**Problem Statement:**

* At 8:30am we want to take the backup of DB.
* At 8:40am we want to upload DB backup to gdrive and delete previous backup from gdrive.
* At 5:30pm again take db backup.
* At 5:40pm upload db backup to gdrive and delete previous backup from gdrive.

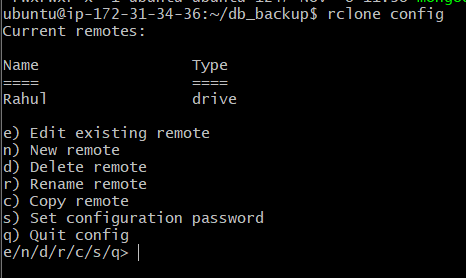
### **Prerequisites**

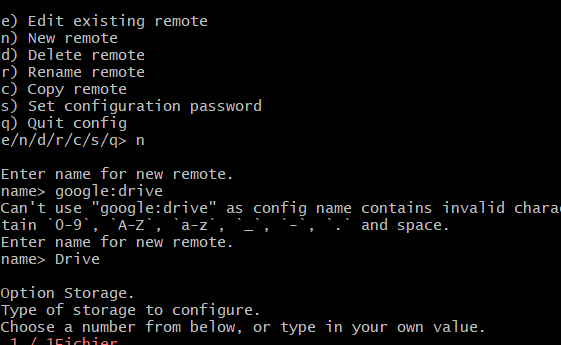
**Install mongodb or mongod** for MongoDB backups:

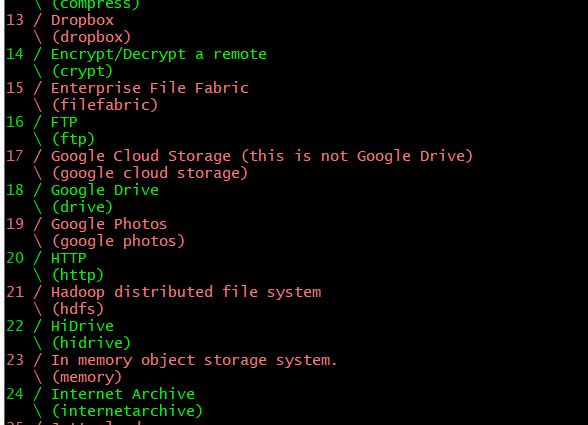
**Install rclone** to interact with Google Drive:  
sudo apt-get install -y rclone

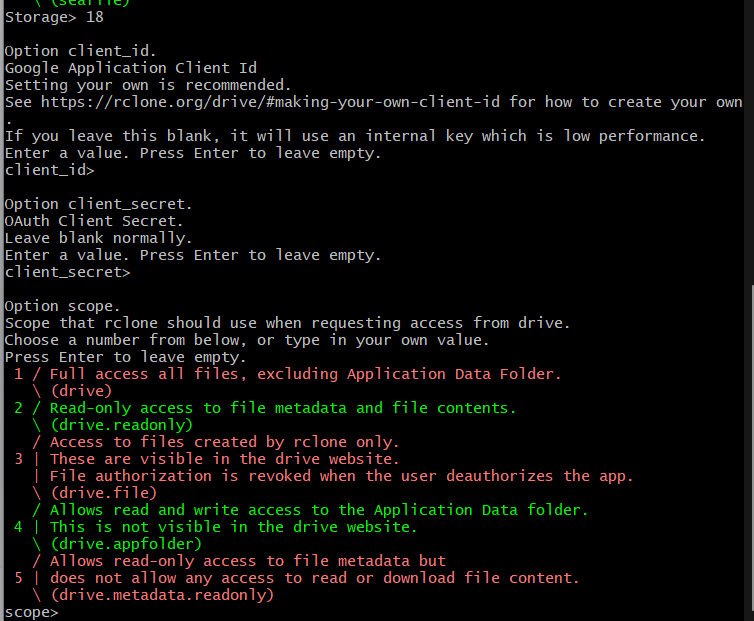
Configure rclone with Google Drive by running:  
rclone config

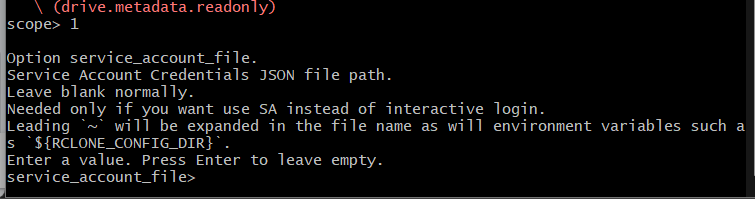
1. Follow the prompts to set up access to your Google Drive and name it (e.g., gdrive).

