

## How to extend ext4 root partition without reboot

Written by BiRU

Monday, 01 August 2016 07:46 - Last Updated Monday, 01 August 2016 08:59

---

Let's say you want to extend existing disk on your already installed VM. Usually to extend the existing ext4 partition where system is running you would use some Live CD, to edit partition while it's unmounted. However it's possible to extend the partition without booting from Live CD. Here are some simple steps to do so:

Here we see that we have disk bigger than already existing partition:

```
root@test:~# df -h /
Filesystem Size Used Avail Use% Mounted on
/dev/vda1 24G 1008M 22G 5% /
```

```
root@test:~# fdisk -l /dev/vda
```

```
Disk /dev/vda: 53.7 GB, 53687091200 bytes
255 heads, 63 sectors/track, 6527 cylinders, total 104857600 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
Disk identifier: 0x0009256d
```

```
Device Boot Start End Blocks Id System
/dev/vda1 * 2048 50329215 25163584 83 Linux
/dev/vda2 50329216 52426367 1048576 82 Linux swap / Solaris
```

First of all you have to make sure to turn off swap:

```
root@test:~# swapoff -a
root@test:~# free -m
total      used      free     shared    buffers     cached
Mem:        994        189        804         0         11        140
-/+ buffers/cache:        38        955
Swap:         0         0         0
```

#Make sure that you see zeroes in "Swap:" row

And now let's remove swap partition and extend root partition. From previous fdisk command we can see that our swap partition is second one (/dev/vda2).

In following steps we're going to remove swap partition and root partition, create root partition with new size (size should be = [last sector] – [swap partition sector count]) and finally new swap partition with same size.

```
root@test:~# fdisk /dev/vda
```

## How to extend ext4 root partition without reboot

Written by BiRU

Monday, 01 August 2016 07:46 - Last Updated Monday, 01 August 2016 08:59

---

Command (m for help): p

Disk /dev/vda: 53.7 GB, 53687091200 bytes

255 heads, 63 sectors/track, 6527 cylinders, total 104857600 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

Disk identifier: 0x0009256d

Device	Boot	Start	End	Blocks	Id	System
/dev/vda1	*	2048	50329215	25163584	83	Linux
/dev/vda2		50329216	52426367	1048576	82	Linux swap / Solaris

Command (m for help): d

Partition number (1-4): 2

Command (m for help): d

Selected partition 1

Command (m for help): n

Partition type:

p primary (0 primary, 0 extended, 4 free)

e extended

Select (default p):

Using default response p

Partition number (1-4, default 1):

Using default value 1

First sector (2048-104857599, default 2048):

Using default value 2048

Last sector, +sectors or +size{K,M,G} (2048-104857599, default 104857599): +103809023

Command (m for help): n

Partition type:

p primary (1 primary, 0 extended, 3 free)

e extended

Select (default p):

Using default response p

Partition number (1-4, default 2):

Using default value 2

First sector (103811072-104857599, default 103811072):

Using default value 103811072

Last sector, +sectors or +size{K,M,G} (103811072-104857599, default 104857599):

Using default value 104857599

Command (m for help): p

## How to extend ext4 root partition without reboot

Written by BiRU

Monday, 01 August 2016 07:46 - Last Updated Monday, 01 August 2016 08:59

---

Disk /dev/vda: 53.7 GB, 53687091200 bytes  
255 heads, 63 sectors/track, 6527 cylinders, total 104857600 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  
Disk identifier: 0x0009256d

Device	Boot	Start	End	Blocks	Id	System
/dev/vda1		2048	103811071	51904512	83	Linux
/dev/vda2		103811072	104857599	523264	83	Linux

Before writing changes to disk we have to set proper partition types and make root partition bootable:

Command (m for help): t  
Partition number (1-4): 1  
Hex code (type L to list codes): 83

Command (m for help): t  
Partition number (1-4): 2  
Hex code (type L to list codes): 82  
Changed system type of partition 2 to 82 (Linux swap / Solaris)

Command (m for help): a  
Partition number (1-4): 1

Command (m for help): p

Disk /dev/vda: 53.7 GB, 53687091200 bytes  
255 heads, 63 sectors/track, 6527 cylinders, total 104857600 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  
Disk identifier: 0x0009256d

Device	Boot	Start	End	Blocks	Id	System
/dev/vda1	*	2048	103811071	51904512	83	Linux
/dev/vda2		103811072	104857599	523264	82	Linux swap / Solaris

Now let's write changes to disk and re-read partition table (to re-read partition table you can also reboot the VM) so the system could see new size:

Command (m for help): w  
The partition table has been altered!

Calling ioctl() to re-read partition table.

## How to extend ext4 root partition without reboot

Written by BiRU

Monday, 01 August 2016 07:46 - Last Updated Monday, 01 August 2016 08:59

---

WARNING: Re-reading the partition table failed with error 16: Device or resource busy.  
The kernel still uses the old table. The new table will be used at  
the next reboot or after you run partprobe(8) or kpartx(8)

Syncing disks.

```
root@test:~# partprobe
```

```
root@test:~# fdisk -l
```

Disk /dev/vda: 53.7 GB, 53687091200 bytes

16 heads, 63 sectors/track, 104025 cylinders, total 104857600 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

Disk identifier: 0x0009256d

Device	Boot	Start	End	Blocks	Id	System
/dev/vda1	*	2048	103811071	51904512	83	Linux
/dev/vda2		103811072	104857599	523264	82	Linux swap / Solaris

Now we can see that we have new partition table with resized partitions, however root partition's filesystem has to be resized as well. It's one simple step left:

```
root@test:~# resize2fs /dev/vda1
```

```
resize2fs 1.42.9 (4-Feb-2014)
```

Filesystem at /dev/vda1 is mounted on /; on-line resizing required

old\_desc\_blocks = 2, new\_desc\_blocks = 4

The filesystem on /dev/vda1 is now 12976128 blocks long.

```
root@test:~# df -h /
```

Filesystem	Size	Used	Avail	Use%	Mounted on
/dev/vda1	49G	1009M	46G	3%	/

Resize2fs did the trick and you can see that we have 49G partition now.