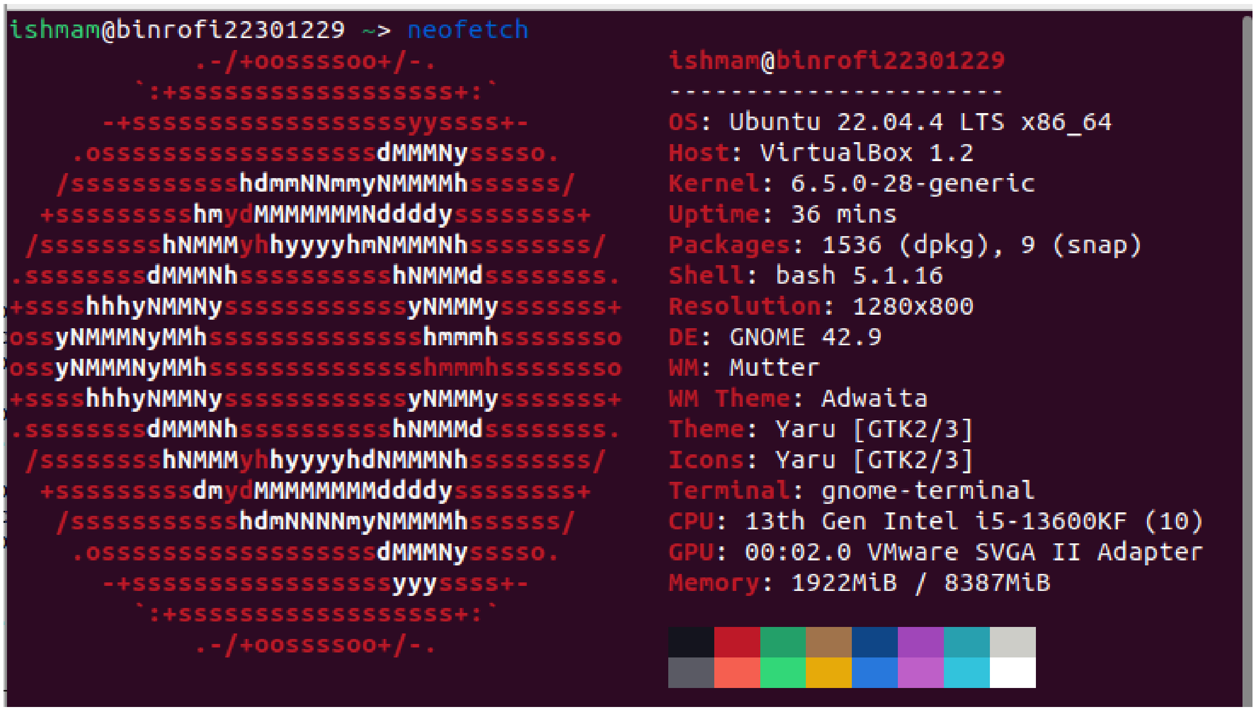
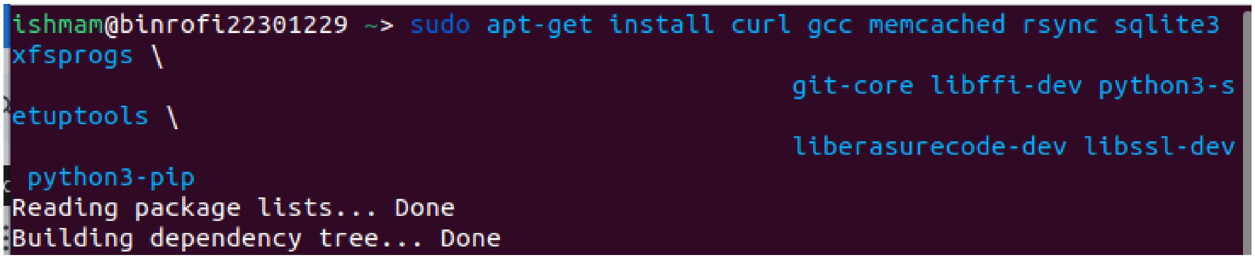
Step 1: Dependencies of the Swift.

We begin our journey with the very common thing that we do on a ubuntu system before proceeding, “sudo apt-get update” and “sudo apt-get upgrade” and install and update the dependencies for our system. Then we check our system and get to start our work.

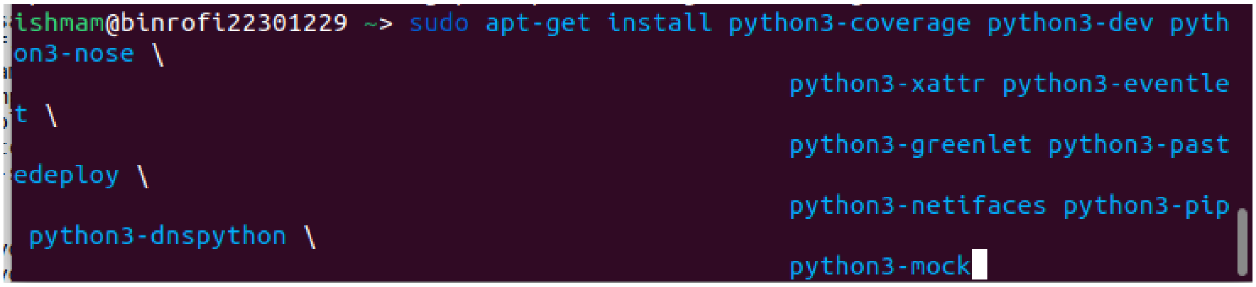


```
ishmam@binrofi22301229 ~> neofetch
            .-/+oossssoo+/-.               ishmam@binrofi22301229
        `:+ssssssssssssssssss+:`           ----------------------
      -+ssssssssssssssssssyyssss+-         OS: Ubuntu 22.04.4 LTS x86_64
    .ossssssssssssssssssdMMMNysssso.       Host: VirtualBox 1.2
   /ssssssssssshdmmNNmmyNMMMMhssssss/      Kernel: 6.5.0-28-generic
  +ssssssssshmydMMMMMMMNddddyssssssss+     Uptime: 36 mins
 /sssssssshNMMMyhhyyyyhmNMMMNhssssssss/    Packages: 1536 (dpkg), 9 (snap)
.ssssssssdMMMNhsssssssssshNMMMdssssssss.   Shell: bash 5.1.16
+sssshhhyNMMNyssssssssssssyNMMMysssssss+   Resolution: 1280x800
ossyNMMMNyMMhsssssssssssssshmmmhssssssso   DE: GNOME 42.9
ossyNMMMNyMMhsssssssssssssshmmmhssssssso   WM: Mutter
+sssshhhyNMMNyssssssssssssyNMMMysssssss+   WM Theme: Adwaita
.ssssssssdMMMNhsssssssssshNMMMdssssssss.   Theme: Yaru [GTK2/3]
 /sssssssshNMMMyhhyyyyhdNMMMNhssssssss/    Icons: Yaru [GTK2/3]
  +sssssssssdmydMMMMMMMMddddyssssssss+     Terminal: gnome-terminal
   /ssssssssssshdmNNNNmyNMMMMhssssss/      CPU: 13th Gen Intel i5-13600KF (10)
    .ossssssssssssssssssdMMMNysssso.       GPU: 00:02.0 VMware SVGA II Adapter
      -+sssssssssssssssssyyyssss+-         Memory: 1922MiB / 8387MiB
        `:+ssssssssssssssssss+:`
            .-/+oossssoo+/-.
```

After getting the stage ready we proceed to install the dependencies that we need for our swift installation.

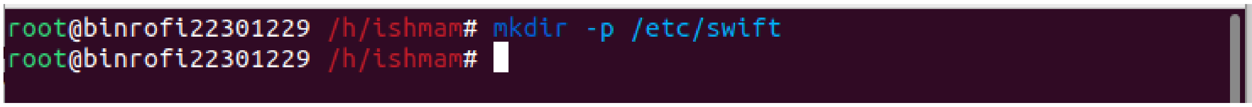


```
ishmam@binrofi22301229 ~> sudo apt-get install curl gcc memcached rsync sqlite3
xfsprogs \
                                                    git-core libffi-dev python3-s
etuptools \
                                                    liberasurecode-dev libssl-dev
 python3-pip
Reading package lists... Done
Building dependency tree... Done
```



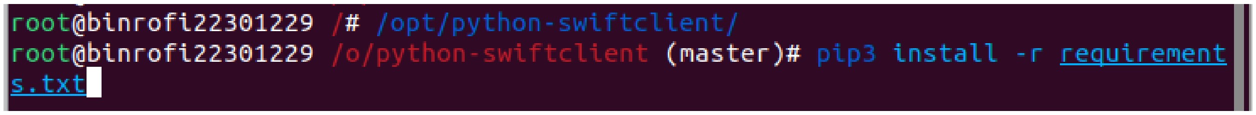
```
ishmam@binrofi22301229 ~> sudo apt-get install python3-coverage python3-dev pyth
on3-nose \
                                                    python3-xattr python3-eventle
t \
                                                    python3-greenlet python3-past
edeploy \
                                                    python3-netifaces python3-pip
 python3-dnspython \
                                                    python3-mock
```

Now we are going to make the directory for our swift.



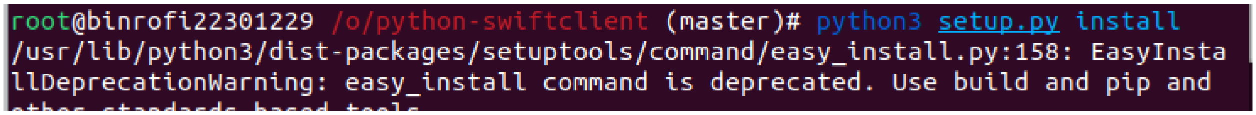
```
root@binrofi22301229 /h/ishmam# mkdir -p /etc/swift
root@binrofi22301229 /h/ishmam# 
```

After entering the cloned repo, we will install the requirements from the file.



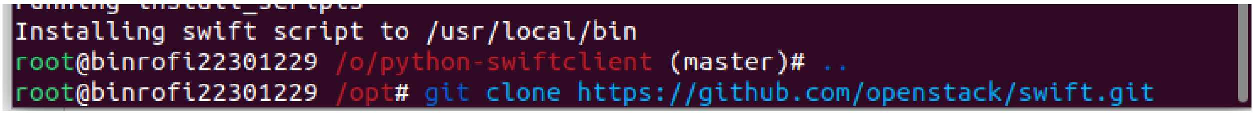
```
root@binrofi22301229 /# /opt/python-swiftclient/
root@binrofi22301229 /o/python-swiftclient (master)# pip3 install -r requirement
s.txt
```

Then we install the python3 setup.py file on our machine.



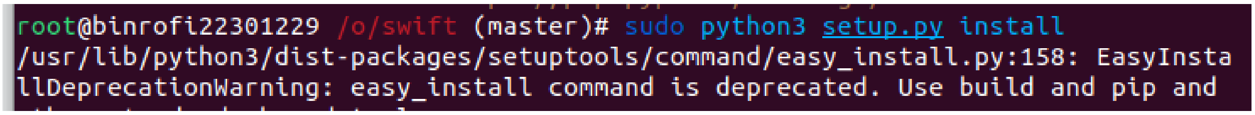
```
root@binrofi22301229 /o/python-swiftclient (master)# python3 setup.py install
/usr/lib/python3/dist-packages/setuptools/command/easy_install.py:158: EasyInsta
llDeprecationWarning: easy_install command is deprecated. Use build and pip and
other standards-based tools
```

Now we will clone the swift repository from github.



