

HDD Resize LV

В нашем примере мы будем добавлять места в раздел **/db**

Предварительные действия:

Устанавливаем необходимое ПО для работы:

```
yum install system-storage-manager -y
```

Останавливаем mysqld и выключаем автозагрузку:

```
systemctl stop mysqld  
systemctl disable mysqld
```

Проверяем состояние mysqld

```
systemctl status mysqld
```

Вывод остановленной службы должен быть следующим

```
[root@localhost ~]# systemctl status mysqld  
• mysqld.service - MySQL Server  
  Loaded: loaded (/usr/lib/systemd/system/mysqld.service; disabled; vendor preset: disabled)  
  Active: inactive (dead)  
    Docs: man:mysqld(8)  
          http://dev.mysql.com/doc/refman/en/using-systemd.html
```

Смотрим текущую ситуацию:

```
df -h
```

Пример вывода:

Filesystem	Size	Used	Avail	Use%	Mounted on
/dev/mapper/ol-root	38G	3.7G	34G	10%	/
devtmpfs	988M	0	988M	0%	/dev
tmpfs	998M	0	998M	0%	/dev/shm
tmpfs	998M	33M	966M	4%	/run
tmpfs	998M	0	998M	0%	/sys/fs/cgroup
/dev/sda1	497M	237M	261M	48%	/boot
/dev/mapper/vg_db-lv_db	300G	358M	300G	1%	/db
tmpfs	200M	0	200M	0%	/run/user/0

Здесь видно:

- /db является Логическим томом(lv_db)
- Находится в группе логических томов(vg_db)
- Выделенное под него место - 300Gb

Рассмотрим детально командой:

```
ssm list
```

Пример вывода:

```
[root@localhost ~]# ssm list
-----
Device           Free           Used           Total  Pool  Mount point
-----
/dev/fd0                4.00 KB
/dev/sda                40.00 GB          PARTITIONED
/dev/sda1              500.00 MB          /boot
/dev/sda2  44.00 MB   39.46 GB   39.51 GB   ol
/dev/sdb                300.00 GB
/dev/sdb1   0.00 KB  200.00 GB  200.00 GB  vg_db
/dev/sdb2   0.00 KB  100.00 GB  100.00 GB  vg_db
-----

Pool  Type  Devices      Free           Used           Total
-----
ol    lvm   1           44.00 MB   39.46 GB   39.51 GB
vg_db lvm   2           0.00 KB   299.99 GB  299.99 GB
-----

Volume           Pool  Volume size  FS      FS size      Free  Type      Mount point
-----
/dev/ol/root      ol      37.46 GB  xfs     37.45 GB   33.79 GB  linear  /
/dev/ol/swap      ol       2.00 GB                linear
/dev/vg_db/lv_db  vg_db   299.99 GB  xfs     299.89 GB  299.48 GB  linear  /db
/dev/sda1                500.00 MB  xfs     496.67 MB  285.15 MB  part    /boot
-----
```

Здесь видно:

- В системе 2 диска: /dev/sda и /dev/sdb
- На /dev/sdb создано два раздела /dev/sdb1 и /dev/sdb2
- Оба раздела подключены к vg_db и образуют lv_db который смонтирован как /db

Процедура расширения

Выключаем виртуальную машину

```
poweroff
```

Делаем снапшот VM.

Увеличиваем размер HDD через интерфейс администратора на необходимое кол-во.

Включаем VM, подключаемся по ssh.

Просматриваем что получилось:

```
fdisk -l
```

Пример вывода:

```
[root@localhost ~]# fdisk -l
```

```
Disk /dev/sda: 42.9 GB, 42949672960 bytes, 83886080 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 label type: dos
Disk identifier: 0x00ad626
```

Device	Boot	Start	End	Blocks	Id	System
/dev/sda1	*	2048	1026047	512000	83	Linux
/dev/sda2		1026048	83886079	41430016	8e	Linux LVM

```
Disk /dev/sdb: 429.5 GB, 429496729600 bytes, 838860800 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 label type: dos
Disk identifier: 0x04652757
```

