

Linux Servers

Classroom Assignment 4 – Disk management

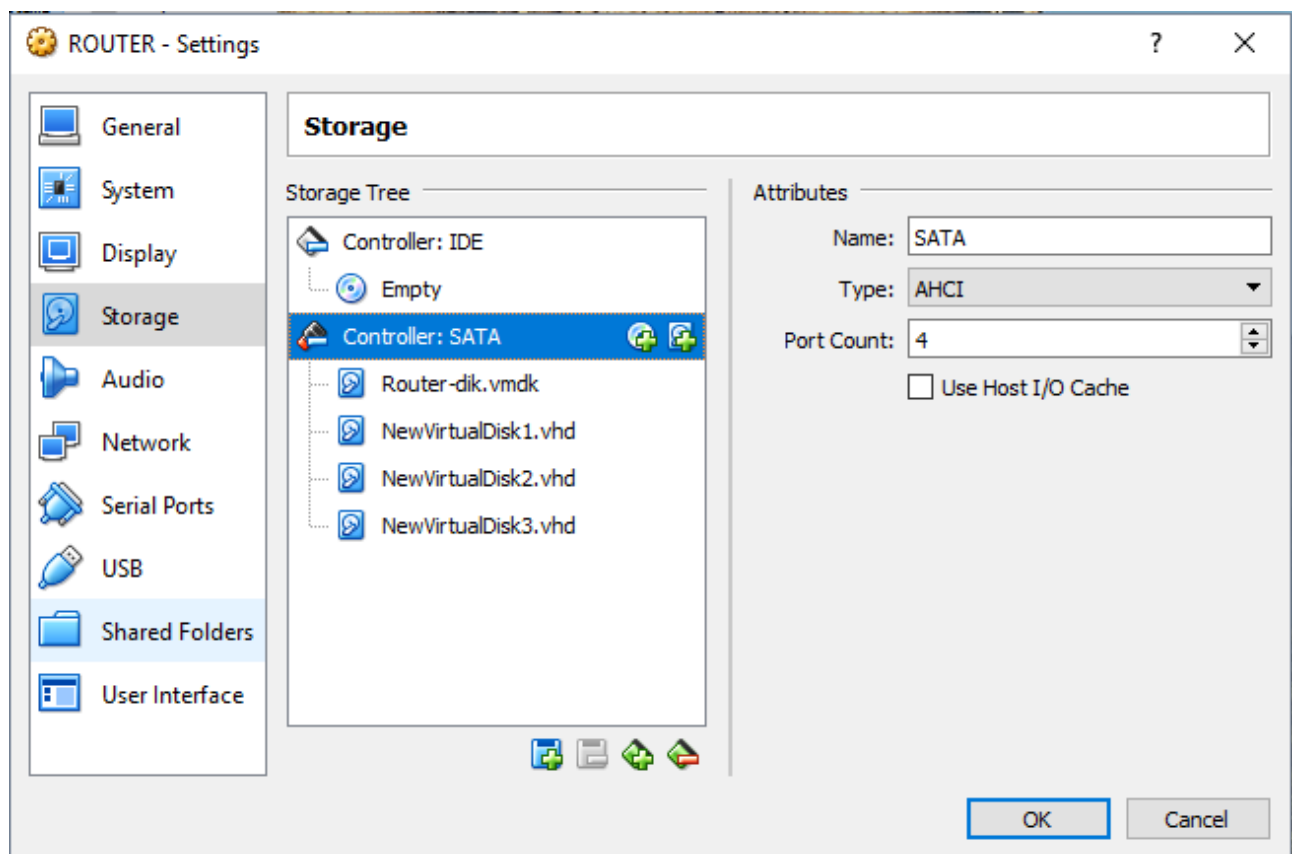
Mikael Romanov

Document your commands or take screenshots. Answer questions in english or finnish.

1. More disks

Add three new 5G disks to your VirtualBox VM. The VM needs to be powered off for this. Boot the VM and check with `fdisk` that you have three new disks as `sdb`, `sdc` and `sdd`. Create one maximum sized partition to all disks with `fdisk`.

Here are the virtualdisks added to Router:



Every disk has been formatted and are full capacity as seen below with `fdisk -l`

```

Device Boot      Start         End      Blocks   Id  System
/dev/sdb1        2048     10485759      5241856   83   Linux

Disk /dev/sdc: 5368 MB, 5368709120 bytes, 10485760 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: 0x496058ac

Device Boot      Start         End      Blocks   Id  System
/dev/sdc1        2048     10485759      5241856   83   Linux

Disk /dev/sdd: 5368 MB, 5368709120 bytes, 10485760 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: 0xc3138324

Device Boot      Start         End      Blocks   Id  System
/dev/sdd1        2048     10485759      5241856   83   Linux

Disk /dev/mapper/centos-root: 14.9 GB, 14889779200 bytes, 29081600 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

```

2. XFS

Create a new XFS filesystem to sdb1. Create a directory called *storage* to */media*. Mount your new XFS partition to this directory permanently using *fstab*.