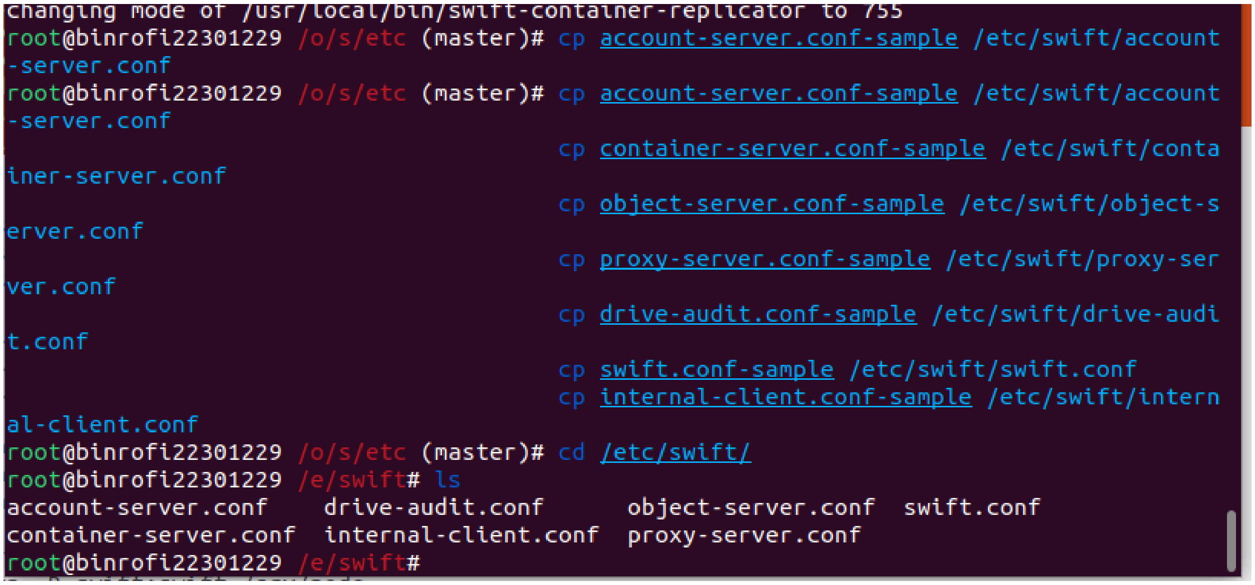
```
running install_scripts
Installing swift script to /usr/local/bin
root@binrofi22301229 /o/python-swiftclient (master)# ..
root@binrofi22301229 /opt# git clone https://github.com/openstack/swift.git
```

Now I will install the setup.py file in my swift directory



```
root@binrofi22301229 /o/swift (master)# sudo python3 setup.py install
/usr/lib/python3/dist-packages/setuptools/command/easy_install.py:158: EasyInsta
llDeprecationWarning: easy_install command is deprecated. Use build and pip and
```

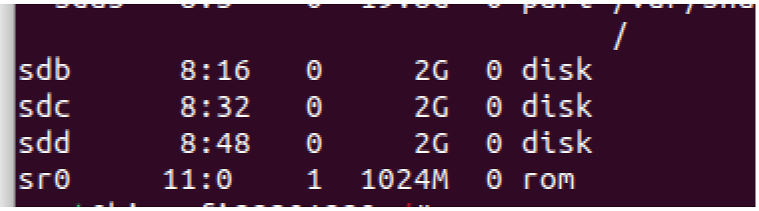
Now I will copy the files from /opt/swift/etc/ to /etc/swift/



```
changing mode of /usr/local/bin/swift-container-replicator to 755
root@binrofi22301229 /o/s/etc (master)# cp account-server.conf-sample /etc/swift/account
-server.conf
root@binrofi22301229 /o/s/etc (master)# cp account-server.conf-sample /etc/swift/account
-server.conf
                                        cp container-server.conf-sample /etc/swift/conta
iner-server.conf
                                        cp object-server.conf-sample /etc/swift/object-s
erver.conf
                                        cp proxy-server.conf-sample /etc/swift/proxy-ser
ver.conf
                                        cp drive-audit.conf-sample /etc/swift/drive-audi
t.conf
                                        cp swift.conf-sample /etc/swift/swift.conf
                                        cp internal-client.conf-sample /etc/swift/intern
al-client.conf
root@binrofi22301229 /o/s/etc (master)# cd /etc/swift/
root@binrofi22301229 /e/swift# ls
account-server.conf    drive-audit.conf      object-server.conf  swift.conf
container-server.conf  internal-client.conf  proxy-server.conf
root@binrofi22301229 /e/swift#
```

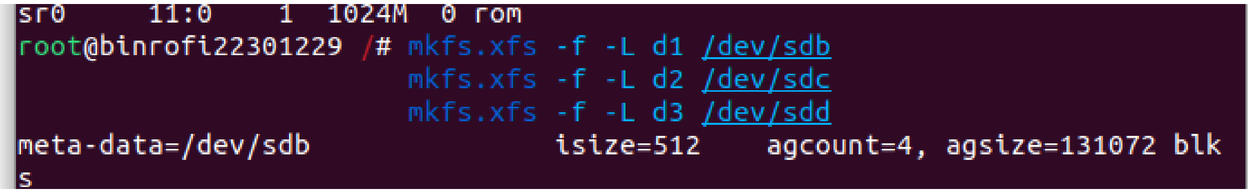
Step 2: Mounting virtual disks

Now lets add the virtual disks here, I have added 2gb each and 3 disks



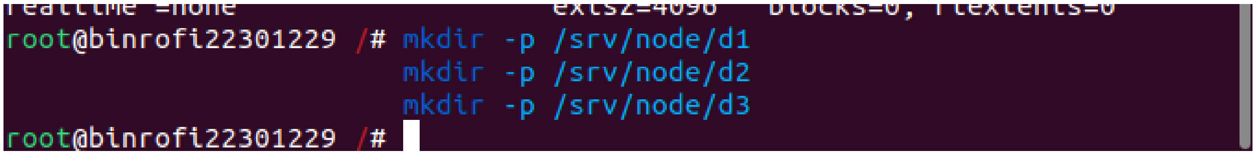
```
                                      /
sdb       8:16   0      2G  0 disk
sdc       8:32   0      2G  0 disk
sdd       8:48   0      2G  0 disk
sr0      11:0    1  1024M  0 rom
```

Now formatting the disks in xfs file system here,



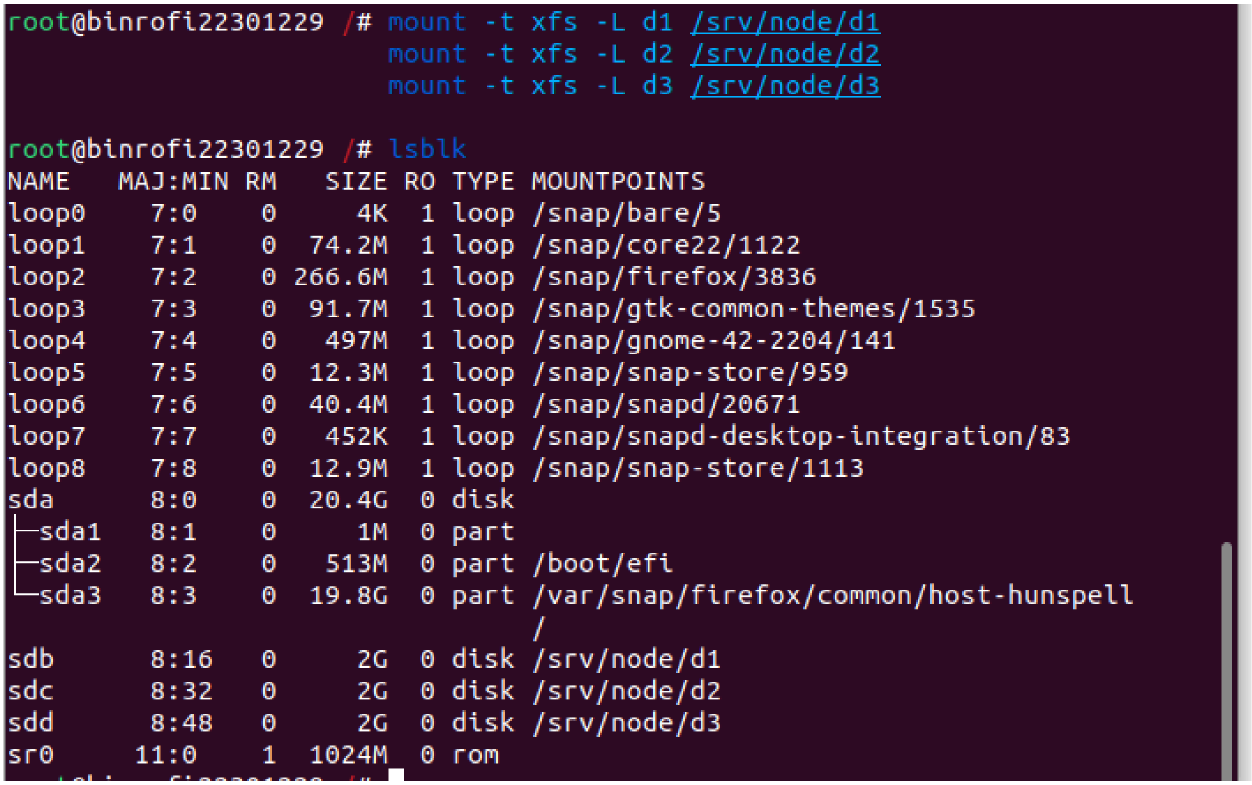
```
sr0      11:0    1  1024M  0 rom
root@binrofi22301229 /# mkfs.xfs -f -L d1 /dev/sdb
                        mkfs.xfs -f -L d2 /dev/sdc
                        mkfs.xfs -f -L d3 /dev/sdd
meta-data=/dev/sdb               isize=512    agcount=4, agsize=131072 blk
s
```

Now we are going to create three nodes for the files systems here



```
realtime =none                   extsz=4096   blocks=0, rtextents=0
root@binrofi22301229 /# mkdir -p /srv/node/d1
                        mkdir -p /srv/node/d2
                        mkdir -p /srv/node/d3
root@binrofi22301229 /# 
```

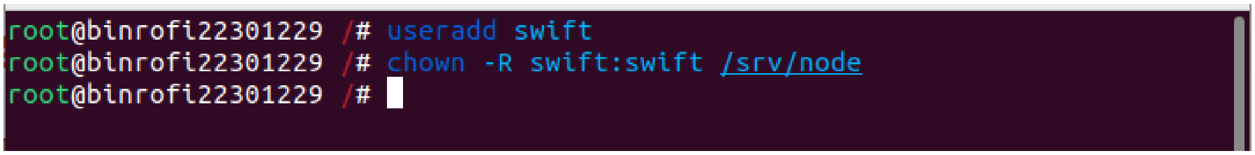
Now we are mounting the disks on our vm



```
root@binrofi22301229 /# mount -t xfs -L d1 /srv/node/d1
                        mount -t xfs -L d2 /srv/node/d2
                        mount -t xfs -L d3 /srv/node/d3

root@binrofi22301229 /# lsblk
NAME    MAJ:MIN RM   SIZE RO TYPE MOUNTPOINTS
loop0     7:0    0     4K  1 loop /snap/bare/5
loop1     7:1    0  74.2M  1 loop /snap/core22/1122
loop2     7:2    0 266.6M  1 loop /snap/firefox/3836
loop3     7:3    0  91.7M  1 loop /snap/gtk-common-themes/1535
loop4     7:4    0   497M  1 loop /snap/gnome-42-2204/141
loop5     7:5    0  12.3M  1 loop /snap/snap-store/959
loop6     7:6    0  40.4M  1 loop /snap/snapd/20671
loop7     7:7    0   452K  1 loop /snap/snapd-desktop-integration/83
loop8     7:8    0  12.9M  1 loop /snap/snap-store/1113
sda       8:0    0  20.4G  0 disk
├─sda1    8:1    0     1M  0 part
├─sda2    8:2    0   513M  0 part /boot/efi
└─sda3    8:3    0  19.8G  0 part /var/snap/firefox/common/host-hunspell
                                  /
sdb       8:16   0     2G  0 disk /srv/node/d1
sdc       8:32   0     2G  0 disk /srv/node/d2
sdd       8:48   0     2G  0 disk /srv/node/d3
sr0      11:0    1  1024M  0 rom
```

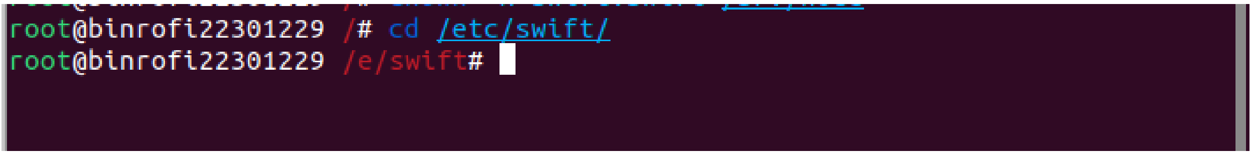
Step 3: Creating user ‘swift’

Now lets create a user swift and give it permissions to the nodes.



