

Работа с диском

Есть жесткий диск, нужно добавить в систему.

Смотрим устройства в системе

```
# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
sda 8:0 0 298,1G 0 disk
└─sda1 8:1 0 298,1G 0 part
sdb 8:16 0 149,1G 0 disk
└─sdb1 8:17 0 149,1G 0 part
sdc 8:32 0 1,8T 0 disk
├─sdc1 8:33 0 1G 0 part /boot
└─sdc2 8:34 0 1,8T 0 part
   ├─centos-root 253:0 0 20G 0 lvm /
   ├─centos-swap 253:1 0 2G 0 lvm [SWAP]
   └─centos-home 253:2 0 1,8T 0 lvm /home
```

Форматируем в xfs (создаем файловую систему)

```
# mkfs.xfs /dev/sdb
mkfs.xfs: /dev/sdb appears to contain a partition table (gpt).
mkfs.xfs: Use the -f option to force overwrite.
```

```
# mkfs.xfs -f /dev/sdb
meta-data=/dev/sdb isize=512 agcount=4, agsize=9768182 blks
= sectsz=512 attr=2, projid32bit=1
= crc=1 finobt=0, sparse=0
data = bsize=4096 blocks=39072726, imaxpct=25
= sunit=0 swidth=0 blks
naming =version 2 bsize=4096 ascii-ci=0 ftype=1
log =internal log bsize=4096 blocks=19078, version=2
= sectsz=512 sunit=0 blks, lazy-count=1
realtime =none extsz=4096 blocks=0, rtextents=0
```

Проверяем

```
# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
sda 8:0 0 298,1G 0 disk
└─sda1 8:1 0 298,1G 0 part
```

```
sdb 8:16 0 149,1G 0 disk
sdc 8:32 0 1,8T 0 disk
├─sdc1 8:33 0 1G 0 part /boot
└─sdc2 8:34 0 1,8T 0 part
   ├─centos-root 253:0 0 20G 0 lvm /
   ├─centos-swap 253:1 0 2G 0 lvm [SWAP]
   └─centos-home 253:2 0 1,8T 0 lvm /home
```

Автоматическое монтирование диска при старте системы

```
# nano /etc/fstab
/dev/sdb /home/data xfs defaults 0 0
```

LVM

Есть диск для мусора. Отформатировать в xfs и накатить LVM разбив на два раздела

Инициализируем его для работы LVM

```
$ sudo pvcreate /dev/sda
```

Чтобы посмотреть действительно ли были созданы физические тома LVM

```
$ sudo pvscan
PV /dev/sdc2 VG centos lvm2 [<1,82 TiB / 4,00 MiB free]
PV /dev/sda lvm2 [298,09 GiB]
Total: 2 [<2,11 TiB] / in use: 1 [<1,82 TiB] / in no VG: 1 [298,09 GiB]
```

Можно посмотреть физические LVM разделы с более подробными атрибутами, такими как размер, общий размер, физический размер, свободное место

```
$ sudo pvdisplay
"/dev/sda" is a new physical volume of "298,09 GiB"
--- NEW Physical volume ---
PV Name /dev/sda
VG Name
PV Size 298,09 GiB
Allocatable NO
```

```
PE Size 0
Total PE 0
Free PE 0
Allocated PE 0
PV UUID 2EKU2q-q4kq-o6ui-k75t-h5Km-1r6s-gfJiDx
```

PE Size – это размер одного блока LVM, по умолчанию он равен четырем мегабайтам, но это значение можно изменить.

СОЗДАНИЕ ГРУППЫ РАЗДЕЛОВ LVM

Группа томов – это не что иное, как пул памяти, который будет распределен между логическими томами и может состоять из нескольких физических разделов. После того как физические разделы инициализированы, вы можете создать из них группу томов (Volume Group, VG):

```
$ sudo vgcreate vg_box /dev/sda
```

Смотрим

```
$ sudo vgs
--- Volume group ---
VG Name vg_box
System ID
Format lvm2
Metadata Areas 1
Metadata Sequence No 1
VG Access read/write
VG Status resizable
MAX LV 0
Cur LV 0
Open LV 0
Max PV 0
Cur PV 1
Act PV 1
VG Size <298,09 GiB
PE Size 4,00 MiB
Total PE 76311
Alloc PE / Size 0 / 0
Free PE / Size 76311 / <298,09 GiB
VG UUID 5eSdwC-7y8N-vf3l-QQA7-Trpu-fcSj-3sAFx3
```

```
--- Volume group ---
VG Name centos
System ID
Format lvm2
Metadata Areas 1
Metadata Sequence No 4
VG Access read/write
VG Status resizable
MAX LV 0
Cur LV 3
Open LV 3
Max PV 0
Cur PV 1
Act PV 1
VG Size <1,82 TiB
PE Size 4,00 MiB
Total PE 476675
Alloc PE / Size 476674 / <1,82 TiB
Free PE / Size 1 / 4,00 MiB
VG UUID uQKzpe-GhWW-rr3C-xSGG-cTyw-VLBy-L1Sftb
```

