

1. В VirtualBox была создана виртуальная машина на Ubuntu 22.04 с 8-мю дополнительными дисками

```
sda                8:0    0    21G  0 disk
├─sda1             8:1    0     1M  0 part
├─sda2             8:2    0    513M  0 part /boot/efi
└─sda3             8:3    0   20,5G  0 part
   └─vgubuntu-root 252:0    0   18,4G  0 lvm  /var/snap/firefox/common/host-hunspell
      └─vgubuntu-swap_1 252:1    0    2,1G  0 lvm  [SWAP]
sdb                8:16   0     3G  0 disk
sdc                8:32   0     3G  0 disk
sdd                8:48   0     3G  0 disk
sde                8:64   0     3G  0 disk
sdf                8:80   0     3G  0 disk
sdg                8:96   0     3G  0 disk
sdh                8:112  0     3G  0 disk
sdi                8:128  0     3G  0 disk
```

2. Для начала создадим zpool и zfs в ней.

```
nastya@ubuntulab:~$ sudo zpool create test1 mirror /dev/sdb /dev/sdc
nastya@ubuntulab:~$ sudo zpool create test2 mirror /dev/sdd /dev/sde
nastya@ubuntulab:~$ sudo zpool create test3 mirror /dev/sdf /dev/sdg
nastya@ubuntulab:~$ sudo zpool create test4 mirror /dev/sdh /dev/sdi
nastya@ubuntulab:~$ zpool list
NAME      SIZE  ALLOC   FREE CKPOINT  EXPANDSZ   FRAG    CAP  DEDUP    HEALTH  A
LTROOT
test1    2.75G   108K  2.75G      -         -     0%    0%   1.00x   ONLINE  -
test2    2.75G   106K  2.75G      -         -     0%    0%   1.00x   ONLINE  -
test3    2.75G   108K  2.75G      -         -     0%    0%   1.00x   ONLINE  -
test4    2.75G   160K  2.75G      -         -     0%    0%   1.00x   ONLINE  -
```

```
nastya@ubuntulab:~$ sudo zfs create test1/fs1
```

```
nastya@ubuntulab:~$ sudo zfs create test2/fs2
nastya@ubuntulab:~$ sudo zfs create test3/fs3
nastya@ubuntulab:~$ sudo zfs create test4/fs4
nastya@ubuntulab:~$ zfs list
NAME          USED  AVAIL    REFER  MOUNTPOINT
test1         134K  2.62G    24K    /test1
test1/fs1      24K   2.62G    24K    /test1/fs1
test2         135K  2.62G    24K    /test2
test2/fs2      24K   2.62G    24K    /test2/fs2
test3         132K  2.62G    24K    /test3
test3/fs3      24K   2.62G    24K    /test3/fs3
test4         134K  2.62G    24K    /test4
test4/fs4      24K   2.62G    24K    /test4/fs4
```

1. Начнем первую часть домашнего задания. Зададим тип сжатия для каждого пула (можно было бы оставить и так, файловая система наследовала бы тип компрессии из пула, но для практики типы сжатия были заданы и для fs, что поможет выявить одну интересную особенность позже)

```
nastya@ubuntu1ab:~$ sudo zfs set compression=lzjb test1
nastya@ubuntu1ab:~$ sudo zfs set compression=lz4 test2
nastya@ubuntu1ab:~$ sudo zfs set compression=gzip-9 test3
nastya@ubuntu1ab:~$ sudo zfs set compression=zle test4
nastya@ubuntu1ab:~$ zfs get all | grep compression
test1      compression      lzjb            local
test1/fs1  compression      lzjb            inherited from test1
test2      compression      lz4            local
test2/fs2  compression      lz4            inherited from test2
test3      compression      gzip-9         local
test3/fs3  compression      gzip-9         inherited from test3
test4      compression      zle            local
test4/fs4  compression      zle            inherited from test4
```

```
nastya@ubuntu1ab:~$ sudo zfs set compression=lzjb test1/fs1
nastya@ubuntu1ab:~$ sudo zfs set compression=lz4 test2/fs2
nastya@ubuntu1ab:~$ sudo zfs set compression=gzip-9 test3/fs3
nastya@ubuntu1ab:~$ sudo zfs set compression=zle test4/fs4
nastya@ubuntu1ab:~$ zfs get all | grep compression
test1      compression      lzjb            local
test1/fs1  compression      lzjb            local
test2      compression      lz4            local
test2/fs2  compression      lz4            local
test3      compression      gzip-9         local
test3/fs3  compression      gzip-9         local
test4      compression      zle            local
test4/fs4  compression      zle            local
```