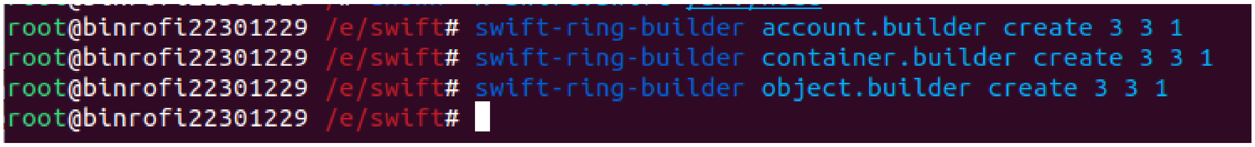
```
root@binrofi22301229 /# useradd swift
root@binrofi22301229 /# chown -R swift:swift /srv/node
root@binrofi22301229 /# 
```

Then we will again go to /etc/swift/



```
root@binrofi22301229 /# cd /etc/swift/
root@binrofi22301229 /e/swift# 
```

Now we will build the rings.



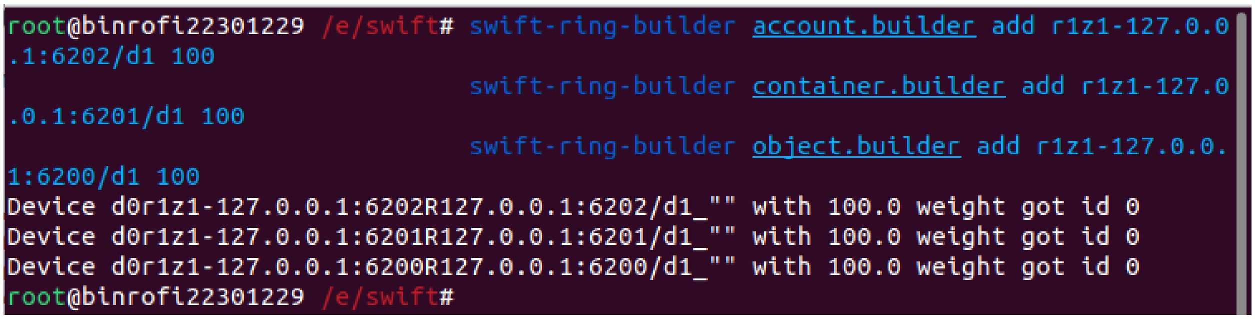
```
root@binrofi22301229 /e/swift# swift-ring-builder account.builder create 3 3 1
root@binrofi22301229 /e/swift# swift-ring-builder container.builder create 3 3 1
root@binrofi22301229 /e/swift# swift-ring-builder object.builder create 3 3 1
root@binrofi22301229 /e/swift# 
```

Now I will add the devices to the ring.

swift-ring-builder account.builder add r1z1-127.0.0.1:6202/d1 100

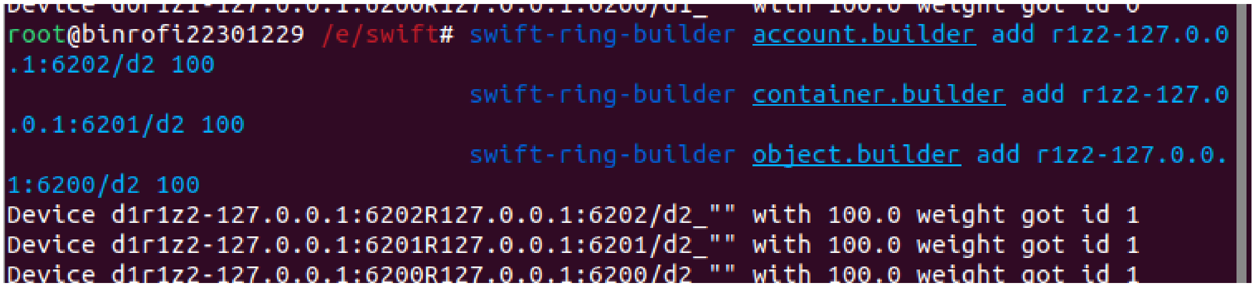
swift-ring-builder container.builder add r1z1-127.0.0.1:6201/d1 100

swift-ring-builder object.builder add r1z1-127.0.0.1:6200/d1 100



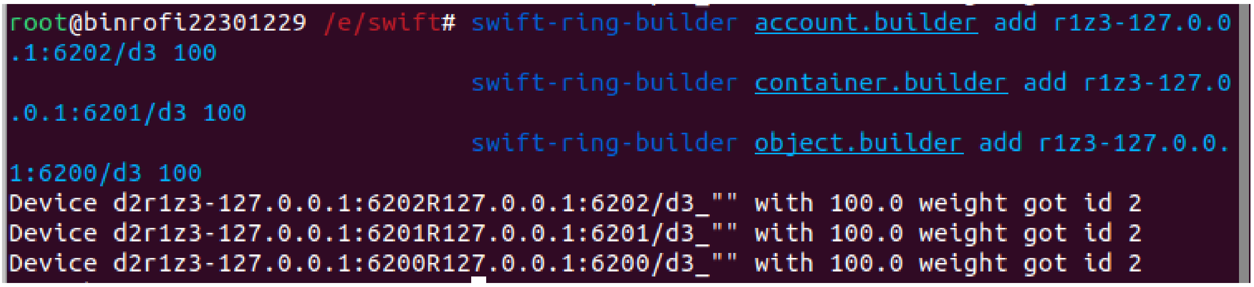
```
root@binrofi22301229 /e/swift# swift-ring-builder account.builder add r1z1-127.0.0
.1:6202/d1 100
                               swift-ring-builder container.builder add r1z1-127.0
.0.1:6201/d1 100
                               swift-ring-builder object.builder add r1z1-127.0.0.
1:6200/d1 100
Device d0r1z1-127.0.0.1:6202R127.0.0.1:6202/d1_"" with 100.0 weight got id 0
Device d0r1z1-127.0.0.1:6201R127.0.0.1:6201/d1_"" with 100.0 weight got id 0
Device d0r1z1-127.0.0.1:6200R127.0.0.1:6200/d1_"" with 100.0 weight got id 0
root@binrofi22301229 /e/swift#
```

swift-ring-builder account.builder add r1z2-127.0.0.1:6202/d2 100
swift-ring-builder container.builder add r1z2-127.0.0.1:6201/d2 100
swift-ring-builder object.builder add r1z2-127.0.0.1:6200/d2 100



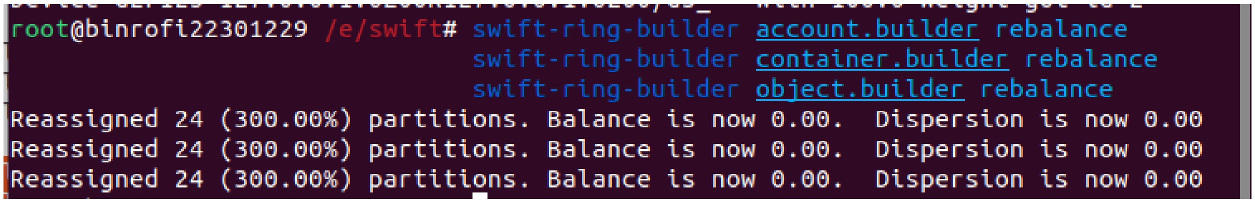
```
Device d0r1z1-127.0.0.1:6200R127.0.0.1:6200/d1_""  with 100.0 weight got id 0
root@binrofi22301229 /e/swift# swift-ring-builder account.builder add r1z2-127.0.0
.1:6202/d2 100
                               swift-ring-builder container.builder add r1z2-127.0
.0.1:6201/d2 100
                               swift-ring-builder object.builder add r1z2-127.0.0.
1:6200/d2 100
Device d1r1z2-127.0.0.1:6202R127.0.0.1:6202/d2_"" with 100.0 weight got id 1
Device d1r1z2-127.0.0.1:6201R127.0.0.1:6201/d2_"" with 100.0 weight got id 1
Device d1r1z2-127.0.0.1:6200R127.0.0.1:6200/d2_"" with 100.0 weight got id 1
```

swift-ring-builder account.builder add r1z3-127.0.0.1:6202/d3 100
swift-ring-builder container.builder add r1z3-127.0.0.1:6201/d3 100
swift-ring-builder object.builder add r1z3-127.0.0.1:6200/d3 100



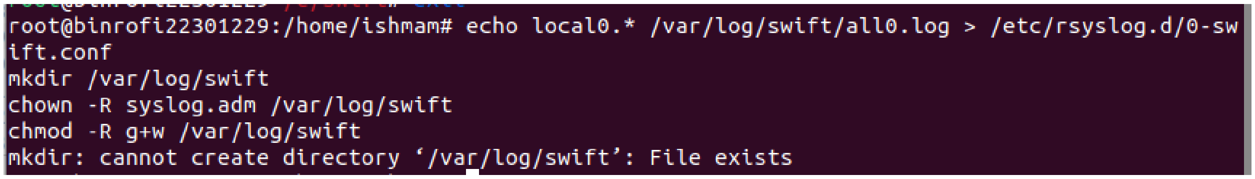
```
root@binrofi22301229 /e/swift# swift-ring-builder account.builder add r1z3-127.0.0
.1:6202/d3 100
                               swift-ring-builder container.builder add r1z3-127.0
.0.1:6201/d3 100
                               swift-ring-builder object.builder add r1z3-127.0.0.
1:6200/d3 100
Device d2r1z3-127.0.0.1:6202R127.0.0.1:6202/d3_"" with 100.0 weight got id 2
Device d2r1z3-127.0.0.1:6201R127.0.0.1:6201/d3_"" with 100.0 weight got id 2
Device d2r1z3-127.0.0.1:6200R127.0.0.1:6200/d3_"" with 100.0 weight got id 2
```

Now I will rebalance the ring here,
swift-ring-builder account.builder rebalance
swift-ring-builder container.builder rebalance
swift-ring-builder object.builder rebalance



```
Device d2r1z3-127.0.0.1:6200R127.0.0.1:6200/d3_"" with 100.0 weight got id 2
root@binrofi22301229 /e/swift# swift-ring-builder account.builder rebalance
                               swift-ring-builder container.builder rebalance
                               swift-ring-builder object.builder rebalance
Reassigned 24 (300.00%) partitions. Balance is now 0.00.  Dispersion is now 0.00
Reassigned 24 (300.00%) partitions. Balance is now 0.00.  Dispersion is now 0.00
Reassigned 24 (300.00%) partitions. Balance is now 0.00.  Dispersion is now 0.00
```

Now, we will check the log service, if it is running properly.



```
root@binrofi22301229 /e/swift# exit
root@binrofi22301229:/home/ishmam# echo local0.* /var/log/swift/all0.log > /etc/rsyslog.d/0-sw
ift.conf
mkdir /var/log/swift
chown -R syslog.adm /var/log/swift
chmod -R g+w /var/log/swift
mkdir: cannot create directory ‘/var/log/swift’: File exists
```

