How to Move the Partition to a New Partition in the Linux Server?

written by sysadmin | 22 March 2025

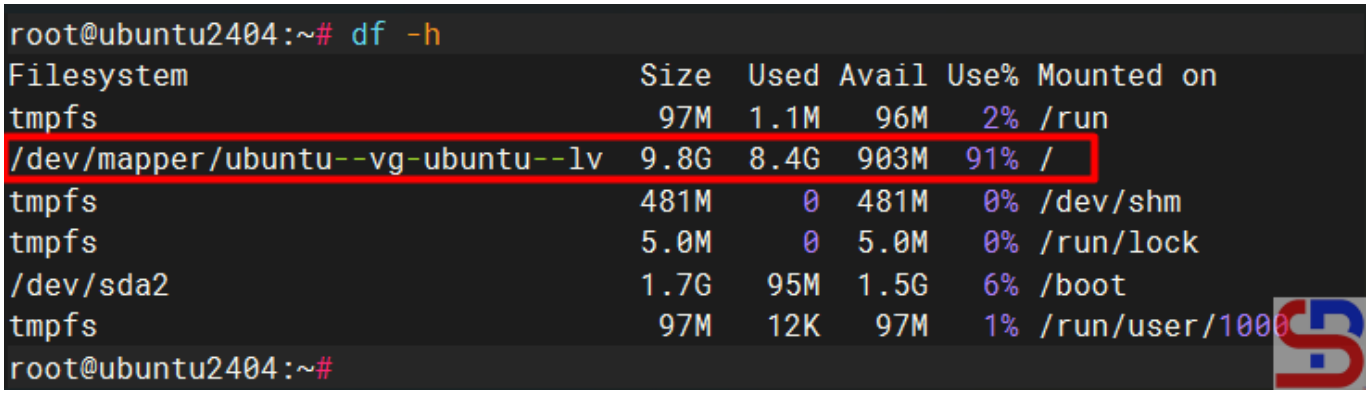
If you install a Linux server, you will usually install it with only one partition and not separate the other partitions. Problems will arise if one of these partitions uses a large enough hard disk, resulting in you running out of HDD space on your Linux server.

Problem

How to move the partition to a new partition in the Linux server?

Solution

In this article, I use the Ubuntu Server OS, and this article should be applied to any Linux distribution. Currently, the condition of the hard disk on my Ubuntu server is like the image below:



Condition of the hard disk in my Ubuntu server

From the image above, the root partition only has a free HDD of 9 percent. After I checked, it turned out that the cause was the /var partition, which took up a lot of hard disk so I want to move the **/var** partition to the new partition.

```
root@ubuntu2404:/# du -sh *
0       bin
4.0K    bin.usr-is-merged
95M     boot
4.0K    cdrom
0       dev
6.1M    etc
32K     home
0       lib
0       lib64
4.0K    lib.usr-is-merged
16K     lost+found
4.0K    media
4.0K    mnt
4.0K    opt
du: cannot access 'proc/1185/task/1185/fd/4': No such file or directory
du: cannot access 'proc/1185/task/1185/fdinfo/4': No such file or directory
du: cannot access 'proc/1185/fd/3': No such file or directory
du: cannot access 'proc/1185/fdinfo/3': No such file or directory
0       proc
32K     root
1.1M    run
0       sbin
4.0K    sbin.usr-is-merged
8.0K    snap
4.0K    srv
2.0G    swap.img
0       sys
64K     tmp
2.0G    usr
4.5G    var
root@ubuntu2404:/#
```

Check the largest partition size

Here are the steps to move the partition to a new partition in the Linux Server:

Info

The steps in this article will make your Linux server **enter maintenance mode** which means that the Linux server cannot be accessed from anywhere results in the application or database that may be in the Linux server also inaccessible. So discuss first with your boss if you want to do the steps in this article