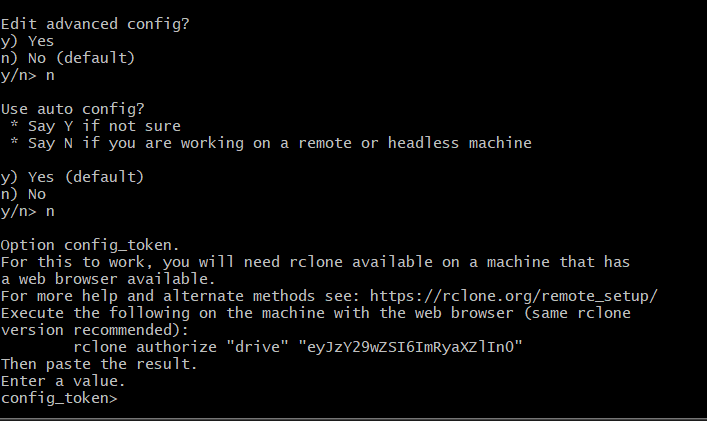










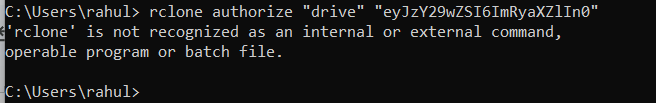


Install rclone for windows

<https://rclone.org/downloads/>

Extract the file.

Without path set

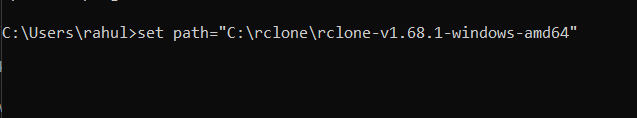


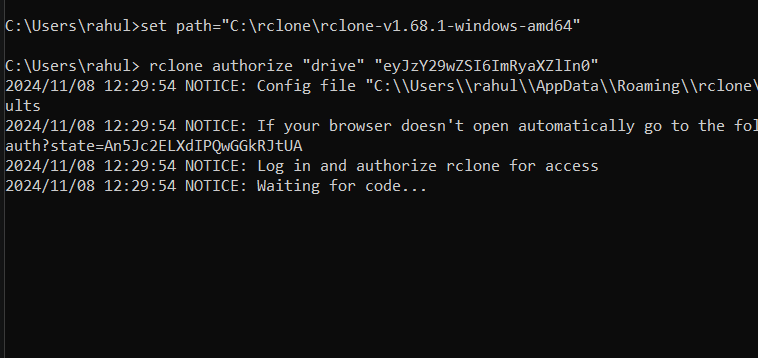
Set the path into environmental variable.

OR

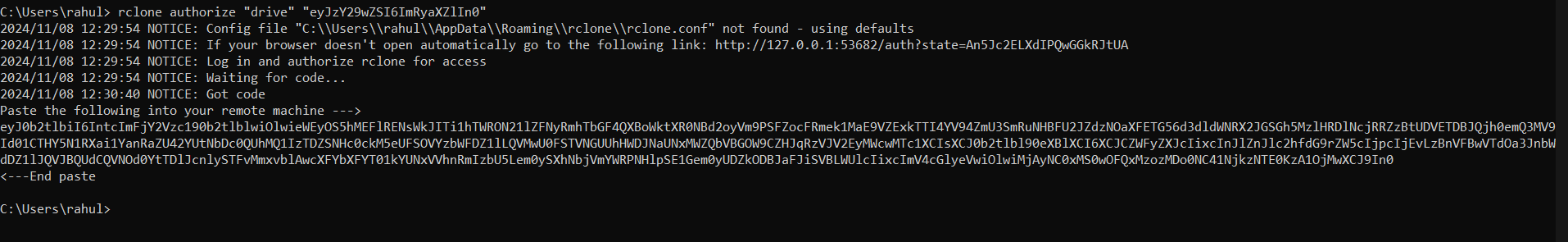
set path=” ”