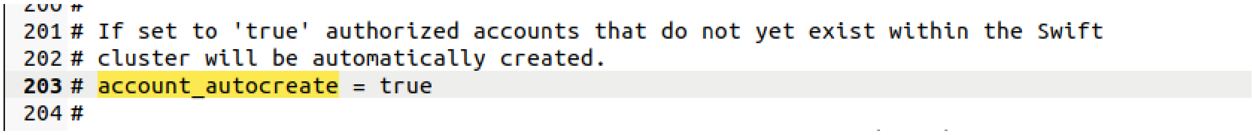
Now in the proxy configuration file I change the account management and account autocreate values to true.

cd /etc/swift/

gedit proxy-server.conf

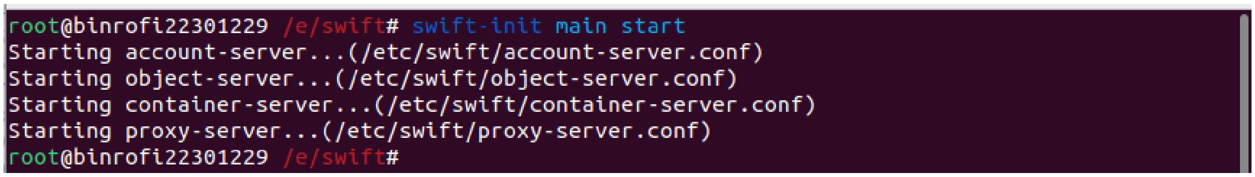
allow_account_management = true

account_autocreate = true



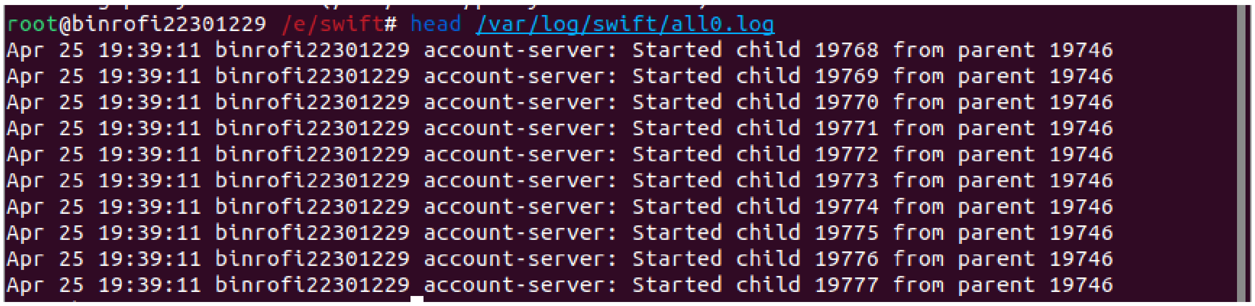
```
#
# If set to 'true' authorized accounts that do not yet exist within the Swift
# cluster will be automatically created.
# account_autocreate = true
#
```

Now I am starting all the main servers of the swift



```
root@binrofi22301229 /e/swift# swift-init main start
Starting account-server...(/etc/swift/account-server.conf)
Starting object-server...(/etc/swift/object-server.conf)
Starting container-server...(/etc/swift/container-server.conf)
Starting proxy-server...(/etc/swift/proxy-server.conf)
root@binrofi22301229 /e/swift#
```

Lets' check the log !

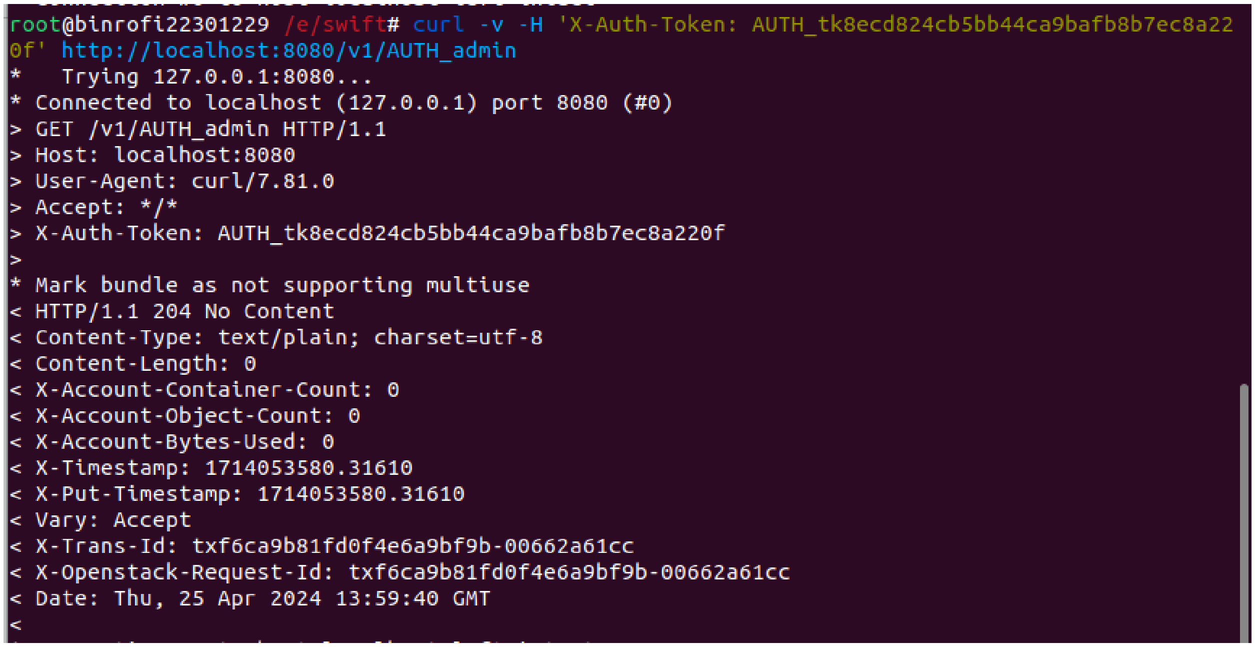


```
root@binrofi22301229 /e/swift# head /var/log/swift/all0.log
Apr 25 19:39:11 binrofi22301229 account-server: Started child 19768 from parent 19746
Apr 25 19:39:11 binrofi22301229 account-server: Started child 19769 from parent 19746
Apr 25 19:39:11 binrofi22301229 account-server: Started child 19770 from parent 19746
Apr 25 19:39:11 binrofi22301229 account-server: Started child 19771 from parent 19746
Apr 25 19:39:11 binrofi22301229 account-server: Started child 19772 from parent 19746
Apr 25 19:39:11 binrofi22301229 account-server: Started child 19773 from parent 19746
Apr 25 19:39:11 binrofi22301229 account-server: Started child 19774 from parent 19746
Apr 25 19:39:11 binrofi22301229 account-server: Started child 19775 from parent 19746
Apr 25 19:39:11 binrofi22301229 account-server: Started child 19776 from parent 19746
Apr 25 19:39:11 binrofi22301229 account-server: Started child 19777 from parent 19746
```

No errors ! 🥳🥳🥳🥳🥳

Step 4: Let me enterrrrrrrr

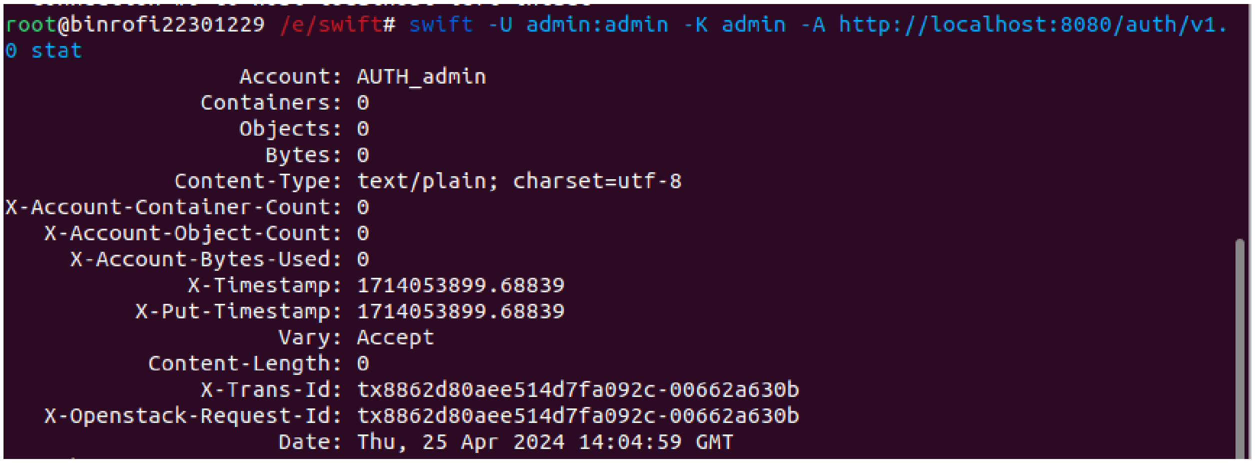
Now let’s use my authentication token to access my account.



```
root@binrofi22301229 /e/swift# curl -v -H 'X-Auth-Token: AUTH_tk8ecd824cb5bb44ca9bafb8b7ec8a22
0f' http://localhost:8080/v1/AUTH_admin
*   Trying 127.0.0.1:8080...
* Connected to localhost (127.0.0.1) port 8080 (#0)
> GET /v1/AUTH_admin HTTP/1.1
> Host: localhost:8080
> User-Agent: curl/7.81.0
> Accept: */*
> X-Auth-Token: AUTH_tk8ecd824cb5bb44ca9bafb8b7ec8a220f
>
* Mark bundle as not supporting multiuse
< HTTP/1.1 204 No Content
< Content-Type: text/plain; charset=utf-8
< Content-Length: 0
< X-Account-Container-Count: 0
< X-Account-Object-Count: 0
< X-Account-Bytes-Used: 0
< X-Timestamp: 1714053580.31610
< X-Put-Timestamp: 1714053580.31610
< Vary: Accept
< X-Trans-Id: txf6ca9b81fd0f4e6a9bf9b-00662a61cc
< X-Openstack-Request-Id: txf6ca9b81fd0f4e6a9bf9b-00662a61cc
< Date: Thu, 25 Apr 2024 13:59:40 GMT
<
```

It works ! Finally it works ! 6 times Ubuntu installation finally pays off 🥳🥳🥳🥳🥳

Now I am logging in with my swift cli. It works tooooooooooo



```
root@binrofi22301229 /e/swift# swift -U admin:admin -K admin -A http://localhost:8080/auth/v1.
0 stat
                  Account: AUTH_admin
               Containers: 0
                  Objects: 0
                    Bytes: 0
             Content-Type: text/plain; charset=utf-8
X-Account-Container-Count: 0
   X-Account-Object-Count: 0
     X-Account-Bytes-Used: 0
              X-Timestamp: 1714053899.68839
          X-Put-Timestamp: 1714053899.68839
                     Vary: Accept
           Content-Length: 0
               X-Trans-Id: tx8862d80aee514d7fa092c-00662a630b
   X-Openstack-Request-Id: tx8862d80aee514d7fa092c-00662a630b
                     Date: Thu, 25 Apr 2024 14:04:59 GMT
```