Here I made an directory, then made xfs and mounted it to */media/storage/*

```

[root@localhost ~]# cd /media
[root@localhost media]# mkdir storage
[root@localhost media]# mkfs.xfs /dev/sdb1
meta-data=/dev/sdb1            isize=256    agcount=4, agsize=327616 blks
=                               sectsz=512   attr=2, projid32bit=1
=                               crc=0      finobt=0
data      =                    bsize=4096   blocks=1310464, imaxpct=25
=                               sunit=0      swidth=0 blks
naming    =version 2          bsize=4096   ascii-ci=0 ftype=0
log       =internal log      bsize=4096   blocks=2560, version=2
=                               sectsz=512   sunit=0 blks, lazy-count=1
realtime  =none              extsz=4096   blocks=0, rtextents=0
[root@localhost media]# mount -t xfs /dev/sdb1 /media/storage
[root@localhost media]# df -Th /media/storage
Filesystem      Type  Size  Used Avail Use% Mounted on
/dev/sdb1       xfs   5.0G  33M  5.0G   1% /media/storage

```

I made a change to /etc/fstab

```
GNU nano 2.3.1 File: /etc/fstab

#
# /etc/fstab
# Created by anaconda on Mon Oct  5 20:23:58 2015
#
# Accessible filesystems, by reference, are maintained under '/dev/disk'
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info
#
/dev/mapper/centos-root / xfs defaults 0 0
UUID=c453c708-5ec3-49f6-b642-e51bbda89ae4 /boot xfs defaults 0 0
/dev/mapper/centos-swap swap swap defaults 0 0
/dev/sdb1 /media/storage xfs defaults 0 0
```

3. LVM

Create a new LVM PV to sdc1. Create a new LVM volume group called *backups* using that PV and create a new logical volume (LV) on it with the name *everything*. What is the correct path to point to this LV now (where in /dev can it be found?)

Create a new xfs filesystem to that LV and mount it to a new directory */media/everything*.

First I made a directory, then created physical volume with `pvcreate`, after that a volume group named *backups* with `vgcreate`, then a logical volume named *everything* to `/dev/backups` with `lvcreate`. The volume size was 4G, because 5G was not possible. After the volumes were created, I made xfs to `/dev/backups/everything`. Then mounted it to `/media/everything`

```
[root@localhost media]# mkdir everything
[root@localhost media]# pvcreate /dev/sdc1
Physical volume "/dev/sdc1" successfully created
[root@localhost media]# vgcreate backups /dev/sdc1
Volume group "backups" successfully created
[root@localhost media]# lvcreate -L 4G -n everything backups
Logical volume "everything" created.
[root@localhost media]# mkfs.xfs /dev/backups/everything
meta-data=/dev/backups/everything isize=256    agcount=4, agsize=262144 blks
          =                               sectsz=512   attr=2, projid32bit=1
          =                               crc=0        finobt=0
data      =                               bsize=4096   blocks=1048576, imaxpct=25
          =                               sunit=0      swidth=0 blks
naming    =version 2                       bsize=4096   ascii-ci=0 ftype=0
log       =internal log                   bsize=4096   blocks=2560, version=2
          =                               sectsz=512   sunit=0 blks, lazy-count=1
realtime  =none                           extsz=4096   blocks=0, rtextents=0
[root@localhost media]# mount -t xfs /dev/backups/everything /media/everything
```

4. Extending LVM

Use the final disk (sdd1) to extend your volume group and LV. First create a new PV on sdd1, then extend the volume group and logical volume, and finally resize the XFS filesystem. Check that you have more free space with `du -sh`.

Created physical volume and extended the backups volume group

```
[root@localhost media]# pvcreate /dev/sdd1
Physical volume "/dev/sdd1" successfully created
[root@localhost media]# vgextend /dev/backups /dev/sdd1
Volume group "backups" successfully extended
```

The extended the everything logical volume with 5G

```
[root@localhost media]# lvextend -L +5G /dev/backups/everything
Size of logical volume backups/everything changed from 4.00 GiB (1024 extents) to 9.00 GiB (2304 extents).
Logical volume everything successfully resized.
```

/dev/backups/everything xfs has been extended to maximum

```
[root@localhost media]# xfs_growfs /dev/backups/everything
meta-data=/dev/mapper/backups-everything isize=256    agcount=4, agsize=262144 blks
        =                       sectsz=512    attr=2, projid32bit=1
        =                       crc=0          finobt=0
data      =                       bsize=4096    blocks=1048576, imaxpct=25
        =                       sunit=0        swidth=0 blks
naming    =version 2              bsize=4096    ascii-ci=0 ftype=0
log       =internal              bsize=4096    blocks=2560, version=2
        =                       sectsz=512    sunit=0 blks, lazy-count=1
realtime  =none                  extsz=4096     blocks=0, rtextents=0
data blocks changed from 1048576 to 2359296
```

As we can see here the /dev/backups/everything size is 9G, which was 4G earlier.

```
[root@localhost media]# df -h
Filesystem                Size      Used Avail Use% Mounted on
/dev/mapper/centos-root    14G       1.3G   13G   10% /
devtmpfs                  487M          0  487M    0% /dev
tmpfs                     497M          0  497M    0% /dev/shm
tmpfs                     497M       6.6M  490M    2% /run
tmpfs                     497M          0  497M    0% /sys/fs/cgroup
/dev/sda1                 497M      164M   334M   33% /boot
tmpfs                    100M          0  100M    0% /run/user/0
/dev/sdb1                 5.0G       33M   5.0G    1% /media/storage
/dev/mapper/backups-everything 9.0G       33M   9.0G    1% /media/everything
```

```
[root@localhost media]# df -Th /media/everything
Filesystem                Type      Size      Used Avail Use% Mounted on
/dev/mapper/backups-everything xfs       9.0G       33M   9.0G    1% /media/everything
```