Device	Boot	Start	End	Blocks	Id	System
/dev/sdb1		2048	419430399	209714176	83	Linux
/dev/sdb2		419430400	629145599	104857600	8e	Linux LVM

```
Disk /dev/mapper/ol-root: 40.2 GB, 40227569664 bytes, 78569472 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 /dev/mapper/ol-swap: 2147 MB, 2147483648 bytes, 4194304 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 /dev/mapper/vg_db-lv_db: 322.1 GB, 322114158592 bytes, 629129216 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
```

Видим:

- Диск `/dev/sdb` теперь обладает емкостью 429.5 GB (Gigabyte)
- Наш раздел `/dev/mapper/vg_db-lv_db`: 322.1 GB (Gigabyte)

Переходим к созданию новой партии на неразмеченной области через `fdisk`:

```
fdisk /dev/sdb
```

Процесс создания:

```
[root@localhost ~]# fdisk /dev/sdb
Welcome to fdisk (util-linux 2.23.2).

Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Command (m for help): p

Disk /dev/sdb: 429.5 GB, 429496729600 bytes, 838860800 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 label type: dos
Disk identifier: 0x04652757
```

Device	Boot	Start	End	Blocks	Id	System
/dev/sdb1		2048	419430399	209714176	83	Linux
/dev/sdb2		419430400	629145599	104857600	8e	Linux LVM

```
Command (m for help): n
Partition type:
  p   primary (2 primary, 0 extended, 2 free)
  e   extended
Select (default p): p
Partition number (3,4, default 3): 3
First sector (629145600-838860799, default 629145600):
Using default value 629145600
Last sector, +sectors or +size{K,M,G} (629145600-838860799, default 838860799):
Using default value 838860799
Partition 3 of type Linux and of size 100 GiB is set

Command (m for help):
```

С помощью этой же утилиты меняем тип раздела (partition's system id):

```
Command (m for help): t
Partition number (1-3, default 3): 3
Hex code (type L to list all codes): 8e
Changed type of partition 'Linux' to 'Linux LVM'

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

Calling ioctl() to re-read partition table.

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@localhost ~]#
```

Для применения изменений необходимо перезагрузить VM:

```
shutdown -r now
```

Проверяем успешность действий командой:

```
fdisk -l
```

Видим что создался 3 раздел на диске /dev/sdb:

```
[root@localhost ~]# fdisk -l
```

```
Disk /dev/sda: 42.9 GB, 42949672960 bytes, 83886080 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 label type: dos
Disk identifier: 0x000ad626
```

Device	Boot	Start	End	Blocks	Id	System
/dev/sda1	*	2048	1026047	512000	83	Linux
/dev/sda2		1026048	83886079	41430016	8e	Linux LVM

```
Disk /dev/sdb: 429.5 GB, 429496729600 bytes, 838860800 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 label type: dos
Disk identifier: 0x04652757
```

Device	Boot	Start	End	Blocks	Id	System
/dev/sdb1		2048	419430399	209714176	83	Linux
/dev/sdb2		419430400	629145599	104857600	8e	Linux LVM
/dev/sdb3		629145600	838860799	104857600	8e	Linux LVM

```
Disk /dev/mapper/ol-root: 40.2 GB, 40227569664 bytes, 78569472 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 /dev/mapper/ol-swap: 2147 MB, 2147483648 bytes, 4194304 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 /dev/mapper/vg_db-lv_db: 322.1 GB, 322114158592 bytes, 629129216 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@localhost ~]#
```

Создаем физический том на новом разделе:

```
pvccreate /dev/sdb3
```

При успешном выполнении вывод будет следующий:

```
[root@localhost ~]# pvcreate /dev/sdb3
Physical volume "/dev/sdb3" successfully created.
[root@localhost ~]#
```

Размонтируем /db:

```
umount /db
```

Надо добавить новый физический том в VG раздела /db, ранее выяснили её наименование - „vg_db“, (можно и уточнить командой **vgdisplay**, **lvdisplay**):

```
vgextend vg_db /dev/sdb3
```

При успешном выполнении вывод будет следующий:

```
[root@localhost ~]# vgextend vg_db /dev/sdb3
Volume group "vg_db" successfully extended
[root@localhost ~]#
```