Step 4: Upload and Download

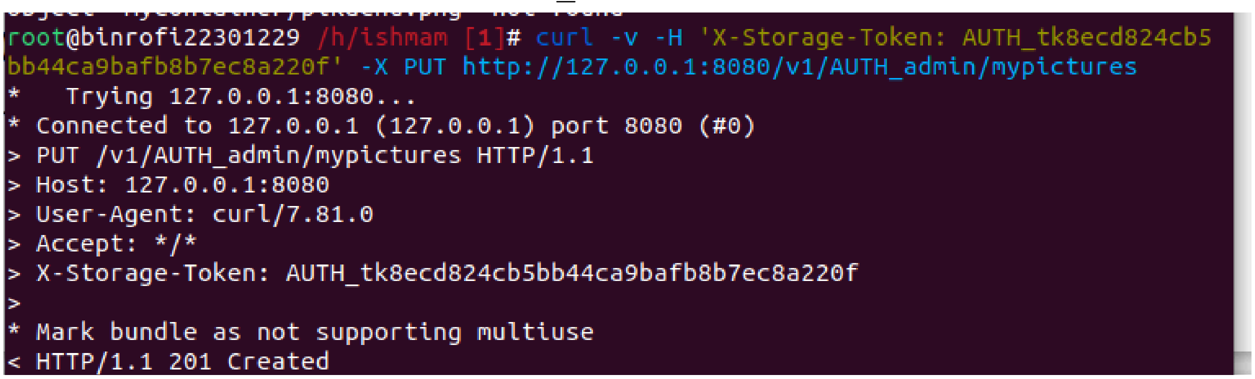
Now let’s upload this Pikachu to my drive using curl and swift
The file is at file:///home/ishmam/Desktop/pikachu.png

Create a container using curl,
NOTE: Use the STORAGE TOKEN from running this command

‘curl -v -H 'X-Auth-User: admin:admin' -H 'X-Auth-Key: admin' http://localhost:8080/auth/v1.0’

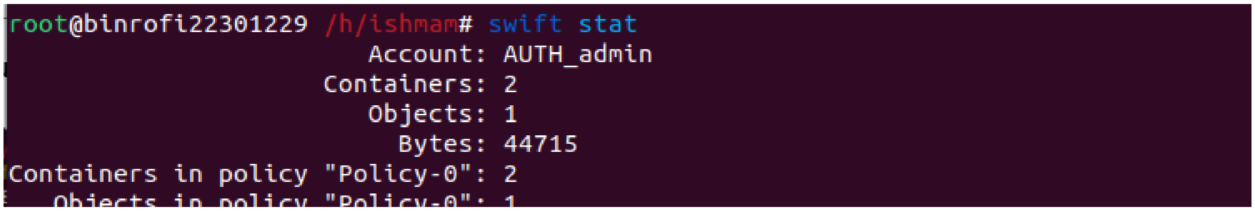
Now create a container using the storage that was found from the previous command:

```
curl -v -H 'X-Storage-Token:
AUTH_tk8ecd824cb5bb44ca9bafb8b7ec8a220f' -X PUT
http://127.0.0.1:8080/v1/AUTH_admin/mypictures
```



```
root@binrofi22301229 /h/ishmam [1]# curl -v -H 'X-Storage-Token: AUTH_tk8ecd824cb5
bb44ca9bafb8b7ec8a220f' -X PUT http://127.0.0.1:8080/v1/AUTH_admin/mypictures
*   Trying 127.0.0.1:8080...
* Connected to 127.0.0.1 (127.0.0.1) port 8080 (#0)
> PUT /v1/AUTH_admin/mypictures HTTP/1.1
> Host: 127.0.0.1:8080
> User-Agent: curl/7.81.0
> Accept: */*
> X-Storage-Token: AUTH_tk8ecd824cb5bb44ca9bafb8b7ec8a220f
>
* Mark bundle as not supporting multiuse
< HTTP/1.1 201 Created
```

After creating the container we can see it from `'swift stat'.` That there is a container created against my admin account.

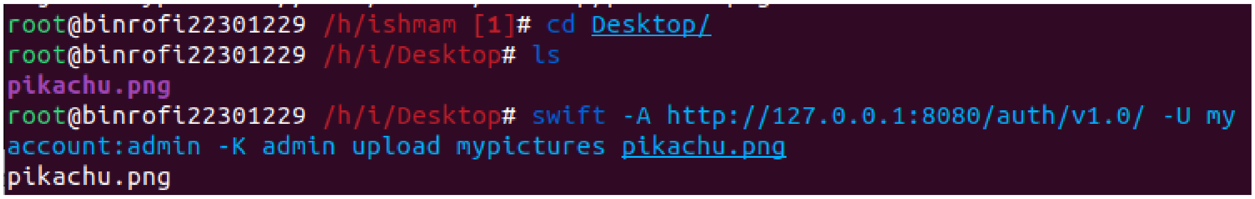


```
root@binrofi22301229 /h/ishmam# swift stat
                        Account: AUTH_admin
                     Containers: 2
                        Objects: 1
                          Bytes: 44715
Containers in policy "Policy-0": 2
   Objects in policy "Policy-0": 1
```

Now let's go to the folder where my Pikachu is doing pika pika. For me it was cd /home/ishmam/Desktop/ also check the existing files by using ls.

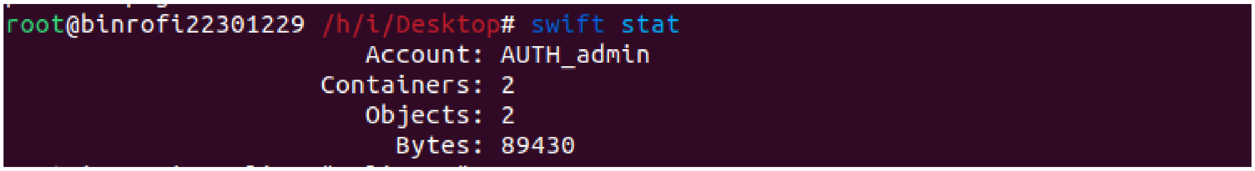
Now let's upload using this command:

```
swift -A http://127.0.0.1:8080/auth/v1.0/ -U myaccount:admin -K
admin upload mypictures pikachu.png
```



```
root@binrofi22301229 /h/ishmam [1]# cd Desktop/
root@binrofi22301229 /h/i/Desktop# ls
pikachu.png
root@binrofi22301229 /h/i/Desktop# swift -A http://127.0.0.1:8080/auth/v1.0/ -U my
account:admin -K admin upload mypictures pikachu.png
pikachu.png
```

Now let's check my stat, an object should increase !

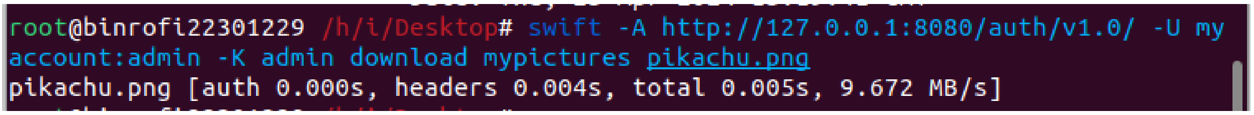


```
root@binrofi22301229 /h/i/Desktop# swift stat
                        Account: AUTH_admin
                     Containers: 2
                        Objects: 2
                          Bytes: 89430
```

It happened !

Now, let's take my Pikachu back from Swift.

```
' swift -A http://127.0.0.1:8080/auth/v1.0/ -U myaccount:admin
-K admin download mypictures pikachu.png '
```



```
root@binrofi22301229 /h/i/Desktop# swift -A http://127.0.0.1:8080/auth/v1.0/ -U my
account:admin -K admin download mypictures pikachu.png
pikachu.png [auth 0.000s, headers 0.004s, total 0.005s, 9.672 MB/s]
```

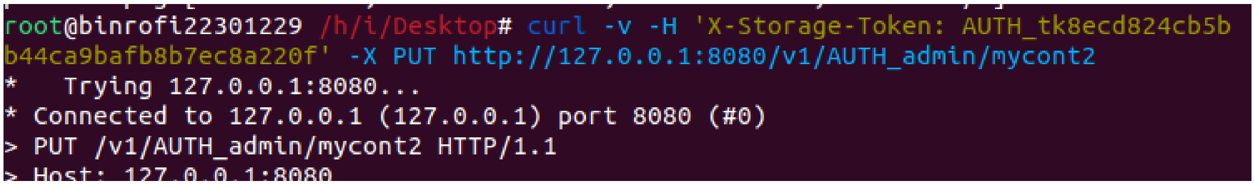
Downloaded the pikachu successfully.

Now let’s go with curl.

First let’s create another container where I will again create a container where I will keep my file using the curl command

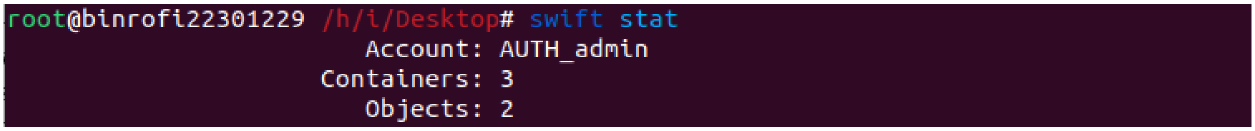
Here is the command to create another container, which is also shown in the stat,

```
‘curl -v -H 'X-Storage-Token:
AUTH_tk8ecd824cb5bb44ca9bafb8b7ec8a220f' -X PUT
http://127.0.0.1:8080/v1/AUTH_admin/mycont2’
```



```
root@binrofi22301229 /h/i/Desktop# curl -v -H 'X-Storage-Token: AUTH_tk8ecd824cb5b
b44ca9bafb8b7ec8a220f' -X PUT http://127.0.0.1:8080/v1/AUTH_admin/mycont2
*   Trying 127.0.0.1:8080...
* Connected to 127.0.0.1 (127.0.0.1) port 8080 (#0)
> PUT /v1/AUTH_admin/mycont2 HTTP/1.1
> Host: 127.0.0.1:8080
```

Here we can see that my container count is 3

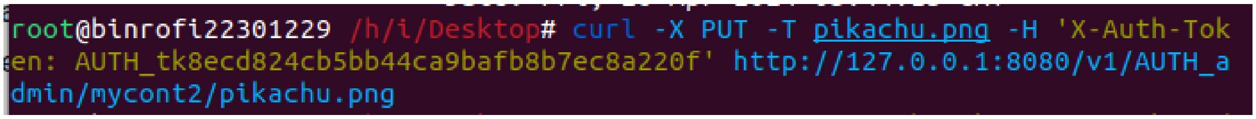


```
root@binrofi22301229 /h/i/Desktop# swift stat
                        Account: AUTH_admin
                     Containers: 3
                        Objects: 2
```

Now let’s upload the pikachu again.

Now using curl command upload the file to the **EXACT CONTAINER** that I created a minutes ago. Here is the command

```
curl -X PUT -T pikachu.png -H 'X-Auth-Token:
AUTH_tk8ecd824cb5bb44ca9bafb8b7ec8a220f'
http://127.0.0.1:8080/v1/AUTH_admin/mycont2/pikachu.png
```

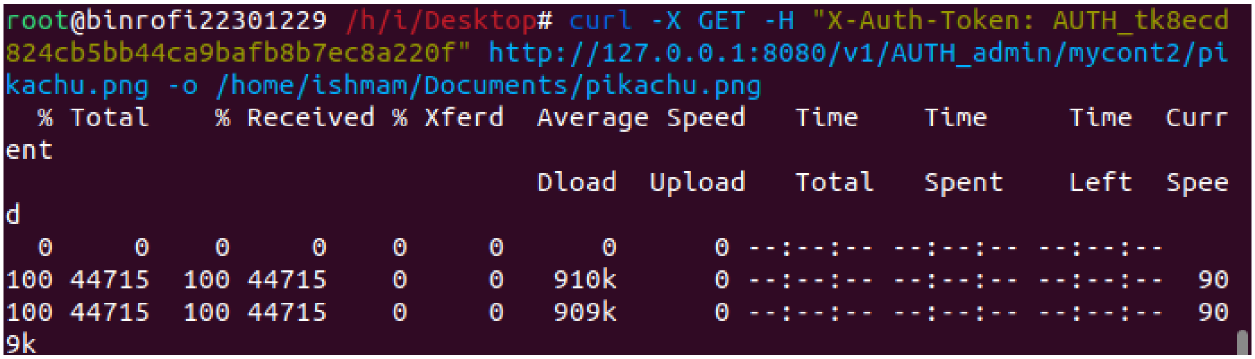


```
root@binrofi22301229 /h/i/Desktop# curl -X PUT -T pikachu.png -H 'X-Auth-Tok
en: AUTH_tk8ecd824cb5bb44ca9bafb8b7ec8a220f' http://127.0.0.1:8080/v1/AUTH_a
dmin/mycont2/pikachu.png
```

As there is no “error” message, we can under stand that the file has been uploaded successfully!

