

# Objective

You will explore some features of ZFS.

ZFS is a combined file system and logical volume manager that can use the best of ability of available disks.

# Do not start VMs now.

1. **Adding disks to the server VM**

Select the *server* VM in the VirtualBox manager main window and click at the **Storage** section from the **Details** pane;

Under the **Storage Devices**, click at Controller: SATA. Then click at the **Add Hard Disk** button (the green plus sign) and select Create new disk. Choose the followings:

Hard disk file type: VDI (default)

Storage on physical hard disk: Dynamically allocated (default) File location and size: poolDisk1 and 1.0 GB

Repeat the above to create another one disk called: poolDiskDisk2 of the size 1.0 GB

Repeat the above to create another two disks called: poolDisk3 and poolDisk4 of the size 2.0 GB.

1. **Install ZFS**

Now start the *server* VM;

The main components of ZFS are maintained as a standard Ubuntu package.

Install the ZFS

sudo apt install zfsutils-linux

1. **Create a ZFS pool**

Identify the newly created disks:这里就很奇怪 一开始盘居然没安上

lsblk -o NAME,SIZE,FSTYPE,TYPE,MOUNTPOINT

You will four new disks, likely /dev/sde, /dev/sdf, /dev/sdg and /dev/sdh, if you have successfully created three disks in last lab. These will be the raw disks we use to build ZFS on.

Create a mirrored pool (RAID0)

A zpool is a pool of storage made from a collection of VDEVs. One or more ZFS file systems (datassets) can be created from a ZFS pool. There are many ways to arrange disks to create a VDEV. In this exercise, we will create a VDEV of 2 disks in a mirror.

zpool是由VDEV集合组成的存储池。可以从ZFS池创建一个或多个ZFS文件系统（数据集）。有许多方法可以安排磁盘来创建VDEV。在本练习中，我们将在镜像中创建一

个包含2个磁盘的VDEV。

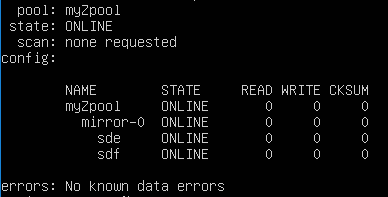
Create a pool from a mirrored VDEV made of the two 1GB disks:

sudo zpool create myZpool mirror /dev/sde /dev/sdf

where myZpool is the name of the pool.

You can use the zpool command to check the status of ZFS pool:

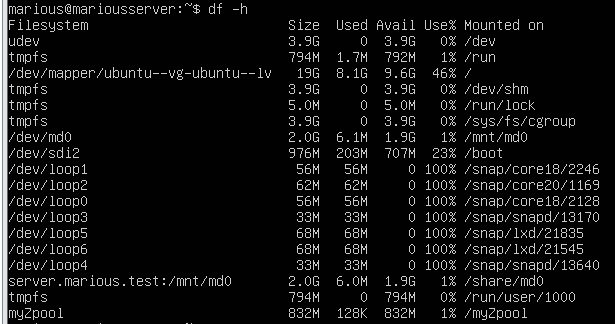
zpool status



You will see the newly created pool myZpool. The mirror-0 is the VDEV, a virtual device that representing the two disks.

You can check the new space by typing:

df -h



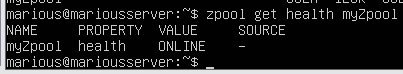
You will see:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Filesystem | Size Used | Avail | Use% | Mounted on |
| myZpool | 880M 0 | 880M | 0% | /myZpool |

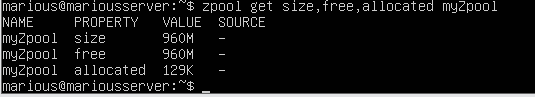
Looking at ZFS pool properties

Like other file systems, ZFS has many properties such as size, free space, capacity and mount options etc.

zpool get health myZpool



You will see that the the **health** of pool myZpool is ONLINE. zpool get size,free,allocated myZpool



You will see the total size, free and used space of the pool.

1. **Create datasets (file systems)**

Files are stored in datasets in a ZFS pool. A dataset will continue to place files in the pool until the pool is filled. You can put quotas on detasets to limiting dataset size that is not covered in this lab.

Create some datasets:

sudo zfs create myZpool/test1 sudo zfs create myZpool/test2 sudo zfs create myZpool/test3

Show the datasets:

zfs list



Each dataset is automatically mounted to its mount point and has full access to the storage pool. Store some data in one of the datasets:

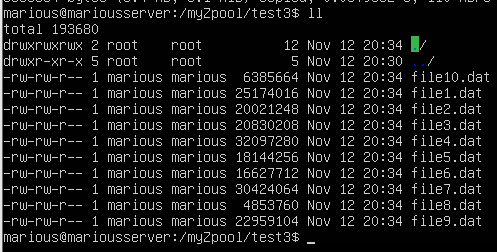
sudo chmod 777 /myZpool/test\*

The following command will 10 data files filled with random data:

cd /myZpool/test3

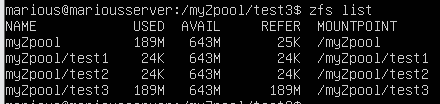
for i in {1..10}; do dd if=/dev/urandom of=file$i.dat bs=1024 count=$RANDOM; done

You may use the ll command to show the created files.