Производим ресайз lv_db с учетом добавленного свободного места:

```
ssm resize -s+100%FREE /dev/vg_db/lv_db
```

При успешном выполнении вывод будет следующий:

```
[root@localhost ~]# ssm resize -s+100%FREE /dev/vg_db/lv_db
Phase 1 - find and verify superblock...
Phase 2 - using internal log
    - zero log...
    - scan filesystem freespace and inode maps...
    - found root inode chunk
Phase 3 - for each AG...
    - scan (but don't clear) agi unlinked lists...
    - process known inodes and perform inode discovery...
    - agno = 0
    - agno = 1
    - agno = 2
    - agno = 3
    - agno = 4
    - agno = 5
    - process newly discovered inodes...
Phase 4 - check for duplicate blocks...
    - setting up duplicate extent list...
    - check for inodes claiming duplicate blocks...
    - agno = 0
    - agno = 1
    - agno = 2
    - agno = 3
    - agno = 4
    - agno = 5
No modify flag set, skipping phase 5
Phase 6 - check inode connectivity...
    - traversing filesystem ...
    - traversal finished ...
    - moving disconnected inodes to lost+found ...
Phase 7 - verify link counts...
No modify flag set, skipping filesystem flush and exiting.
Size of logical volume vg_db/lv_db changed from 299.99 GiB (76798 extents) to 399.99 GiB (102397 extents).
Logical volume vg_db/lv_db successfully resized.
meta-data=/dev/mapper/vg_db-lv_db isize=256    agcount=6, agsize=13106944 blks
        =                               sectsz=512    attr=2, projid32bit=1
        =                               crc=0        finobt=0 spinodes=0
data     =                               bsize=4096   blocks=78641152, imaxpct=25
        =                               sunit=0      swidth=0 blks
naming   =version 2                     bsize=4096   ascii-ci=0 ftype=0
log      =internal                      bsize=4096   blocks=25599, version=2
        =                               sectsz=512   sunit=0 blks, lazy-count=1
realtime =none                          extsz=4096   blocks=0, rtextents=0
data blocks changed from 78641152 to 104854528
```

Перезагружаем хост:

```
shutdown -r now
```

Подключаемся по ssh, проверяем:

```
[root@localhost ~]# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/mapper/ol-root 38G  3.7G   34G  10% /
devtmpfs        988M    0   988M   0% /dev
tmpfs           998M    0   998M   0% /dev/shm
tmpfs           998M  41M   958M   5% /run
tmpfs           998M    0   998M   0% /sys/fs/cgroup
/dev/sda1       497M  237M  261M  48% /boot
/dev/mapper/vg_db-lv_db 400G  358M  400G   1% /db
tmpfs          200M    0   200M   0% /run/user/0
```

Видим, место на разделе /db увеличилось до 400Gb как и должно быть. Теперь надо включить автозагрузку mysqld и запустить демон:

```
systemctl enable mysqld
systemctl start mysqld
```

Статус запущенной mysqld выглядит примерно так:

```
[root@localhost ~]# systemctl status mysqld
• mysqld.service - MySQL Server
   Loaded: loaded (/usr/lib/systemd/system/mysqld.service; enabled; vendor preset: disabled)
   Active: active (running) since Mon 2017-11-27 17:17:43 MSK; 13s ago
     Docs: man:mysqld(8)
           http://dev.mysql.com/doc/refman/en/using-systemd.html
   Process: 1149 ExecStart=/usr/sbin/mysqld --daemonize --pid-file=/var/run/mysqld/mysqld.pid $MY
   YSQLD_OPTS (code=exited, status=0/SUCCESS)
   Process: 1102 ExecStartPre=/usr/bin/mysqld_pre_systemd (code=exited, status=0/SUCCESS)
   Main PID: 1151 (mysqld)
     CGroup: /system.slice/mysqld.service
             └─1151 /usr/sbin/mysqld --daemonize --pid-file=/var/run/mysqld/mysqld.pid

Nov 27 17:17:34 localhost.localdomain mysqld[1149]: Executing: /usr/sbin/mysqld --daemonize --p
id-file=/var/run/mysqld/mysqld.pid
[root@localhost ~]#
```

Готово!