Task 3 - Internal storage support

Yuan Fan <yfan@kth.se>
Xuwei Gong<xuweig@kth.se>

I. Google Cloud Storage FUSE

As we mentioned in Task 1 report, the google cloud storage support us with buckets to storage objects in the cloud, if we want to integrated the storage with our compute engine server vm from task 2 to do operations such as adding the directories, the cloud storage FUSE tool was introduced.

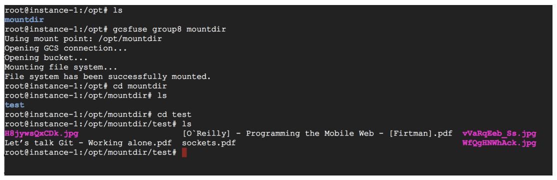
A. Introduction

Cloud Storage FUSE is an open source FUSE adapter that allows you to mount Google Cloud Storage buckets as file systems on Linux or OS X systems. Its working mechanism is by translating object storage names into a file and directory system, interpreting the "/" character in object names as a directory separator so that objects with the same common prefix are treated as files in the same directory [1].

To have GCS fuse working on the vm, the first thing is to download the gcsfuse package and setup the environment, the gcsfuse will use the credential that the service account we authenticated with.

B. Implementation

Following the instruction and document in the Google website, we mount our vm with the storage bucket that we created in Task 1. We took the following steps to first create a directory after we login to our vm, then use command *`sudo gcsfuse bucket-name /path/to/mount/point`* to mounting a storage bucket. Shown in following figures:



```
root@instance-1:/opt# ls
mountdir
root@instance-1:/opt# gcsfuse group8 mountdir
Using mount point: /opt/mountdir
Opening GCS connection...
Opening bucket...
Mounting file system...
File system has been successfully mounted.
root@instance-1:/opt# cd mountdir
root@instance-1:/opt/mountdir# ls
test
root@instance-1:/opt/mountdir# cd test
root@instance-1:/opt/mountdir/test# ls
H8jywsQxCDk.jpg                   [O`Reilly] - Programming the Mobile Web - [Firtman].pdf  vVaRqEeb_Ss.jpg
Let's talk Git - Working alone.pdf  sockets.pdf                                              WfQgHNWhAck.jpg
root@instance-1:/opt/mountdir/test# 
```

Fig. 1. Mounting steps

At this time, applications can interact with the mounted bucket like a simple file system, providing virtually limitless file storage running in the cloud [1].

II. Access Google Cloud Storage within vm

Once we mount the vm with the buckets which means we can upload/download files and analyze the different storage performance like we done locally. After copying the code to the vm, this time we choose to mount with another bucket named *external8*. We modified our code using *shutil*, which basically the unix command *"cp src dst"* to move files between the file system.

The whole test with single transfers we use 8 files in total 423.32mb same with task1, the experiment keeps record of each single file about the R/W information, then we calculate the average speed of these two operations. Table 1 summarize the findings:

TABLE I
External access benchmark

	movie.mp4	reference.pdf	picture.jpg	Total
Size	364.94mb	26.91mb	8.52mb	423.32mb
Write Speed	21.18mb/s	10.82mb/s	4.78mb/s	14.0mb/s(AVG)
Read speed	20.21mb/s	173.48mb/s	73.78mb/s	22.0mb/s(AVG)

Compared to task1, this time the transmission rate are much higher than uploading with blob, we moved the existing files to the mounting directory and move back.

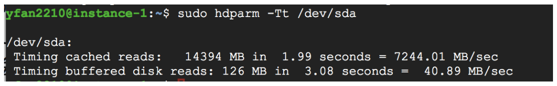
For the parallel transmission, we took 2 different hosts (Mac OS X and Win 10) under the same network conditions to redoing the test with same amount of file and same sizes, login to the same vm and simultaneously done the move operations, the result shows that the R/W average speed of host1 are 8.38mb/s and 10.35mb/s where the other one gives the speed of 10.47mb/s and 15.0mb/s. These data shows that the synchronized transmission will share the data flow bandwidth.

III. Benchmark access to local file system(SSD)

At this point, we searched for some useful tools to test out the hard disk performance and decided to use Linux command line tool: "`hdparm`" and "`dd`".

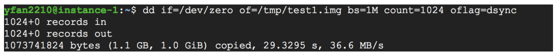
Command hdparm is used to get or set the parameters of the SATA/IDE device,while command dd (device to device) is used to detect the write performance of hard disk devices under Linux and Unix-like systems. In fact hdparm is a user-level program in linux. When checking on the source code, we found that this command sends some `ioctl` commands to the disk. However, note that since you can get and set it, be careful when using this command. We only need to check the hard disk information, so it will not cause any harm to the hard disk and the system. Following figures shown the related command and the information of vm disk.

In writing test, we write into the disk with 1M data with 1024 times the writing speed is 36.6mb/s, the operation was taken by synchronous I/O, setting this can remove the effects of caching in order to present with more accurate results.



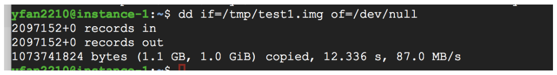
```
yfan2210@instance-1:~$ sudo hdparm -Tt /dev/sda

/dev/sda:
 Timing cached reads:   14394 MB in  1.99 seconds = 7244.01 MB/sec
 Timing buffered disk reads: 126 MB in  3.08 seconds =  40.89 MB/sec
```

Fig. 2. Reading Speed of cached and bufferd disk tested by hdparm



```
yfan2210@instance-1:~$ dd if=/dev/zero of=/tmp/test1.img bs=1M count=1024 oflag=dsync
1024+0 records in
1024+0 records out
1073741824 bytes (1.1 GB, 1.0 GiB) copied, 29.3295 s, 36.6 MB/s
```

Fig. 3. Writing Speed of Server throughput tested by dd



```
yfan2210@instance-1:~$ dd if=/tmp/test1.img of=/dev/null
2097152+0 records in
2097152+0 records out
1073741824 bytes (1.1 GB, 1.0 GiB) copied, 12.336 s, 87.0 MB/s
```

Fig. 4. Reading Speed of Server throughput tested by dd

Comparing these two methods in buffered disk, it can be seen that there is a difference between dd test read speed and hdparm, the measured speed of dd is related to the size of the read-write block, and may also be affected by the process of IO reading and writing in the system. Also, it's encouraged to R/W through the internal storage because of the performance.

References

[1] "Cloud storage fuse, google cloud platform document." [Online]. Available: https://cloud.google.com/storage/docs/gcs-fuseusing