For that particular terminal





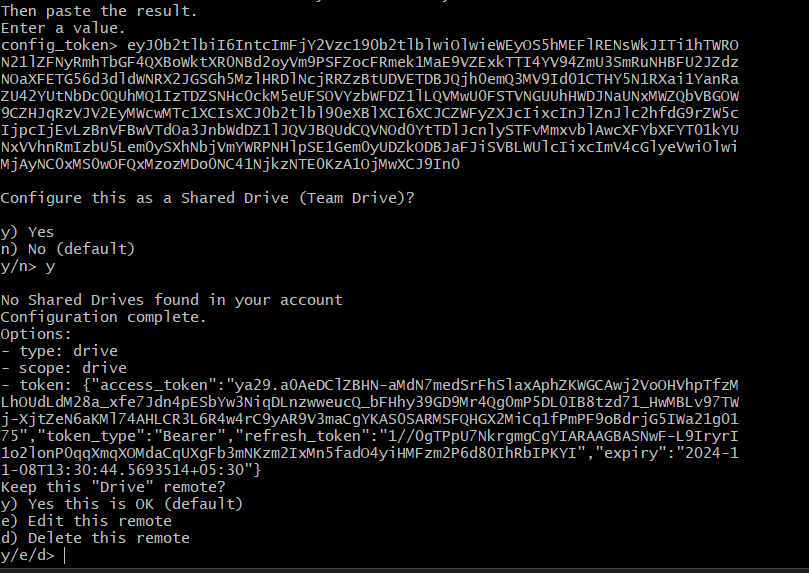


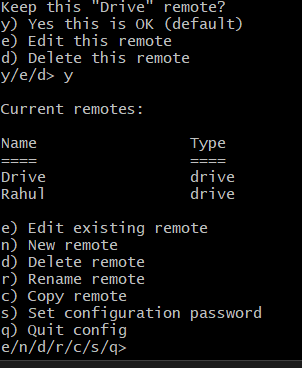


Paste the following into your remote machine --->



<---End paste





T**his bash script will create .zip file**

#!/bin/bash

# Define directories

LOCAL\_BACKUP\_DIR="/var/backups/mongodb"

GDRIVE\_DIR="Rahul:mongodb\_backups" # Use "Rahul" as the remote name

DATABASE\_NAME="your\_database\_name"

# Log file

LOG\_FILE="${LOCAL\_BACKUP\_DIR}/mongodb\_backup.log"

# Get current time

current\_time=$(date +"%H:%M")

# Determine backup file based on the time

if [ "$current\_time" == "11:40" ]; then

backup\_time="11\_40"

delete\_time="10:28"

elif [ "$current\_time" == "10:16" ]; then

backup\_time="10\_16"

delete\_time="08:40"

else

echo "This script should only run at 8:30 AM or 5:30 PM."

exit 1

fi

# Step 1: Take MongoDB backup

backup\_dir="${LOCAL\_BACKUP\_DIR}/${backup\_time}\_backup"

mongodump --db "$DATABASE\_NAME" --out "$backup\_dir"

# Remove any extra files in the target directory to avoid accidental uploads find "$backup\_dir" -type f -name "\*.sh" -delete

# Step 2: Compress the backup directory into a .tar.gz file

backup\_file="${LOCAL\_BACKUP\_DIR}/${backup\_time}\_backup.zip"

zip -r "$backup\_file" "$backup\_dir" .

# Step 3: Delete the backup directory after compression

rm -rf "$backup\_dir"

# Step 4: Delete the previous backup from Google Drive

rclone delete "${GDRIVE\_DIR}/${delete\_time}\_backup.zip"

# Step 5: Upload the new compressed backup to Google Drive

rclone copy "$backup\_file" "${GDRIVE\_DIR}/${backup\_time}\_backup.zip"

echo "Backup for ${backup\_time} completed, compressed, and uploaded to Google Drive."

**This bash script will directly make db copy and upload files directly to gdrive but not in zip**

#!/bin/bash

# Define directories

LOCAL\_BACKUP\_DIR="/var/backups/mongodb"

GDRIVE\_DIR="Rahul:mongodb\_backups" # Use "Rahul" as the remote name

DATABASE\_NAME="your\_database\_name" # Specify the database to back up

# Log file

LOG\_FILE="${LOCAL\_BACKUP\_DIR}/mongodb\_backup.log"

# Get current time

current\_time=$(date +"%H:%M")

# Determine backup time based on the time

if [ "$current\_time" == "15:47" ]; then

backup\_time="15\_47"

delete\_time="17\_40"

elif [ "$current\_time" == "17:30" ]; then

backup\_time="17\_30"

delete\_time="8\_40"

else

echo "This script should only run at 8:30 AM or 5:30 PM." | tee -a "$LOG\_FILE"

exit 1

fi

# Step 1: Take MongoDB backup of the specific database

backup\_dir="${LOCAL\_BACKUP\_DIR}/${backup\_time}\_backup"

mongodump --db "$DATABASE\_NAME" --out "$backup\_dir" # Backup only the specified database

