**Extend existing unencrypted logical volumes.**

**Prerequisites**

This tutorial requires **200MB** of available disk space.

Before extending any unencrypted logical volume, let’s create one of **100MB** called **lv\_vol** in the **vg** volume group:

# **lvcreate --size 100M --name lv\_vol vg**

Only unencrypted **Ext4** and **XFS** file systems can be extended, not **vfat**.

Note1: In this tutorial, the **lvextend** and **lvreduce** commands are used to respectively increase and decrease the size of a logical volume. For brevity’s sake, the **lvresize** command is not discussed here, but the same operations can be done in a slightly different way with this command (see comments).  
Note2: With **RHEL 7.3**, the **NetworkManager** service got two new directives in its unit file: **ProtectSystem=true** and **ProtectHome=read-only**. These directives forbid some changes made to system (**/usr**, **/boot**, **/root**, **/run/user**) and **/home** directories, mainly the creation of symbolic links to them (see details [here](http://www.spinics.net/lists/centos/msg163489.html)). For most operations, this shouldn’t change anything, but it’s better to know that.

**Ext4 logical volume extension**

To create an **ext4** file system on the previously created logical volume, type:

# **mkfs.ext4 /dev/vg/lv\_vol**

mke2fs 1.42.9 (28-Dec-2013)

Filesystem label=

OS type: Linux

Block size=1024 (log=0)

Fragment size=1024 (log=0)

Stride=0 blocks, Stripe width=0 blocks

25688 inodes, 102400 blocks

5120 blocks (5.00%) reserved for the super user

First data block=1

Maximum filesystem blocks=33685504

13 block groups

8192 blocks per group, 8192 fragments per group

1976 inodes per group

Superblock backups stored on blocks:

        8193, 24577, 40961, 57345, 73729

Allocating group tables: done

Writing inode tables: done

Creating journal (4096 blocks): done

Writing superblocks and filesystem accounting information: done

Optionally, mount the new file system on **/mnt**:

# **mount /dev/vg/lv\_vol /mnt**

To allocate all the space available in the volume group to an unencrypted **Ext4**-formatted logical volume (here **/dev/vg/lv\_vol**), type:

# **lvextend -l +100%FREE -r /dev/vg/lv\_vol**

Note: The file system is automatically extended (**-r**) without any need to unmount it.

To extend the size of an unencrypted **Ext4**-formatted logical volume (here **/dev/vg/lv\_vol**) by **50MB**, type:

# **lvextend --size +50M -r /dev/vg/lv\_vol**

  Rounding size to boundary between physical extents: 52.00 MiB

Extending logical volume lv\_vol to 152.00 MiB

Logical volume lv\_vol successfully resized

resize2fs 1.42.9 (28-Dec-2013)

Filesystem at /dev/mapper/vg-lv\_vol is mounted on /mnt; on-line resizing required

old\_desc\_blocks = 1, new\_desc\_blocks = 2

The filesystem on /dev/mapper/vg-lv\_vol is now 155648 blocks long.

**Although there is no reason to do so**, you can still do it in two steps:

# **lvextend --size +50M /dev/vg/lv\_vol**

  Rounding size to boundary between physical extents: 52.00 MiB

  Extending logical volume lv\_vol to 152.00 MiB

  Logical volume lv\_vol successfully resized

# **resize2fs /dev/vg/lv\_vol**

resize2fs 1.42.9 (28-Dec-2013)

Filesystem at /dev/vg/lv\_vol is mounted on /mnt; on-line resizing required

old\_desc\_blocks = 1, new\_desc\_blocks = 2

The filesystem on /dev/vg/lv\_vol is now 155648 blocks long.

