DALHOUSIE UNIVERSITY

Serverless Data Processing

(CSCI-5410)

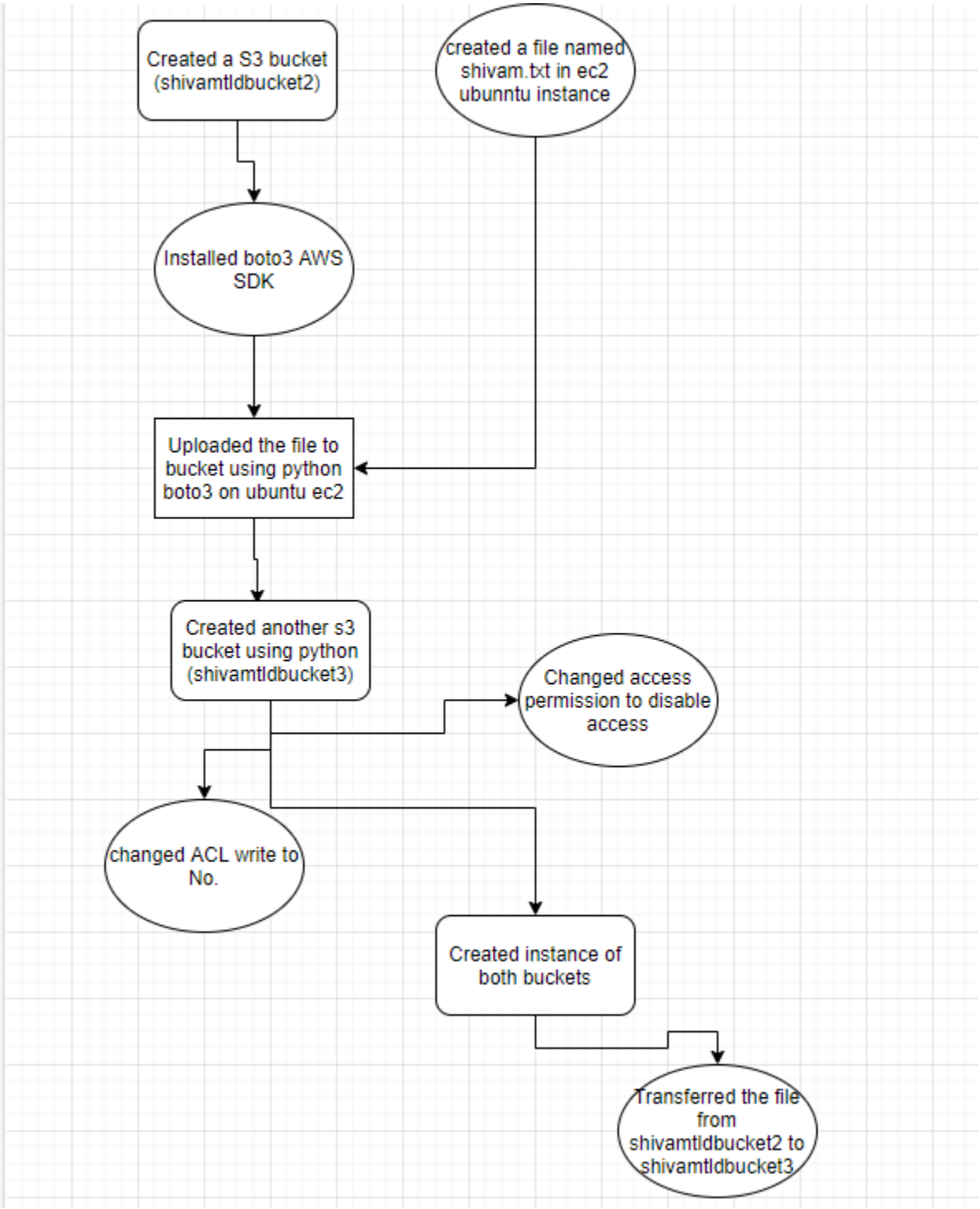
Part – B

AWS STORAGE EXPERIMENT

Submitted by:

Shivam Gupta (B00810723)

a) Flowchart:

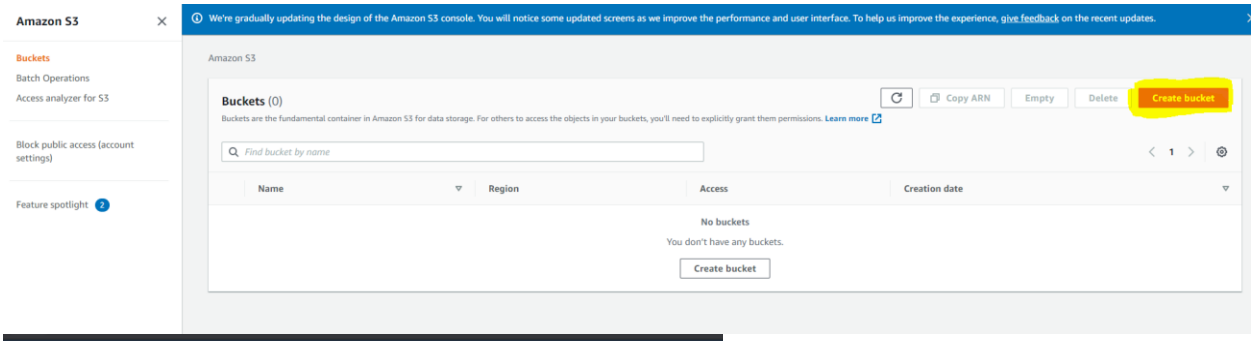

Created a S3 bucket (shivamtldbucket2)
created a file named shivam.txt in ec2 ubunntu instance
Installed boto3 AWS SDK
Uploaded the file to bucket using python boto3 on ubuntu ec2
Created another s3 bucket using python (shivamtldbucket3)
Changed access permission to disable access
changed ACL write to No.
Created instance of both buckets
Transferred the file from shivamtldbucket2 to shivamtldbucket3

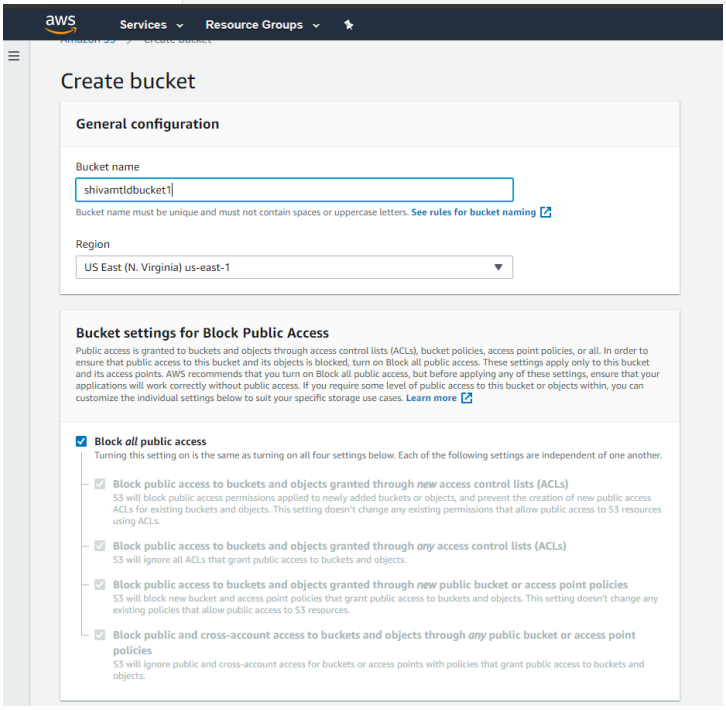
b) Paragraph:

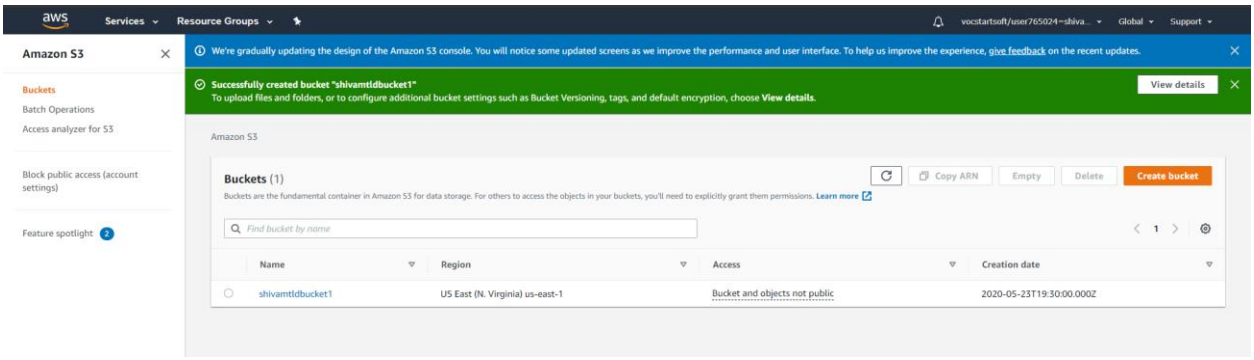
AWS S3 bucket is used for storage on cloud. AWS offers an SDK kit known as boto3, which can be used to access S3 bucket by python and update/modify the bucket. In this experiment, we learned how to create AWS S3 bucket and access it on EC2 ubuntu instance and by using boto3 we can create and manage the S3 buckets. We created an S3 bucket manually and by using python as well. We uploaded a file on bucket by python SDK.

We change the access permissions and ACL operation permissions by python. Lastly, we transferred a file from one bucket to another.

c) Bucket created with the name shivamtldbucket2.