Now, show the datasets again:

zfs list



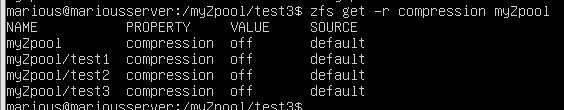
You will see that the myZpool/tests dataset occupies about 165MB of the pool and the unused space, about 715MB, is available to all datasets. (The actual values may differ on your VM as they are created with random data of random lengths.)

You do not need to worry how much space to preallocate for each datasets as you would have to do with traditional file systems. ZFS manages the entire pool of disks.

1. **Enable compression on datasets**

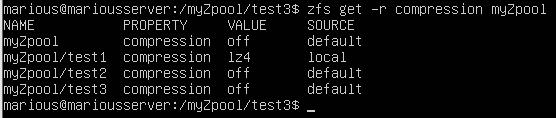
Look at the compression property:

zfs get -r compression myZpool



By default the compression property of all datasets is off. Enable compression:

sudo zfs set compression=lz4 myZpool/test1 zfs get -r compression myZpool



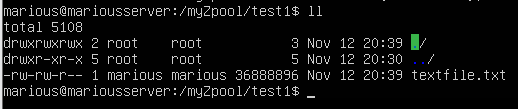
Now, the LZ4 compression on the dataset myZpool/test1 is enabled. Test compression

Create a text file with redundancy on compressed and uncompressed datasets:

cd /myZpool/test1

for i in {1..1000000}; do echo "Line $i: This is a line of text." >> textfile.txt; done

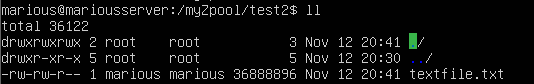
ll



cd /myZpool/test2

for i in {1..1000000}; do echo "Line $i: This is a line of text." >> textfile.txt; done

ll



zfs list



These two files created on two different datasets have the same content and size while ocuppy different space on the ZFS pool. The uncompressed copy on /myZpool/test2 uses 35.3MB, which is the same as shown as the size file while the compressed copy on /myZpool/test1 uses only 5.01MB.

You can check the compression ratio:

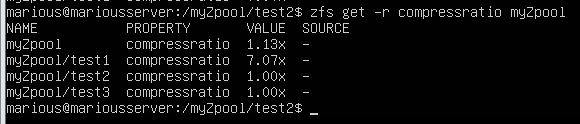
zfs get compressratio myZpool/test1



It shows a compression ratio of 7.07x.

You can check the compression ratio of the whole pool:

zfs get -r compressratio myZpool



The other uncompressed datasets have a compression ratio of 1.00x, which means no compression. The pool achieves 1.14x compression with current data on it. Some data will be compressed well to reach a high compression ratios while others will not be easily compressed.

1. **Snapshot a dataset**

A ZFS snapshot is a read-only copy of a dataset or the whole pool. It saves the state of the ZFS file system at a point of time that can be rolled back at a later time. Files can be extracted from a snapshot without performing a complete roll back.

Create a text file before snapshot:

cd /myZpool/test1

echo `date` >> myfile.txt cat myfile.txt



The current time is saved in the myfile.txt. Take a snapshot of the dataset:

sudo zfs snapshot -r myZpool/test1@snapshot1

Show the snapshots:

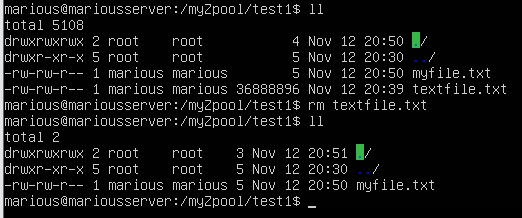
zfs list -t snapshot



Now lets "accidentally" delete a file: 假装不小心

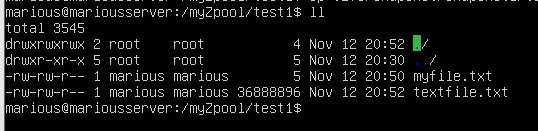
ll

rm textfile.txt ll



Extract the deleted file from the snapshot: 回来了

cp .zfs/snapshot/snapshot1/textfile.txt textfile.txt ll



Modify an existing file:

echo `date` >> myfile.txt cat myfile.txt



Now you have one more line in the myfile.txt to show the current time.