# Remove any extra files in the target directory to avoid accidental uploads

find "$backup\_dir" -type f -name "\*.sh" -delete

# Step 2: Clear previous backups in Google Drive

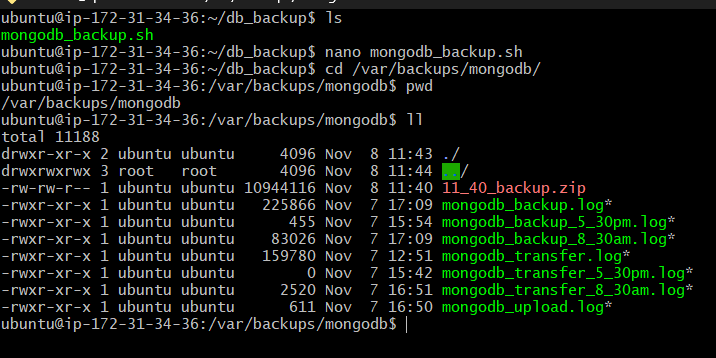
rclone purge "${GDRIVE\_DIR}"

# Step 3: Upload only the MongoDB backup directory to Google Drive

rclone copy "$backup\_dir/$DATABASE\_NAME" "${GDRIVE\_DIR}/${backup\_time}\_backup" --create-empty-src-dirs

# Log success

echo "Backup for ${backup\_time} of database '${DATABASE\_NAME}' completed and uploaded to Google Drive." | tee -a "$LOG\_FILE"



Set all the permissions for ubuntu user.

Run the cmd

crontab -e

# Backup at 8:30 AM and 5:30 PM

40 11 \* \* \* /home/ubuntu/db\_backup/mongodb\_backup.sh # Backup at 8:30 AM

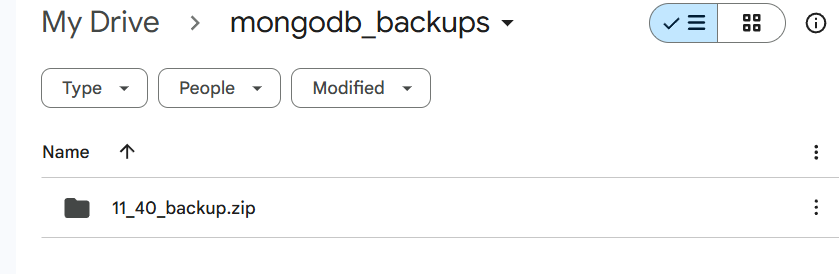
16 10 \* \* \* /home/ubuntu/db\_backup/mongodb\_backup.sh # Backup at 5:30 PM

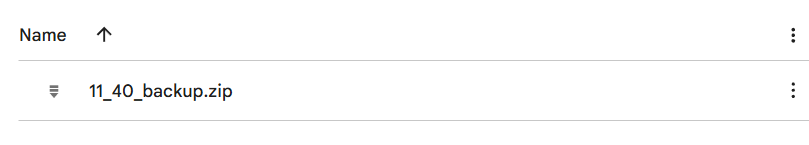
#Delete old backup and upload new backup at 8:40 AM and 5:40 PM

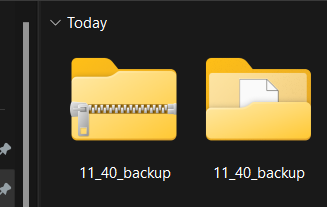
40 11 \* \* \* rclone delete Rahul:mongodb\_backups/10\_28\_backup.zip && rclone copy /var/backups/mongodb/8\_30\_backup.zip Rahul:mongodb\_backups/8\_30\_backup.zip

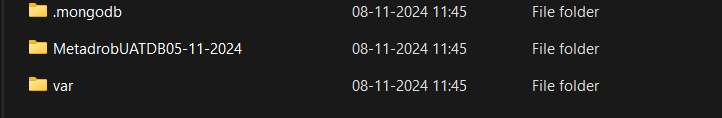
17 10 \* \* \* rclone delete Rahul:mongodb\_backups/8\_40\_backup.zip && rclone copy /var/backups/mongodb/17\_30\_backup.zip Rahul:mongodb\_backups/17\_30\_backup.zip

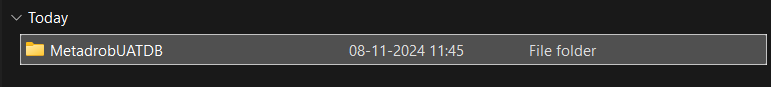
Change time according in cron job and also .sh file .