СОЗДАНИЕ ЛОГИЧЕСКИХ ТОМОВ LVM

Теперь все готово, чтобы создать логические LVM разделы. Для этого будем использовать команду `lvcreate` и создадим логический раздел размером 240000 Mb, или 60000 блоков LVM (1 блок 4000Kb):

```
$ sudo lvcreate -l 60000 -n lv_box1 vg_box
Logical volume "lv_box1" created.
```

Посмотреть список доступных логических разделов LVM можно по аналогии

с предыдущими разделами с помощью команды `lvdisplay`:

```
$ sudo lvdisplay
--- Logical volume ---
LV Path /dev/vg_box/lv_box1
LV Name lv_box1
VG Name vg_box
LV UUID 2rAl80-wMo2-Q3X6-zz48-Pdxy-E3Gk-Qros51
LV Write Access read/write
```

```
LV Creation host, time tst.tst-amo.net.ua, 2018-08-19 18:56:33
+0300
LV Status available
# open 0
LV Size <234,38 GiB
Current LE 60000
Segments 1
Allocation inherit
Read ahead sectors auto
- currently set to 256
Block device 253:3
```

Теперь, когда раздел создан, мы можем работать с ним как с обычным разделом.

Например, отформатируем его в файловую систему ext4, а затем примонтируем в /mnt:

```
$ sudo mkfs.ext4 /dev/vg_box/lv_box1
$ sudo mount /dev/vg_box/lv_box1 /mnt/
# umount /mnt
```

ТИПЫ РАЗДЕЛОВ

lvm разделы могут быть трех типов:

- Линейные разделы (linear volume)
- Полосные разделы (Striped Volume)
- Зеркалированные разделы (Mirrored Volume)

Линейные разделы – это обычные LVM тома, они могут быть созданы как их одного, так и нескольких физических дисков. Например, если у вас есть два диска по 2 гигабайта, то вы можете их объединить и в результате получите один раздел LVM linux, размером 4 гигабайта. По умолчанию используются именно линейные LVM разделы.

Полосные разделы очень полезны при больших нагрузках на жесткий диск. Здесь вы можете настроить одновременную запись на разные физические устройства, для одновременных операций, это может очень сильно увеличить производительность работы системы.

Для этого нужно задать количество полос записи с помощью опции `-i`, а также размер полосы опцией `-l`. Количество полос не должно превышать количества физических дисков. Например:

```
$ lvcreate -L 1G -i2 -l64 -n lv_stripe vol_grp1
```

Зеркалированный том позволяет записывать данные одновременно на два устройства. Когда данные пишутся на один диск, они сразу же копируются на другой. Это позволяет защититься от сбоя одного из дисков. Если один из дисков испортится, то разделы LVM просто станут линейными и все данные по-прежнему будут доступны. Для создания такого раздела LVM Linux можно использовать команду:

```
$ sudo lvcreate -L 200M -m1 -n lv_mirror vol_grp1
```

УДАЛИТЬ LVM РАЗДЕЛ

Вы можете не только настроить lvm тома изменяя их размер и атрибуты, но и удалить lvm раздел, если он вам больше не нужен. Это делается с помощью lvm команды `lvremove`:

```
# lvremove /dev/vg_box/lv_box1
```

Или полностью группу:

```
# vgremove /dev/vg_box  
# lvdisplay /dev/vg_box/lv_box1
```

Чтобы изменить размер lvm используйте команду `lvextend`. Например, lvm увеличим размер диска из 80 до 100 мегабайт:

```
$ sudo lvextend -L100 /dev/vol_grp1/logical_vol1  
Extending logical volume logical_vol1 to 100.00 MB  
Logical volume logical_vol1 successfully resize
```

Также вы можете увеличить размер lvm, использовав знак `+`:

```
$ sudo lvextend -L+100 /dev/vol_grp1/logical_vol1  
Extending logical volume logical_vol1 to 200.00 MB  
Logical volume logical_vol1 successfully resized
```

Точно так же можно выполнить уменьшение lvm раздела с помощью

знака - .

https://wiki.russianfedora.pro/index.php?title=Уменьшение_размера_LVM_раздела

lvreduce урезает размер LVM раздела. Работает она следующим образом, параметр -L указывает на изменение размера раздела, здесь же есть 2 варианта: указание размера и относительное значение, со знаком + или -. Например, -L 83G урежет размер до 83G(то бишь lvm_root станет 83Gb вместо 98Gb), соответственно -L -15G или -L +15G уменьшит или увеличит размер на указанный (lvm_root станет 98G минус 15G или 98G плюс 15G).

Уменьшение тома на 2Gb

```
# lvreduce -L -2G /dev/vg_box/lv_box2
```

```
# lvreduce -r -L -2G /dev/vg_box/lv_box2
```

Наличие опции **-r** указывает, что надо также изменить размер файловой системы, без неё данные будут утеряны (не касается xfs).

Проверить наличие свободного места

```
# vgs vg_box
VG #PV #LV #SN Attr VSize VFree
vg_box 1 2 0 wz--n- <298,09g 0
```

Если xfs не основана на LVM нужно использовать

```
# xfs_growfs <mount_point> -D <size>
```

-D size – задается выборочный размер, без этой опции размер будет задан максимальный поддерживаемый устройством.