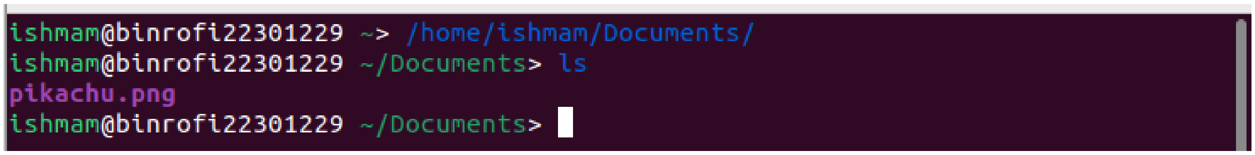
Now let’s download my pikachu using the download command. Here is the command that I used to download

```
‘ curl -X GET -H “X-AUTH-Token:<my token>”
http://127.0.0.1:8080/v1/AUTH_admin/<name_of_the_container>/<fil
e_name> -o <where_I_want_to_download_my_file>
```



```
root@binrofi22301229 /h/i/Desktop# curl -X GET -H "X-Auth-Token: AUTH_tk8ecd
824cb5bb44ca9bafb8b7ec8a220f" http://127.0.0.1:8080/v1/AUTH_admin/mycont2/pi
kachu.png -o /home/ishmam/Documents/pikachu.png
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Curr
ent
                                 Dload  Upload   Total   Spent    Left  Spee
d
  0     0    0     0    0     0      0      0 --:--:-- --:--:-- --:--:--
100 44715  100 44715    0     0   910k      0 --:--:-- --:--:-- --:--:--   90
100 44715  100 44715    0     0   909k      0 --:--:-- --:--:-- --:--:--   90
9k
```

Here is my pikachu downloaded successfully.



```
ishmam@binrofi22301229 ~> /home/ishmam/Documents/
ishmam@binrofi22301229 ~/Documents> ls
pikachu.png
ishmam@binrofi22301229 ~/Documents> 
```

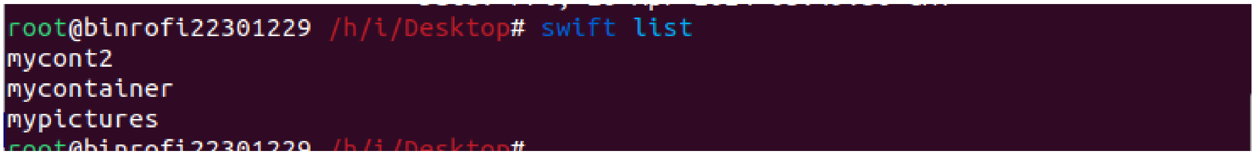
Continued in the next page.

Step 4: Swifty Swift !

Now let’s check the commands using swift commands on my swift client.

Listing swift: Let’s list all my containers using swift client command

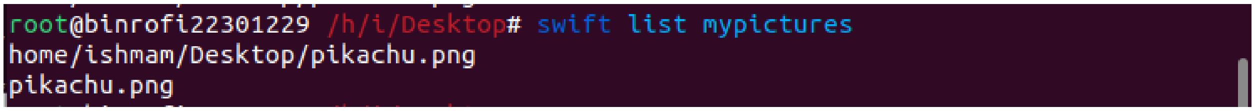
`swift list`



```
root@binrofi22301229 /h/i/Desktop# swift list
mycont2
mycontainer
mypictures
root@binrofi22301229 /h/i/Desktop#
```

Listing Swift to see the containers: Let’s use swift to see what’s in our containers.

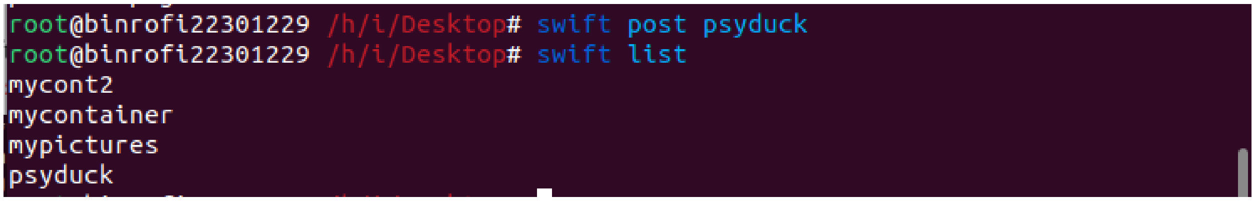
Command: `swift list <your_container_name>`



```
root@binrofi22301229 /h/i/Desktop# swift list mypictures
home/ishmam/Desktop/pikachu.png
pikachu.png
```

Swift post: Let’s post a container in my swift server using swift client.

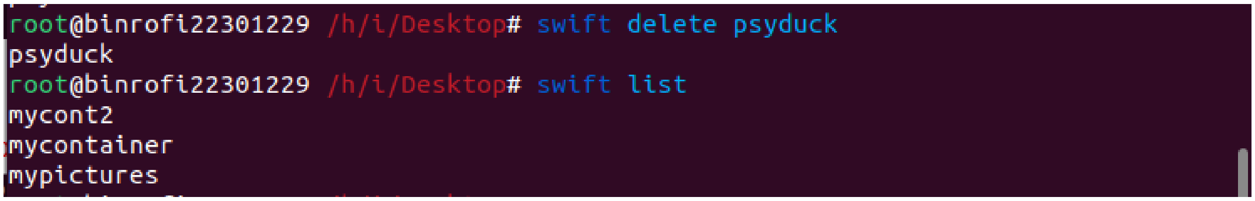
Command: `swift post <your_new_container_name>`



```
root@binrofi22301229 /h/i/Desktop# swift post psyduck
root@binrofi22301229 /h/i/Desktop# swift list
mycont2
mycontainer
mypictures
psyduck
```

Swift delete: Let’s delete a server from my cloud.

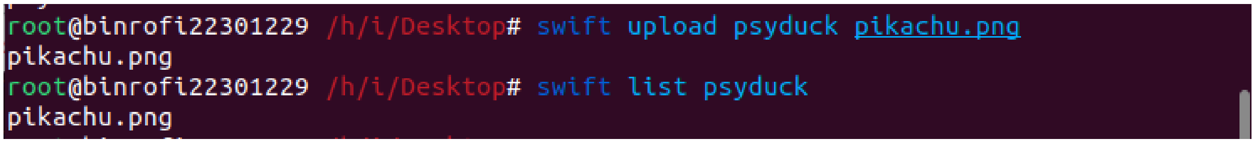
Command: `swift delete your_container_name`



```
root@binrofi22301229 /h/i/Desktop# swift delete psyduck
psyduck
root@binrofi22301229 /h/i/Desktop# swift list
mycont2
mycontainer
mypictures
```

Uploader Swift: Now let's upload a file to our container.

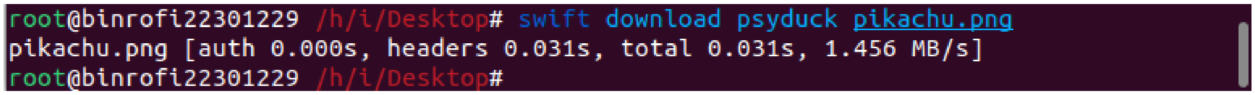
Command: `swift upload your_container_name /path/to/your/file`



```
root@binrofi22301229 /h/i/Desktop# swift upload psyduck pikachu.png
pikachu.png
root@binrofi22301229 /h/i/Desktop# swift list psyduck
pikachu.png
```

Downloader Swift: Now let's download a file from our container.

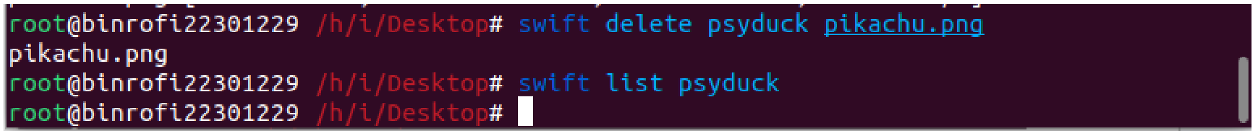
Command: `swift download your_container_name your_file_name`



```
root@binrofi22301229 /h/i/Desktop# swift download psyduck pikachu.png
pikachu.png [auth 0.000s, headers 0.031s, total 0.031s, 1.456 MB/s]
root@binrofi22301229 /h/i/Desktop#
```

Deleter Swift: Now let's delete a file from our container.

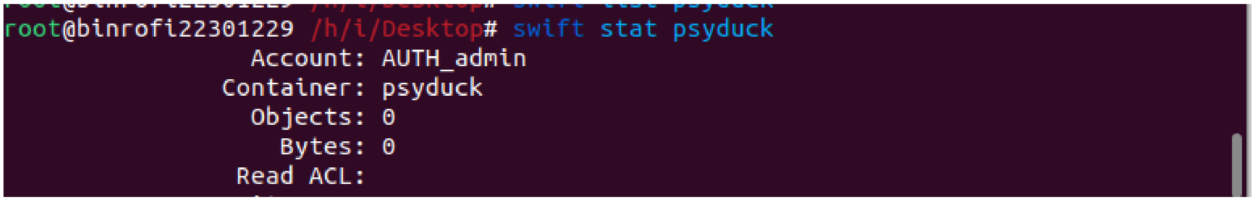
Command: `swift delete your_container_name your_file_name`



```
root@binrofi22301229 /h/i/Desktop# swift delete psyduck pikachu.png
pikachu.png
root@binrofi22301229 /h/i/Desktop# swift list psyduck
root@binrofi22301229 /h/i/Desktop#
```

Metadata Swift: Now let's get a metadata of a swift of a container.

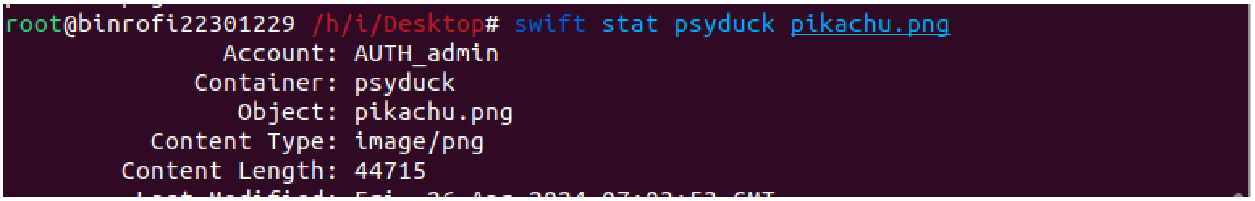
Command: `swift stat your_container_name`



```
root@binrofi22301229 /h/i/Desktop# swift list psyduck
root@binrofi22301229 /h/i/Desktop# swift stat psyduck
                 Account: AUTH_admin
               Container: psyduck
                 Objects: 0
                   Bytes: 0
                Read ACL:
```

Metadata Swift Object: Now let's get a metadata of a swift of a object.