Далее создадим файл относительно большого размера используя команду `yes` и проверим его размер.

```
nastya@ubuntu1ab:~$ yes > file
```

```
nastya@ubuntu1ab:~$ ls -lh | grep file
-rw-rw-r-- 1 nastya nastya 2,2G фев 10 17:10 file
```

Скопируем данный файл в каждый каталог с файловой системой ZFS:

```
nastya@ubuntu1ab:~$ for i in {1..4}; do sudo cp file /test$i/fs$i/file; done
```

Переходим к проверке размера скопированного файла в соответствующем каталоге. После проверки размера файла стало понятно, что в каталоге test3/fs3 он был сжат лучше всего. Таким образом, приходим к выводу, что тип компрессии gzip с уровнем 9 является самым эффективным способом сжатия. За ним следует lz4, затем lzjb, меньше всего сжимает файл тип zle. Интересно отметить, что проверка размера файла путем ls -lh и zfs list дают примерно одинаковые результаты для gzip-9, lz4 и lzjb, однако для zle результат ls -lh выглядит так, как будто файл не был скомпрессован, хотя вывод zfs list показывает обратное.

```
nastya@ubuntu1ab:~$ ls -lh /test1/fs1
total 78M
-rw-r--r-- 1 root root 2,2G фев 10 17:11 file
nastya@ubuntu1ab:~$ ls -lh /test2/fs2
total 18M
-rw-r--r-- 1 root root 2,2G фев 10 17:11 file
nastya@ubuntu1ab:~$ ls -lh /test3/fs3
total 8,8M
-rw-r--r-- 1 root root 2,2G фев 10 17:12 file
nastya@ubuntu1ab:~$ ls -lh /test4/fs4
total 2,2G
-rw-r--r-- 1 root root 2,2G фев 10 17:12 file
```

```
nastya@ubuntu1ab:~$ zfs list
NAME                USED    AVAIL    REFER    MOUNTPOINT
test1               77.8M   2.55G    24K      /test1
test1/fs1           77.6M   2.55G    77.6M    /test1/fs1
test2               17.6M   2.61G    24K      /test2
test2/fs2           17.4M   2.61G    17.4M    /test2/fs2
test3                8.97M   2.62G    24K      /test3
test3/fs3            8.82M   2.62G    8.82M    /test3/fs3
test4                2.15G   485M     24K      /test4
test4/fs4            2.15G   485M    2.15G    /test4/fs4
```

Отметим еще одну интересную особенность. Вывод zfs get all показывает, что есть небольшие расхождения в уровне сжатия между пулом и созданной файловой системой. Файловая система имеет немного больший процент компрессии.

```
nastya@ubuntu1ab:~$ zfs get all | grep compressratio | grep -v ref
test1      compressratio 28.38x -
test1/fs1  compressratio 28.40x -
test2      compressratio 126.81x -
test2/fs2  compressratio 127.21x -
test3      compressratio 251.31x -
test3/fs3  compressratio 252.86x -
test4      compressratio 1.00x  -
test4/fs4  compressratio 1.00x  -
```

2. Перейдем ко второй части домашнего задания — определению настроек пула.

Скачиваем пул по ссылке, указанной в методичке.

```
nastya@ubuntu1ab:~$ wget -O archive.tar.gz 'https://drive.usercontent.google.com/download?id=1MvrcEp-WgAQe57aDEzxSRa1PAwbNN1Bb&export=download'
```

Разархивируем содержимое полученного на предыдущем шаге файла.

```
nastya@ubuntu1ab:~$ tar -xzvf archive.tar.gz
zpoolexport/
zpoolexport/filea
zpoolexport/fileb
```

