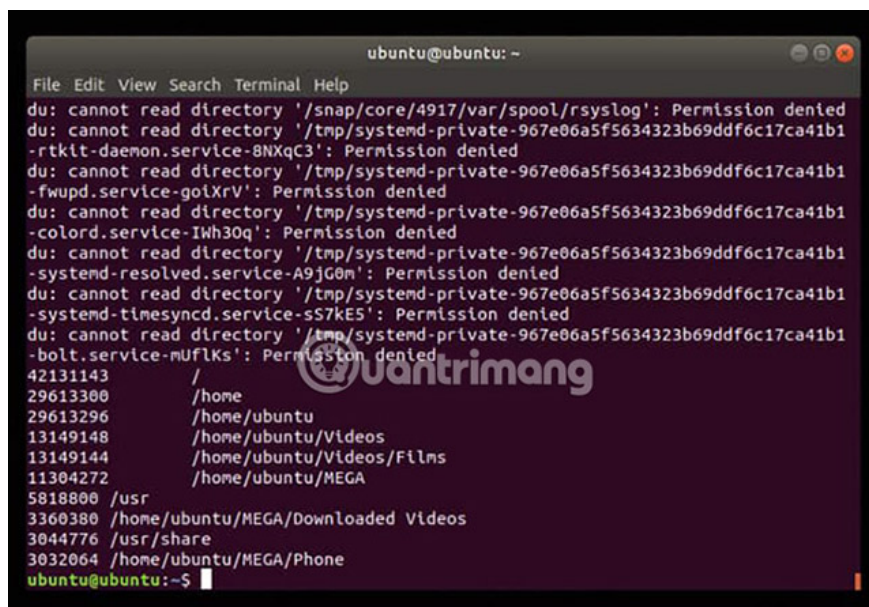
Use du command

Users may find that the drive is nearly full, but what causes this situation? This is the time to use the **du** command to display problem files. In a practical example, a user has detected a remote server with 98% disk space without any warning or real reason. Turns out, there are a lot of log files with java errors, with a total size of about 40GB. There is a lot of wasted disk space, so after using the **du** command, users can delete unnecessary files.

It should be warned that, if only typing **du**, the command will return all files and that can take a lot of time. Better define a few parameters. Let's find the 10 largest directories on the system.

```
du -a / | sort -n -r | head -n 10
```

This command will scan and create a similar type of result. Ignore warnings about rights at the present time.



```
ubuntu@ubuntu: ~  
File Edit View Search Terminal Help  
du: cannot read directory '/snap/core/4917/var/spool/rsyslog': Permission denied  
du: cannot read directory '/tmp/systemd-private-967e06a5f5634323b69ddf6c17ca41b1  
-rtkit-daemon.service-8NXqC3': Permission denied  
du: cannot read directory '/tmp/systemd-private-967e06a5f5634323b69ddf6c17ca41b1  
-fwupd.service-goIXrv': Permission denied  
du: cannot read directory '/tmp/systemd-private-967e06a5f5634323b69ddf6c17ca41b1  
-colord.service-IWh30q': Permission denied  
du: cannot read directory '/tmp/systemd-private-967e06a5f5634323b69ddf6c17ca41b1  
-systemd-resolved.service-A9jG0n': Permission denied  
du: cannot read directory '/tmp/systemd-private-967e06a5f5634323b69ddf6c17ca41b1  
-systemd-timesyncd.service-sS7kE5': Permission denied  
du: cannot read directory '/tmp/systemd-private-967e06a5f5634323b69ddf6c17ca41b1  
-bolt.service-mUfLkS': Permission denied  
42131143 /  
29613300 /home  
29613296 /home/ubuntu  
13149148 /home/ubuntu/Videos  
13149144 /home/ubuntu/Videos/Films  
11304272 /home/ubuntu/MEGA  
5818800 /usr  
3360380 /home/ubuntu/MEGA/Downloaded Videos  
3044776 /usr/share  
3032064 /home/ubuntu/MEGA/Phone  
ubuntu@ubuntu:~$
```

For a complete list of usable parameters, see the **man** page : http://linuxcommand.org/lc3_man_pages/du1.html

ncdu - Alternative to du

If you prefer a more interactive way to view hard drive information, try reviewing **ncdu**. Ncdu tool provides interface based on ncurses for travelers. This tool displays the same information as the travel command but in a more intuitive way. Ncdu also allows users to navigate between different folders with the arrow keys and use the **Enter** key to make a selection.

Users can install ncdu in Ubuntu using the command:

```
sudo apt install ncdu
```

To use ncdu, enter:

```
ncdu /directory-to-scan
```

Replace '**directory-to-scan**' to the actual directory you want to scan. For example, to scan the entire hard drive, enter:

```
ncdu /
```

```
ncdu 1.12 ~ Use the arrow keys to navigate, press ? for help
-----
. 3.7 GiB [#####] /home
. 1.7 GiB [####]  ] /var
. 1.1 GiB [##]   ] /usr
794.9 MiB [##]  ] /lib
139.2 MiB [ ]   ] /boot
80.5 MiB [ ]   ] /opt
14.9 MiB [ ]   ] /bin
14.5 MiB [ ]   ] /sbin
. 7.4 MiB [ ]   ] /etc
. 712.0 KiB [ ]  ] /root
. 32.0 KiB [ ]  ] /tmp
! 16.0 KiB [ ]  ] /lost+found
. 8.0 KiB [ ]   ] /media
. 8.0 KiB [ ]   ] /snap
. 4.0 KiB [ ]   ] /lib64
e 4.0 KiB [ ]   ] /srv
! 4.0 KiB [ ]   ] /root
e 4.0 KiB [ ]   ] /mnt
. 0.0 B [ ]    ] /proc
. 0.0 B [ ]    ] /sys
. 0.0 B [ ]    ] /dev
-----
Total disk usage: 7.5 GiB Apparent size: 128.0 TiB Items: 336952
```

Good 'cleanup' tools

In addition to the above commands, there are some basic things users can do to help reduce their disk usage to a minimum.

Autoremove

The most obvious way on Ubuntu-based systems is to check for outdated packages. In Terminal, users can enter:

```
sudo apt autoremove
```

When the password has been entered correctly, the system will start deleting the orphaned package (the Orphaned package is a generic term, meaning that the package has no purpose on the running system). It can also remove old kernels that take up space and are mostly unnecessary.

Delete APT cache

When Ubuntu downloads packages, it keeps apt files so they can be easily reinstalled if needed. These files take up disk space and this space can be recovered with:

```
sudo du -sh /var/cache/apt
```

Now, it is possible to clean these files with the following command:

```
sudo apt autoclean
```

GUI-based options

What if the user feels all of the above Terminal commands are too complicated, or simply doesn't like messing things up with the command line because they can damage the system? Thankfully, Linux has a number of GUI-based tools to use.

Bleachbit

This is a program available for both Windows and Linux systems. Bleachbit works in a similar way on these two platforms. Bleachbit can be downloaded here, but most are available in system repositories. After installation, users can run the tool by selecting the checkboxes.

As the reader has seen, checking the system and the drive capacity will help prevent problems. Users also need to clean the drive regularly.

Hope you are successful.

You finished reading the article "**How to check and manage disk space 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.