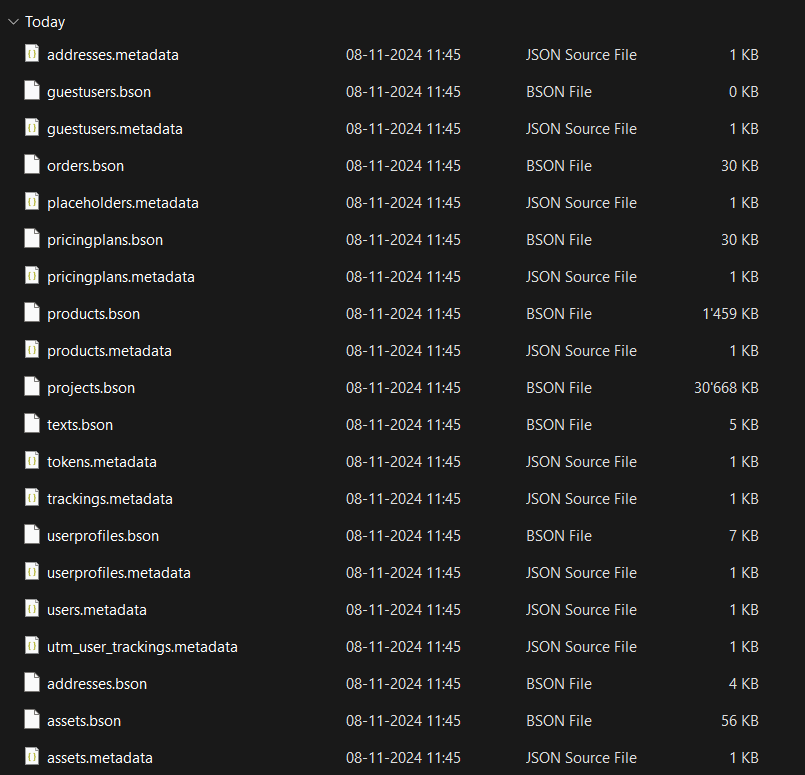


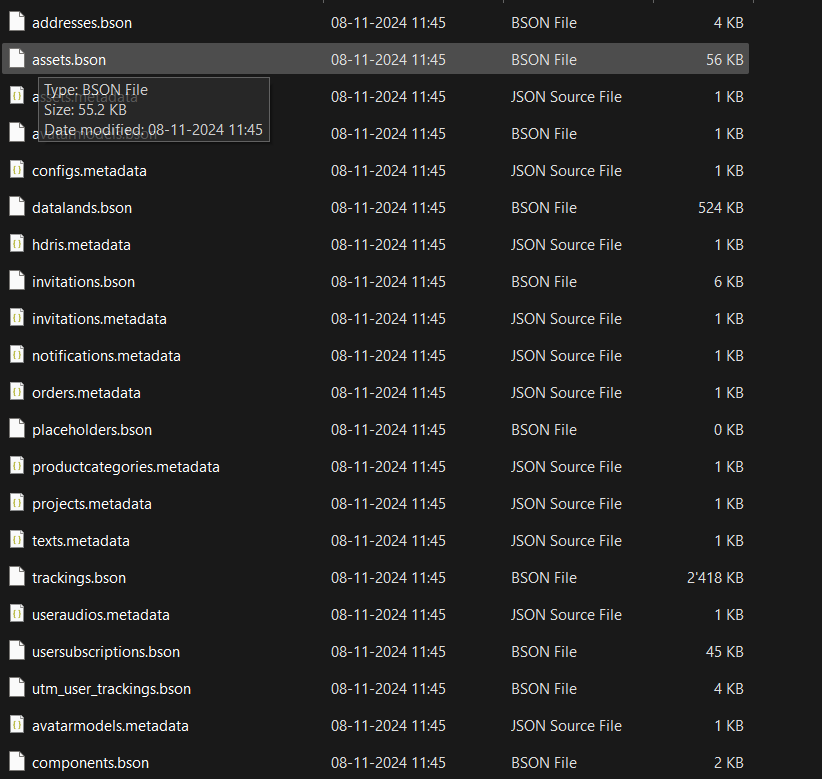




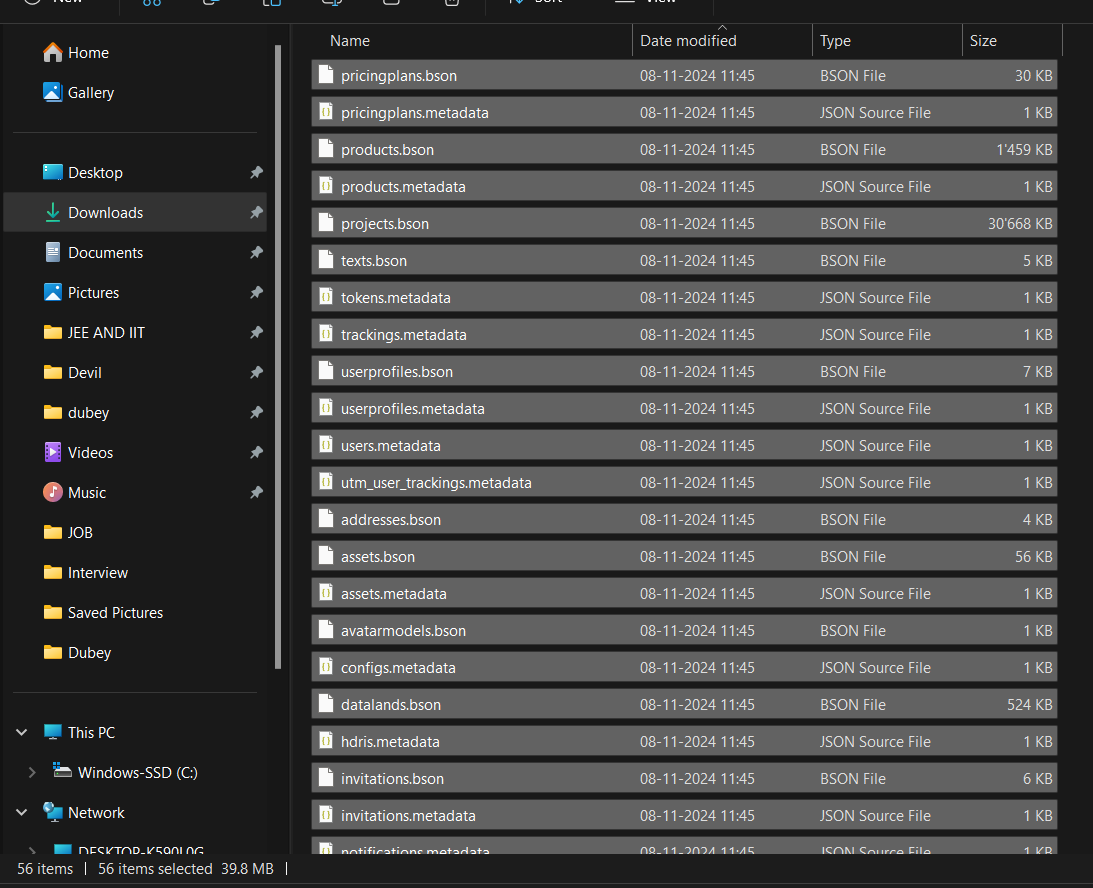


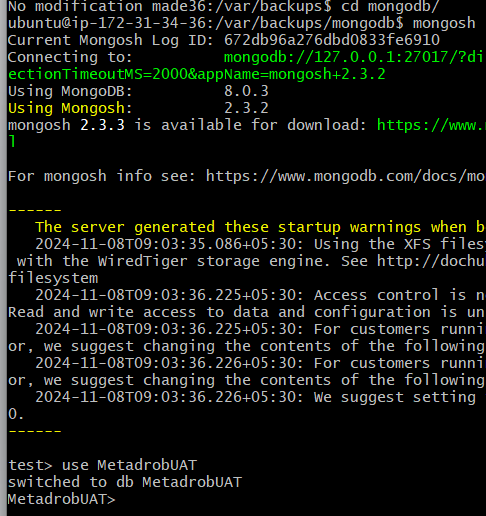


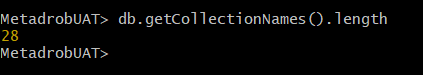




Total 56 time and 28 collection







Note:

* Set the required permission to run the job also for the .sh file.
* Set the required permission for ubuntu user to transfer file from local to gdrive and execute all the cmd.
* Check the time of ubuntu as IST no UTC.
* Change the time in .sh and cron job according to requirements.