1. Add a new hard drive

I insert a new 10 GB HDD into my Linux server. After that, I check if the new HDD is detected by Linux using the command:

```
fdisk -l
```

```
root@ubuntu2404:/# fdisk -l
Disk /dev/sda: 15 GiB, 16106127360 bytes, 31457280 sectors
Disk model: VBOX HARDDISK
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: 33862586-79A4-4B47-A641-CFC4AB6AF897

Device       Start      End  Sectors  Size Type
/dev/sda1     2048     4095     2048    1M BIOS boot
/dev/sda2     4096  3674111  3670016  1.8G Linux filesystem
/dev/sda3  3674112 31455231 27781120 13.2G Linux filesystem


Disk /dev/sdb: 10 GiB, 10737418240 bytes, 20971520 sectors
Disk model: VBOX HARDDISK
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes


Disk /dev/mapper/ubuntu--vg-ubuntu--lv: 10 GiB, 10737418240 bytes, 20971520 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
root@ubuntu2404:/#
```

Check the new HDD in the Linux server

From the image above, it can be seen that the new HDD was detected by Linux with a partition in sdb.

2. Create a new partition

Run the command below to create a new partition in Linux (Adjust to the hard disk partition detected on your Linux server after typing the **fdisk -l** command):

```
fdisk /dev/sdb
```

Press the **n** and **p** keys, then the number **,1** and **enter 2x,** then press the **w** button as seen in the image below:

```
root@ubuntu2404:/# fdisk /dev/sdb

Welcome to fdisk (util-linux 2.39.3).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Device does not contain a recognized partition table.
Created a new DOS (MBR) disklabel with disk identifier 0xf65c8ed5.

Command (m for help): n
Partition type
   p   primary (0 primary, 0 extended, 4 free)
   e   extended (container for logical partitions)
Select (default p): p
Partition number (1-4, default 1): 1
First sector (2048-20971519, default 2048):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (2048-20971519, default 20971519):

Created a new partition 1 of type 'Linux' and of size 10 GiB.

Command (m for help): w
The partition table has been altered.
Calling ioctl() to re-read partition table.
Syncing disks.

root@ubuntu2404:/#
```

Create a new partition in the new HDD

Then create a filesystem from the new HDD, and I want to use **ext4** for the filesystem of the new HDD using the command:

```
mkfs.ext4 /dev/sdb1
```

```
root@ubuntu2404:/# mkfs.ext4 /dev/sdb1
mke2fs 1.47.0 (5-Feb-2023)
Creating filesystem with 2621184 4k blocks and 655360 inodes
Filesystem UUID: 930db9c1-62d3-49ab-8482-713510fa2604
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632

Allocating group tables: done
Writing inode tables: done
Creating journal (16384 blocks): done
Writing superblocks and filesystem accounting information: done

root@ubuntu2404:/#
```

Create a filesystem in the partition of the new HDD

3. Create and mount a new folder

After that, create a new folder using the command:

```
mkdir /mnt/newvar
```

Then, mount the new partition to the new folder using the command:

```
mount /dev/sdb1 /mnt/newvar
```

```
root@ubuntu2404:/# mkdir /mnt/newvar
root@ubuntu2404:/#
root@ubuntu2404:/# mount /dev/sdb1 /mnt/newvar
root@ubuntu2404:/#
root@ubuntu2404:/# df -h
Filesystem                         Size  Used Avail Use% Mounted on
tmpfs                               97M  1.1M   96M   2% /run
/dev/mapper/ubuntu--vg-ubuntu--lv  9.8G  8.4G  905M  91% /
tmpfs                              481M     0  481M   0% /dev/shm
tmpfs                              5.0M     0  5.0M   0% /run/lock
/dev/sda2                          1.7G   95M  1.5G   6% /boot
tmpfs                               97M   12K   97M   1% /run/user/1000
/dev/sdb1                          9.8G   24K  9.3G   1% /mnt/newvar
root@ubuntu2404:/#
```

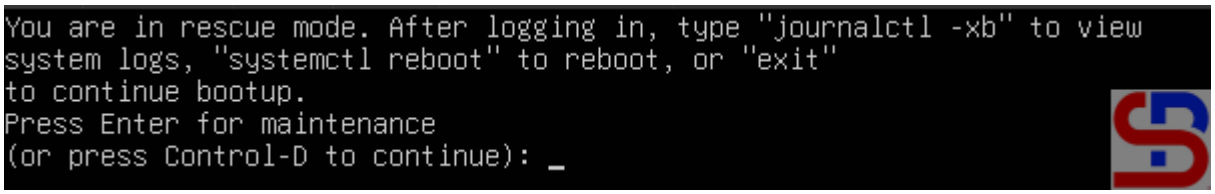
Create and mount a new folder

4. Enter maintenance mode

Type the command below:

```
init 1
```

to enter the rescue mode:



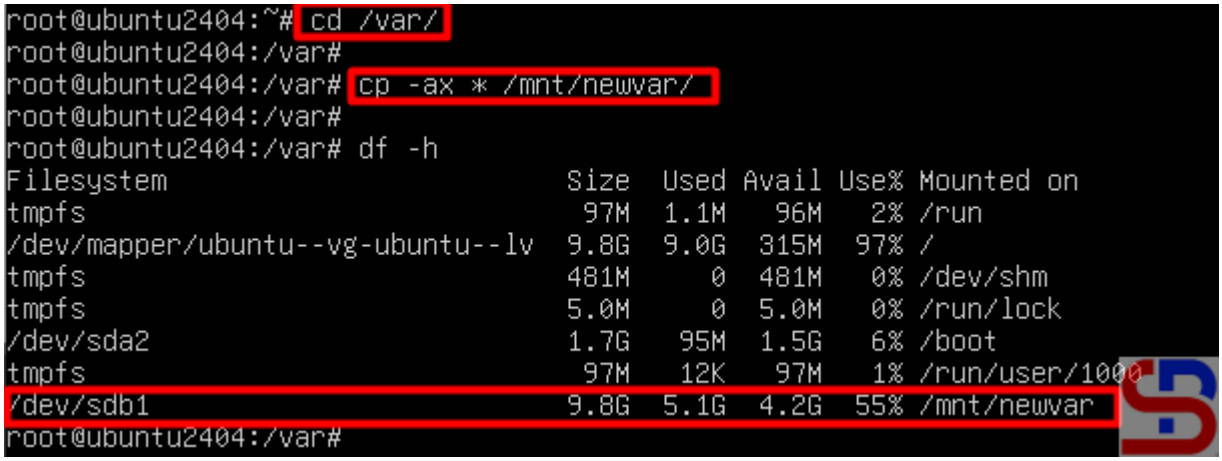
Enter the maintenance mode

After that, press the **Enter button** to enter maintenance mode.

5. Copy the folder

Go to the /var folder and copy all the files and folders in the folder into a new folder by typing the following commands:

```
cd /var
cp -ax * /mnt/newvar
```



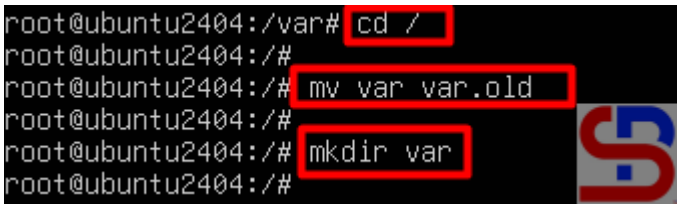
Copy the folder

6. Rename the folder

Once the copy process is complete, change the /var folder to the var.old folder and then create a new /var folder using