Проверяем, можно ли данный пул импортировать на свою VM.

```
nastya@ubuntu1ab:~$ sudo zpool import -d zpoolexport/
pool: otus
id: 6554193320433390805
state: ONLINE
status: Some supported features are not enabled on the pool.
       (Note that they may be intentionally disabled if the
       'compatibility' property is set.)
action: The pool can be imported using its name or numeric identifier, though
       some features will not be available without an explicit 'zpool upgrade'.
config:

        otus                                     ONLINE
        mirror-0                                 ONLINE
          /home/nastya/zpoolexport/filea         ONLINE
          /home/nastya/zpoolexport/fileb         ONLINE
```

Импортируем пул и проверяем, что все получилось

```
nastya@ubuntu1ab:~$ sudo zpool import -d zpoolexport/ otus
```

```
nastya@ubuntu1ab:~$ zpool list
NAME      SIZE  ALLOC   FREE CKPOINT  EXPANDSZ   FRAG    CAP  DEDUP   HEALTH  ALTROOT
otus      480M  2.18M   478M    -         -         0%     0%   1.00x  ONLINE  -
test1     2.75G  77.8M   2.67G    -         -         0%     2%   1.00x  ONLINE  -
test2     2.75G  17.6M   2.73G    -         -         0%     0%   1.00x  ONLINE  -
test3     2.75G  8.97M   2.74G    -         -         0%     0%   1.00x  ONLINE  -
test4     2.75G  2.15G   613M    -         -         3%    78%   1.00x  ONLINE  -
```

Выполняем команду `zpool get all otus`, далее выбираем отдельный параметр пула для вывода, например `zpool get size otus`

```
nastya@ubuntu1ab:~$ zpool get size otus
NAME      PROPERTY  VALUE   SOURCE
otus      size      480M    -
```



### 3. Начинаем работу со снапшотами.

В первой части домашнего задания был создан файл, который был скопирован в /test1/fs1

```
nastya@ubuntu1ab:~$ ls /test1/fs1
file
nastya@ubuntu1ab:~$ ls -lh /test1/fs1
total 78M
-rw-r--r-- 1 root root 2,2G фев 10 17:11 file
```

Сделаем снапшот этой файловой системы, удалим файл и «откатимся» к снапшоту, чтобы вернуть его.

```
nastya@ubuntu1ab:~$ sudo zfs snapshot test1/fs1@snapshot
nastya@ubuntu1ab:~$ sudo rm /test1/fs1/file
nastya@ubuntu1ab:~$ ls /test1/fs1
nastya@ubuntu1ab:~$
```

```
nastya@ubuntu1ab:~$ sudo zfs rollback test1/fs1@snapshot
nastya@ubuntu1ab:~$ ls /test1/fs1
file
```

Переходим к части из методички:

Скачиваем файл:

```
nastya@ubuntu1ab:~$ sudo wget -O otus_task2.file 'https://drive.usercontent.google.com/download?id=1wgxjih8YZ-cqLqaZVa0LA3h3Y029c3oI&export=download'
--2024-02-14 14:47:27-- https://drive.usercontent.google.com/download?id=1wgxjih8YZ-cqLqaZVa0LA3h3Y029c3oI&export=download
```

Восстанавливаем файловую систему из снапшота:

```
nastya@ubuntu1ab:~$ sudo zfs receive test1/test@today < otus_task2.file
nastya@ubuntu1ab:~$ zfs list
```

NAME	USED	AVAIL	REFER	MOUNTPOINT
test1	80.3M	2.55G	24K	/test1
test1/fs1	77.6M	2.55G	77.6M	/test1/fs1
test1/test	2.46M	2.55G	2.44M	/test1/test
test2	17.6M	2.61G	24K	/test2
test2/fs2	17.4M	2.61G	17.4M	/test2/fs2
test3	9.90M	2.62G	24K	/test3
test3/fs3	8.82M	2.62G	8.82M	/test3/fs3
test4	2.15G	485M	24K	/test4
test4/fs4	2.15G	485M	2.15G	/test4/fs4

Находим содержимое файла с именем «secret\_message»

```
nastya@ubuntu1ab:~$ find /test1/test -name "secret_message"  
/test1/test/task1/file_mess/secret_message  
nastya@ubuntu1ab:~$ cat /test1/test/task1/file_mess/secret_message  
https://otus.ru/lessons/linux-h1/
```