Command: `swift stat your_container_name your_file_name`

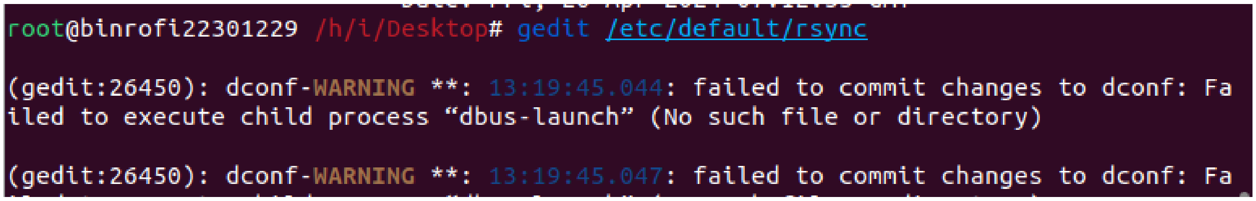


```
root@binrofi22301229 /h/i/Desktop# swift stat psyduck pikachu.png
               Account: AUTH_admin
             Container: psyduck
                Object: pikachu.png
          Content Type: image/png
        Content Length: 44715
         Last Modified: Fri, 26 Apr 2024 07:03:53 GMT
```

Step 4: CopyCat Swift, The Replication Process

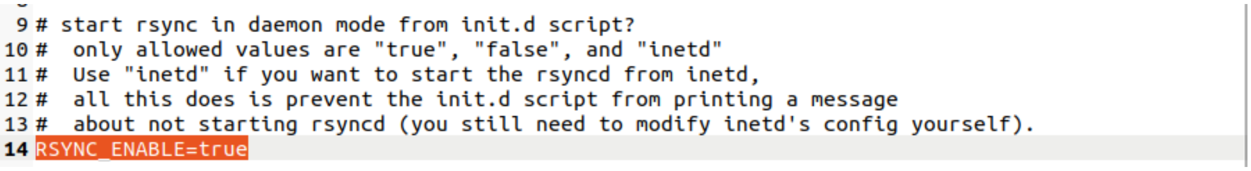
Now let’s check the commands using swift commands on my swift client.

First, we need to make RSYNC_ENABLE=true from our rsync file which can be accessed from
etc/default/rsync
So run this command gedit /etc/default/rsync



```
root@binrofi22301229 /h/i/Desktop# gedit /etc/default/rsync

(gedit:26450): dconf-WARNING **: 13:19:45.044: failed to commit changes to dconf: Fa
iled to execute child process “dbus-launch” (No such file or directory)

(gedit:26450): dconf-WARNING **: 13:19:45.047: failed to commit changes to dconf: Fa
```

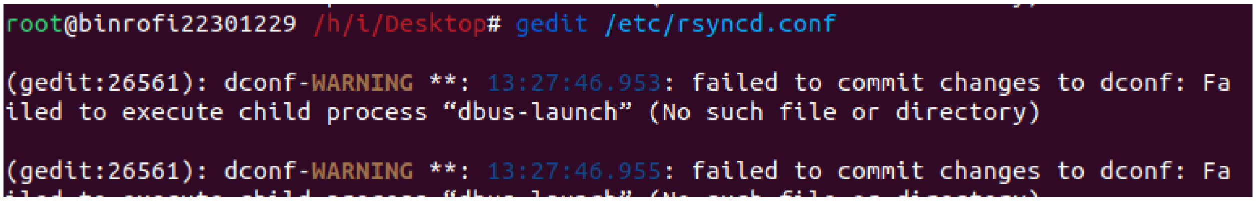
Then using the text editor alter the value to “true” in the line “RSYNC_ENABLE”



```
# start rsync in daemon mode from init.d script?
#  only allowed values are "true", "false", and "inetd"
#  Use "inetd" if you want to start the rsyncd from inetd,
#  all this does is prevent the init.d script from printing a message
#  about not starting rsyncd (you still need to modify inetd's config yourself).
RSYNC_ENABLE=true
```

Now save the file using ctrl + s

Now I have to create a rsync configuration file in /etc/rsyncd.conf.
For that I will use the command gedit /etc/rsyncd.conf



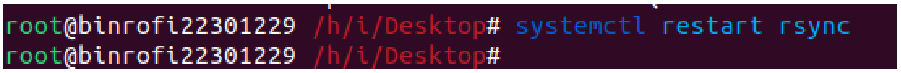
```
root@binrofi22301229 /h/i/Desktop# gedit /etc/rsyncd.conf

(gedit:26561): dconf-WARNING **: 13:27:46.953: failed to commit changes to dconf: Fa
iled to execute child process “dbus-launch” (No such file or directory)

(gedit:26561): dconf-WARNING **: 13:27:46.955: failed to commit changes to dconf: Fa
```

In that file, I will copy paste the following commands:

```
uid = swift
gid= swift
log file = /var/log/rsyncd.log
pid file = /var/run/rsyncd.pid
[account]
max connections = 25
path = /srv/node/
read only = false
lock file = /var/lock/account.lock
[container]
max connections = 25
path = /srv/node/
read only = false
lock file = /var/lock/container.lock
[object]
max connections = 25
path = /srv/node/
read only = false
lock file = /var/lock/object.lock
```

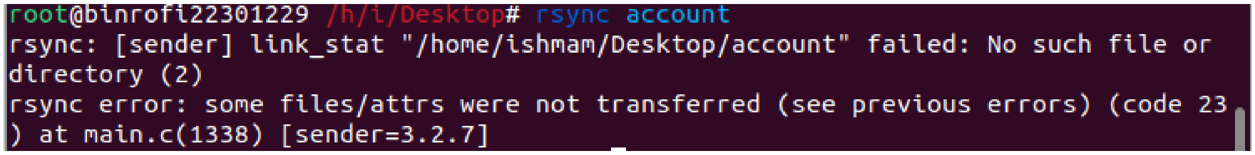
Now save the configuration file using ctrl + s.

In the next step we are going to restart rsync by using the command ‘systemctl restart rsync’



```
root@binrofi22301229 /h/i/Desktop# systemctl restart rsync
root@binrofi22301229 /h/i/Desktop#
```

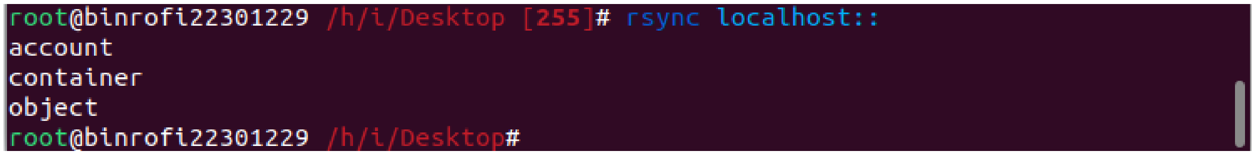
Now let’s check our rsync account using ‘rsync account’



```
root@binrofi22301229 /h/i/Desktop# rsync account
rsync: [sender] link_stat "/home/ishmam/Desktop/account" failed: No such file or
directory (2)
rsync error: some files/attrs were not transferred (see previous errors) (code 23
) at main.c(1338) [sender=3.2.7]
```

Now let’s check rsync localhost

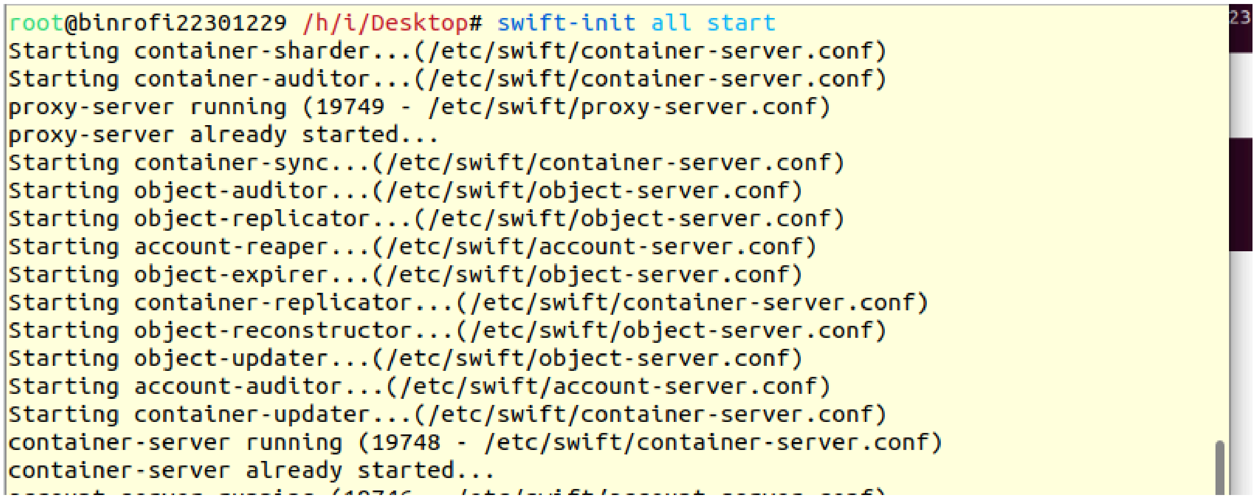
Command: ‘rsync localhost::’



```
root@binrofi22301229 /h/i/Desktop [255]# rsync localhost::
account
container
object
root@binrofi22301229 /h/i/Desktop#
```

If it shows an account, container, object, it’s working!

Now let’s start everything on swift using ‘swift-init all start’



```
root@binrofi22301229 /h/i/Desktop# swift-init all start
Starting container-sharder...(/etc/swift/container-server.conf)
Starting container-auditor...(/etc/swift/container-server.conf)
proxy-server running (19749 - /etc/swift/proxy-server.conf)
proxy-server already started...
Starting container-sync...(/etc/swift/container-server.conf)
Starting object-auditor...(/etc/swift/object-server.conf)
Starting object-replicator...(/etc/swift/object-server.conf)
Starting account-reaper...(/etc/swift/account-server.conf)
Starting object-expirer...(/etc/swift/object-server.conf)
Starting container-replicator...(/etc/swift/container-server.conf)
Starting object-reconstructor...(/etc/swift/object-server.conf)
Starting object-updater...(/etc/swift/object-server.conf)
Starting account-auditor...(/etc/swift/account-server.conf)
Starting container-updater...(/etc/swift/container-server.conf)
container-server running (19748 - /etc/swift/container-server.conf)
container-server already started...
```

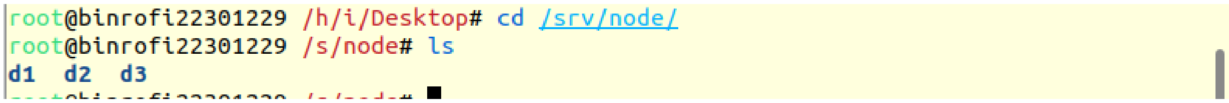
Here we can see that our sharder, auditor, replicator, sync, reaper, reconstructor etc has started working and our system is now running.

Now lets check it.