the command:

```
cd /
mv var var.old
mkdir /var
```



Rename the folder

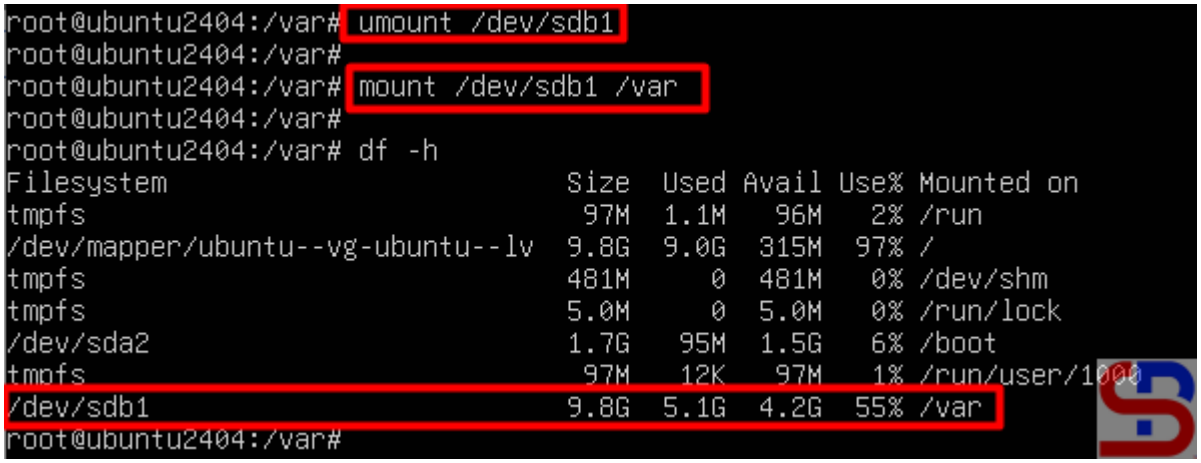
7. Mount the new folder

Next, do umount on the /sdb1 partition by using the command:

```
umount /dev/sdb1
```

And mount the /sdb1 partition to the new /var folder using the command:

```
mount /dev/sdb1 /var
```



Mount the new folder

8. Change the fstab file

Change the **/etc/fstab** file by adding the following script to the file:

```
/dev/sdb1  /var        ext4        defaults        0 0
```

```
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point>   <type>  <options>       <dump>  <pass>
# / was on /dev/ubuntu-vg/ubuntu-lv during curtin installation
/dev/disk/by-id/dm-uuid-LVM-g4VC0MaxbNzT5D0AVCzrf17icg24GBNlPwhuSsXpRZC80pJaXQMcFFctYEhUO0j5 / ext4 defaults 0 1
# /boot was on /dev/sda2 during curtin installation
/dev/disk/by-uuid/c59b0229-fcf2-4f2f-a6c7-e183c8ca6093 /boot ext4 defaults 0 1
/swap.img       none    swap    sw      0       0
/dev/sdb1       /var    ext4    defaults        0 0
```

Script additions in fstab file

9. Restart the server

After that, restart the Linux server and make sure there is no problem when the Linux server reboots.

10. Delete the folder

If the Linux server has finished restarting, then you can delete the var.old folder so that the size of the hard disk of the root partition increases by using the command:

```
cd /
rm -rf var.old
```

```
root@ubuntu2404:~# df -h
Filesystem                         Size  Used Avail Use% Mounted on
tmpfs                               97M  1.1M   96M   2% /run
/dev/mapper/ubuntu--vg-ubuntu--lv  9.8G  8.4G  905M  91% /   Before delete var.old folder
tmpfs                              481M     0  481M   0% /dev/shm
tmpfs                              5.0M     0  5.0M   0% /run/lock
/dev/sdb1                          9.8G  4.6G  4.8G  49% /var
/dev/sda2                          1.7G   95M  1.5G   6% /boot
tmpfs                               97M   12K   97M   1% /run/user/1000
root@ubuntu2404:~#
root@ubuntu2404:~# cd /
root@ubuntu2404:/#
root@ubuntu2404:/# rm -rf var.old
root@ubuntu2404:/#
root@ubuntu2404:/# df -h
Filesystem                         Size  Used Avail Use% Mounted on
tmpfs                               97M  1.1M   96M   2% /run
/dev/mapper/ubuntu--vg-ubuntu--lv  9.8G  3.9G  5.4G  42% /   After delete var.old folder
tmpfs                              481M     0  481M   0% /dev/shm
tmpfs                              5.0M     0  5.0M   0% /run/lock
/dev/sdb1                          9.8G  4.6G  4.8G  49% /var
/dev/sda2                          1.7G   95M  1.5G   6% /boot
tmpfs                               97M   12K   97M   1% /run/user/1000
root@ubuntu2404:/#
```

Before and after moving the partition

Note

Reboot the server again to make sure there are no problems after you delete the var.old folder. You can use the steps above when you want to move another folder to a new partition in the Linux server.

References

blog.oshim.net
phoenixnap.com