Extract the earlier copy of a file from the snapshot

For some reasons, you may need to get your earlier copy of a file back from the snapshot.

cp .zfs/snapshot/snapshot1/myfile.txt myfile.txt cat myfile.txt



Rollback the whole dataset

Now lets "accidentally" destroy all files:

rm /myZpool/test1/\* ll

Rollback the files from the snapshot:

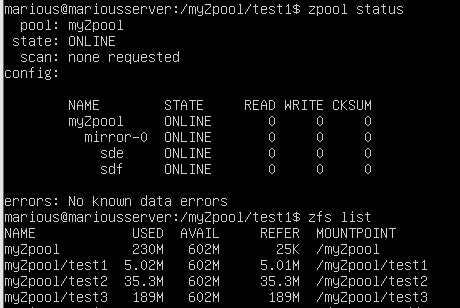
sudo zfs rollback myZpool/test1@snapshot1 ll



1. **Expand the capacity of the pool**

More disks can be added to the pool to expand the capacity. Show the current status:

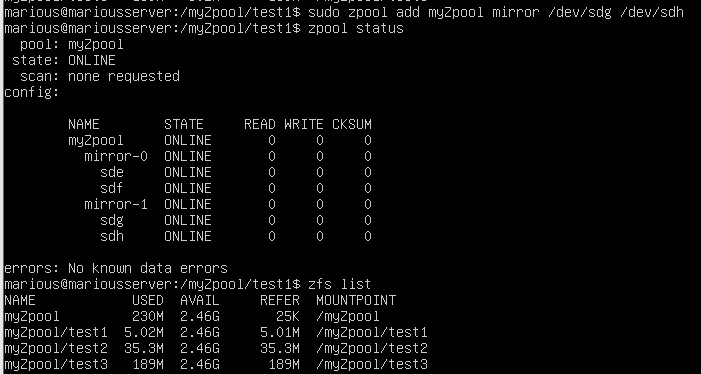
zpool status zfs list



Now add another VDEV of 2 disks in a mirror to the pool:

sudo zpool add myZpool mirror /dev/sdg /dev/sdh zpool status

zfs list



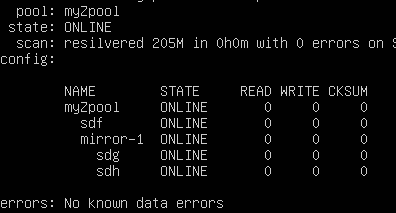
Now you see that another VDEV mirror-1 is added to the pool and the available capacity is increased for all datasets.

We have a 2x2 mirrored pool.

1. **Test reliability**

Simulate a disk failure by first removing it from the pool:

sudo zpool detach myZpool /dev/sde zpool status



You will see that the mirror-0 disappears with only one disk sdf in the pool. 555我的mirror0

cat myfile.txt

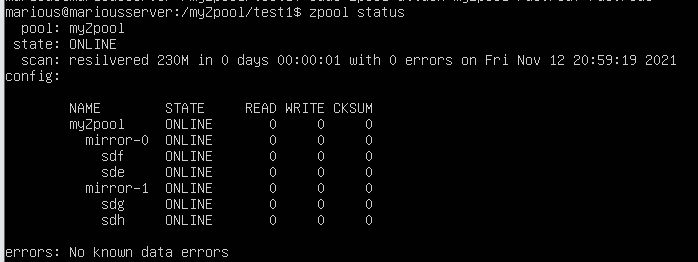
You can still access your file.

Add the disk back as a new disk to transform the existing disk sdf into a new mirror. First destroy the disk /dev/sde by overwriting it with zeros.

sudo dd if=/dev/zero of=/dev/sde bs=1M count=1024

Now add it back to the pool to form a mirrored VDEV with /dev/sdf. sudo zpool attach myZpool /dev/sdf /dev/sde

zpool status



Now you will see the two disks form a new mirrored VDEV, stilled called mirror-0.

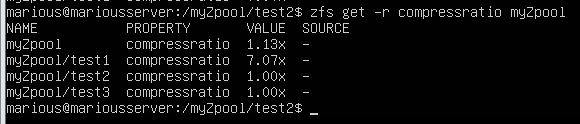
# Submission and mark

Show your work to the teacher. Of 6 marks, you can get

1.5 for showing datasets on the pool;



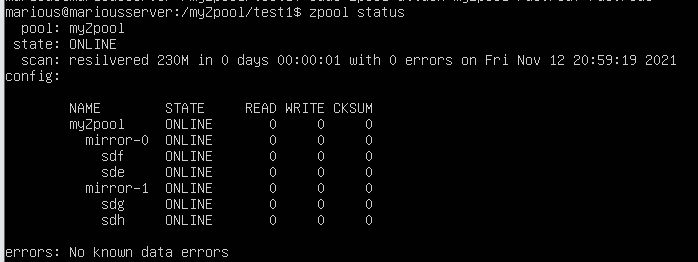
1.5 for showing the compression ratio of all datasets;



1.5 for showing the snapshot;



1.5 for showing the pool with two mirrored VDEVs.



You should be ready to answer any questions to demonstrate that all work is done by yourself otherwise you may receive 0 mark.

IMPORTANT NOTE: You will need to document all of your lab work in your wiki.