For that we have to go to node folder using

Command: cd /srv/node

After that using ls we can see that there are d1, d2 and d3

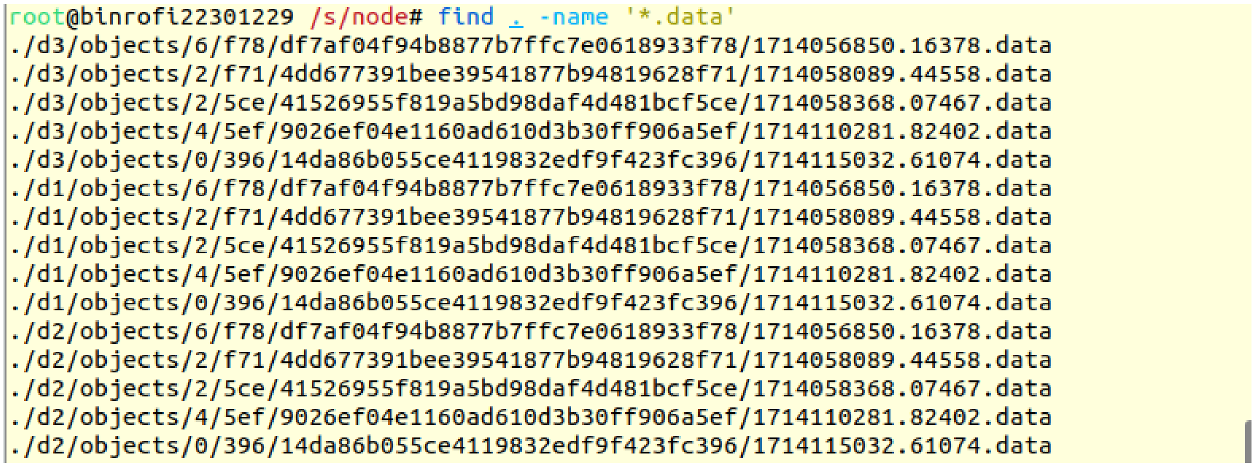


```
root@binrofi22301229 /h/i/Desktop# cd /srv/node/
root@binrofi22301229 /s/node# ls
d1  d2  d3
```

After that I am going to check the files that I have uploaded so far, using the command:

```
find . -name '*.data'
```

It's a wildcard command that will fetch all the files ending with .data



```
root@binrofi22301229 /s/node# find . -name '*.data'
./d3/objects/6/f78/df7af04f94b8877b7ffc7e0618933f78/1714056850.16378.data
./d3/objects/2/f71/4dd677391bee39541877b94819628f71/1714058089.44558.data
./d3/objects/2/5ce/41526955f819a5bd98daf4d481bcf5ce/1714058368.07467.data
./d3/objects/4/5ef/9026ef04e1160ad610d3b30ff906a5ef/1714110281.82402.data
./d3/objects/0/396/14da86b055ce4119832edf9f423fc396/1714115032.61074.data
./d1/objects/6/f78/df7af04f94b8877b7ffc7e0618933f78/1714056850.16378.data
./d1/objects/2/f71/4dd677391bee39541877b94819628f71/1714058089.44558.data
./d1/objects/2/5ce/41526955f819a5bd98daf4d481bcf5ce/1714058368.07467.data
./d1/objects/4/5ef/9026ef04e1160ad610d3b30ff906a5ef/1714110281.82402.data
./d1/objects/0/396/14da86b055ce4119832edf9f423fc396/1714115032.61074.data
./d2/objects/6/f78/df7af04f94b8877b7ffc7e0618933f78/1714056850.16378.data
./d2/objects/2/f71/4dd677391bee39541877b94819628f71/1714058089.44558.data
./d2/objects/2/5ce/41526955f819a5bd98daf4d481bcf5ce/1714058368.07467.data
./d2/objects/4/5ef/9026ef04e1160ad610d3b30ff906a5ef/1714110281.82402.data
./d2/objects/0/396/14da86b055ce4119832edf9f423fc396/1714115032.61074.data
```

Here we can see that each of our files are stored by replication in three different replications,

For example, the file ending with 16378.data has copies in d1, d2 and d3

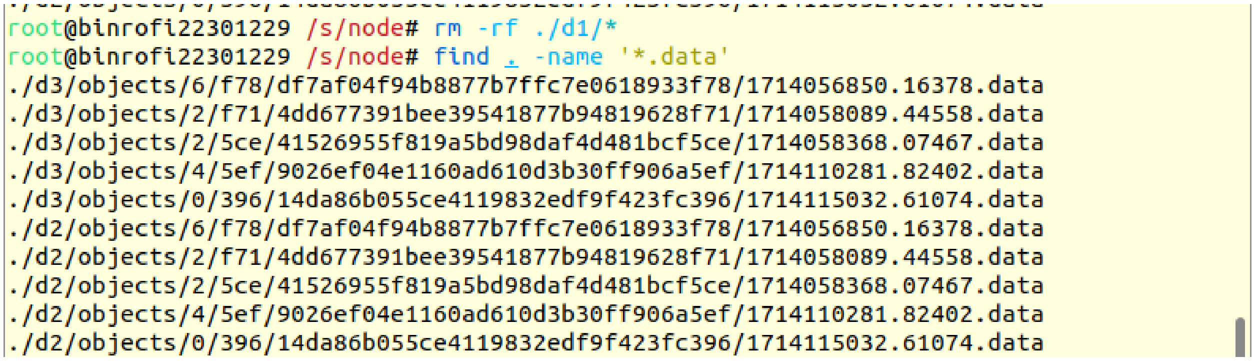
Now I will delete one whole drive to check and show the replication process

Using the command:

rm -rf ./d1/*

This command will delete everything from d1

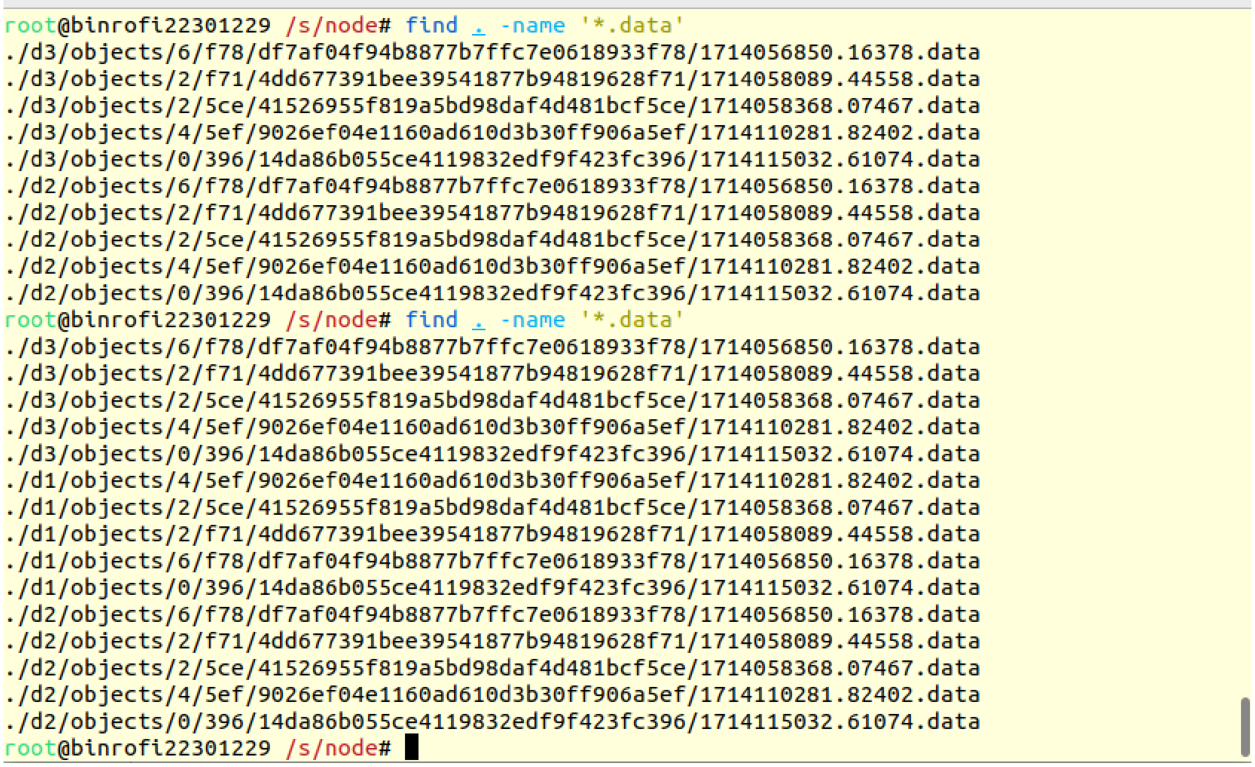
Now here I can see that there is no data at d1 !



```
root@binrofi22301229 /s/node# rm -rf ./d1/*
root@binrofi22301229 /s/node# find . -name '*.data'
./d3/objects/6/f78/df7af04f94b8877b7ffc7e0618933f78/1714056850.16378.data
./d3/objects/2/f71/4dd677391bee39541877b94819628f71/1714058089.44558.data
./d3/objects/2/5ce/41526955f819a5bd98daf4d481bcf5ce/1714058368.07467.data
./d3/objects/4/5ef/9026ef04e1160ad610d3b30ff906a5ef/1714110281.82402.data
./d3/objects/0/396/14da86b055ce4119832edf9f423fc396/1714115032.61074.data
./d2/objects/6/f78/df7af04f94b8877b7ffc7e0618933f78/1714056850.16378.data
./d2/objects/2/f71/4dd677391bee39541877b94819628f71/1714058089.44558.data
./d2/objects/2/5ce/41526955f819a5bd98daf4d481bcf5ce/1714058368.07467.data
./d2/objects/4/5ef/9026ef04e1160ad610d3b30ff906a5ef/1714110281.82402.data
./d2/objects/0/396/14da86b055ce4119832edf9f423fc396/1714115032.61074.data
```

** continued at next page **

Now let's wait for 15-20 seconds and check the drives again !
Now see ! all the data is back at d1 ! why ? Because the replicator replicated all the data from d2 and d3 to d1 and created the backup copy. This maintains the replication, so if one drive of mine is gone, I can get my support from another drive ! And this is how replication is done!



```
root@binrofi22301229 /s/node# find . -name '*.data'
./d3/objects/6/f78/df7af04f94b8877b7ffc7e0618933f78/1714056850.16378.data
./d3/objects/2/f71/4dd677391bee39541877b94819628f71/1714058089.44558.data
./d3/objects/2/5ce/41526955f819a5bd98daf4d481bcf5ce/1714058368.07467.data
./d3/objects/4/5ef/9026ef04e1160ad610d3b30ff906a5ef/1714110281.82402.data
./d3/objects/0/396/14da86b055ce4119832edf9f423fc396/1714115032.61074.data
./d2/objects/6/f78/df7af04f94b8877b7ffc7e0618933f78/1714056850.16378.data
./d2/objects/2/f71/4dd677391bee39541877b94819628f71/1714058089.44558.data
./d2/objects/2/5ce/41526955f819a5bd98daf4d481bcf5ce/1714058368.07467.data
./d2/objects/4/5ef/9026ef04e1160ad610d3b30ff906a5ef/1714110281.82402.data
./d2/objects/0/396/14da86b055ce4119832edf9f423fc396/1714115032.61074.data
root@binrofi22301229 /s/node# find . -name '*.data'
./d3/objects/6/f78/df7af04f94b8877b7ffc7e0618933f78/1714056850.16378.data
./d3/objects/2/f71/4dd677391bee39541877b94819628f71/1714058089.44558.data
./d3/objects/2/5ce/41526955f819a5bd98daf4d481bcf5ce/1714058368.07467.data
./d3/objects/4/5ef/9026ef04e1160ad610d3b30ff906a5ef/1714110281.82402.data
./d3/objects/0/396/14da86b055ce4119832edf9f423fc396/1714115032.61074.data
./d1/objects/4/5ef/9026ef04e1160ad610d3b30ff906a5ef/1714110281.82402.data
./d1/objects/2/5ce/41526955f819a5bd98daf4d481bcf5ce/1714058368.07467.data
./d1/objects/2/f71/4dd677391bee39541877b94819628f71/1714058089.44558.data
./d1/objects/6/f78/df7af04f94b8877b7ffc7e0618933f78/1714056850.16378.data
./d1/objects/0/396/14da86b055ce4119832edf9f423fc396/1714115032.61074.data
./d2/objects/6/f78/df7af04f94b8877b7ffc7e0618933f78/1714056850.16378.data
./d2/objects/2/f71/4dd677391bee39541877b94819628f71/1714058089.44558.data
./d2/objects/2/5ce/41526955f819a5bd98daf4d481bcf5ce/1714058368.07467.data
./d2/objects/4/5ef/9026ef04e1160ad610d3b30ff906a5ef/1714110281.82402.data
./d2/objects/0/396/14da86b055ce4119832edf9f423fc396/1714115032.61074.data
root@binrofi22301229 /s/node# █
```

All thanks to Allah, the assignment is now done and the OpenStack Swift SAIO is now running successfully.

I really enjoyed setting up the OpenStack Swift SAIO, it also gave us a thorough understanding of the cloud environment and how it works, though we all struggled, it was fun and hard at the same time.

"You've got no reason to be afraid
You're on your own, kid
Yeah, you can face this" - Taylor SWIFT (Another Swift :))