**Ext4 logical volume reduction**

**Conversely**, to reduce the size of a logical volume (here by **50MB**), you have to follow these steps:  
Unmount the file system (here **/dev/vg/lv\_vol**):

# **umount /dev/vg/lv\_vol**

Reduce the size of the logical volume (here **/dev/vg/lv\_vol**) and the associated file system at the same time (**-r**):

# **lvreduce --size -50M -r /dev/vg/lv\_vol**

  Rounding size to boundary between physical extents: 48.00 MiB

fsck from util-linux 2.23.2

/dev/mapper/vg-lv\_vol: 11/37544 files (9.1% non-contiguous), 10390/155648 blocks

resize2fs 1.42.9 (28-Dec-2013)

Resizing the filesystem on /dev/mapper/vg-lv\_vol to 106496 (1k) blocks.

The filesystem on /dev/mapper/vg-lv\_vol is now 106496 blocks long.

  Reducing logical volume lv\_vol to 104.00 MiB

  Logical volume lv\_vol successfully resized

Mount the file system (here **/dev/vg/lv\_vol**):

# **mount /dev/vg/lv\_vol /mnt**

**XFS logical volume extension**

To create a **XFS** file system on the previously created logical volume called **lv\_vol** (see **prerequisites**), type:

# **mkfs.xfs /dev/vg/lv\_vol**

meta-data=/dev/vg/lv\_vol isize=256 agcount=4, agsize=6400 blks

= sectsz=512 attr=2, projid32bit=1

= crc=0

data = bsize=4096 blocks=25600, imaxpct=25

= sunit=0 swidth=0 blks

naming =version 2 bsize=4096 ascii-ci=0 ftype=0

log =internal log bsize=4096 blocks=853, version=2

= sectsz=512 sunit=0 blks, lazy-count=1

realtime =none extsz=4096 blocks=0, rtextents=0

Mount the new file system on **/mnt**:

# **mount /dev/vg/lv\_vol /mnt**

To extend the size of an unencrypted **XFS**-formatted logical volume (here **/dev/vg/lv\_vol**) by **50MB**, type:

# **lvextend --size +50M -r /dev/vg/lv\_vol**

Rounding size to boundary between physical extents: 52.00 MiB

Extending logical volume lv\_vol to 152.00 MiB

Logical volume lv\_vol successfully resized

meta-data=/dev/mapper/vg-lv\_vol isize=256 agcount=4, agsize=6400 blks

= sectsz=512 attr=2, projid32bit=1

= crc=0

data = bsize=4096 blocks=25600, imaxpct=25

= sunit=0 swidth=0 blks

naming =version 2 bsize=4096 ascii-ci=0 ftype=0

log =internal bsize=4096 blocks=853, version=2

= sectsz=512 sunit=0 blks, lazy-count=1

realtime =none extsz=4096 blocks=0, rtextents=0

data blocks changed from 25600 to 38912

Note1: The file system is automatically extended (**-r**) without any need to unmount it.  
Note2: **You can’t reduce a XFS file system even though you unmount it. You have to back it up (with tar or another tool), drop it and recreate it.**

**Although there is no reason to do so**, you can still do it in two steps:

# **lvextend --size +50M /dev/vg/lv\_vol**

  Rounding size to boundary between physical extents: 52.00 MiB

  Extending logical volume lv\_vol to 152.00 MiB

  Logical volume lv\_vol successfully resized

# **xfs\_growfs /mnt**

meta-data=/dev/mapper/vg-lv\_vol isize=256    agcount=4, agsize=6400 blks

         =                       sectsz=512   attr=2, projid32bit=1

         =                       crc=0

data     =                       bsize=4096   blocks=25600, imaxpct=25

         =                       sunit=0      swidth=0 blks

naming   =version 2              bsize=4096   ascii-ci=0 ftype=0

log      =internal               bsize=4096   blocks=853, version=2

         =                       sectsz=512   sunit=0 blks, lazy-count=1

realtime =none                   extsz=4096   blocks=0, rtextents=0

data blocks changed from 25600 to 38912

Note: The **xfs\_growfs** command accepts only a mounted **XFS** filesystem as parameter.