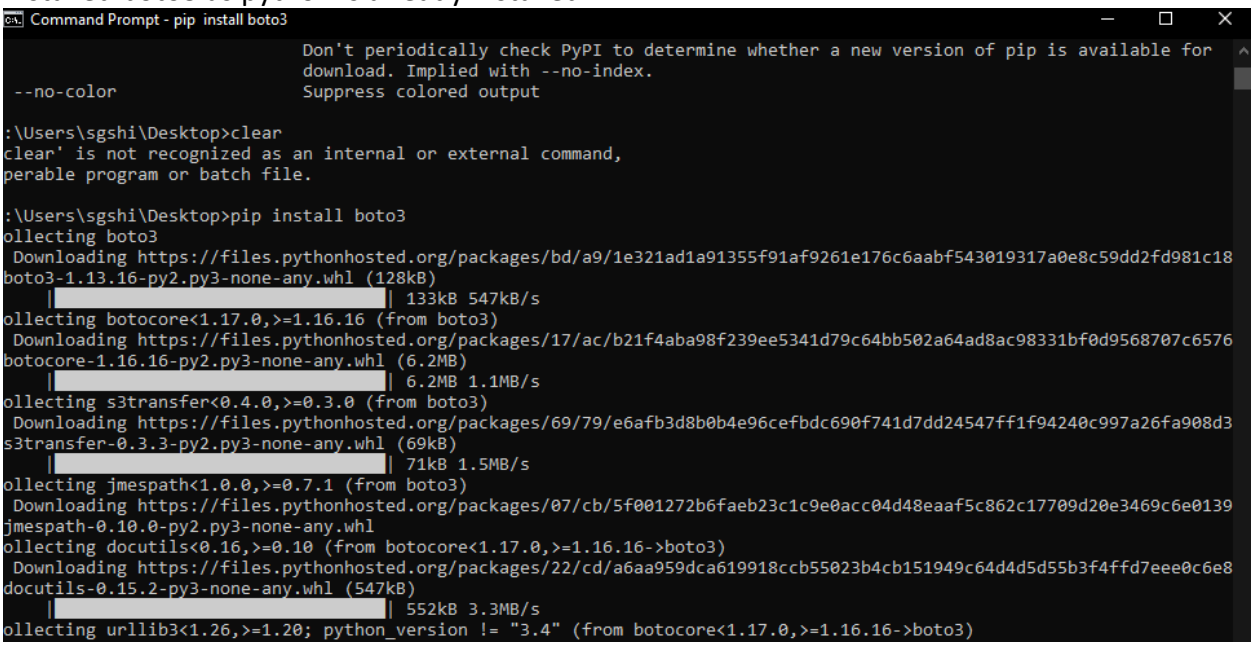




d) Explore AWS SDK for Python - and using python program, upload the file on the S3 bucket you created.

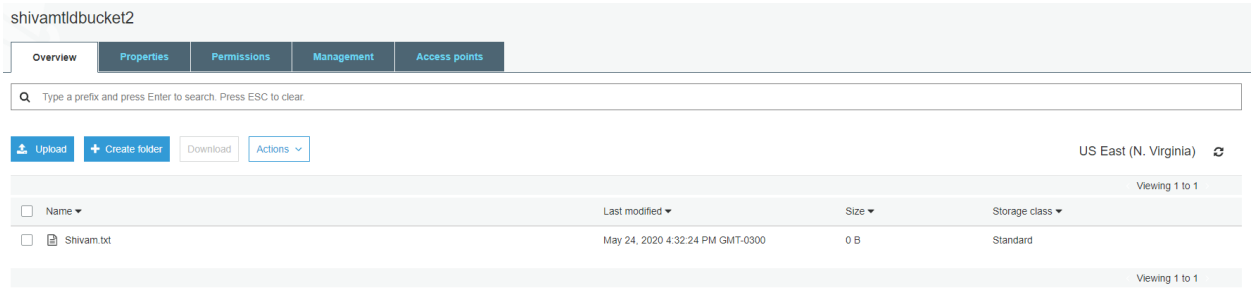
Adding user:

Installed boto3 as python is already installed:



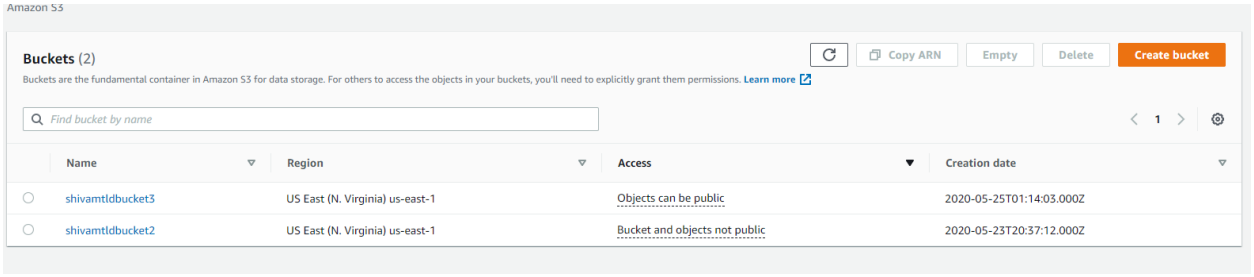
Python program to add file to S3:

File uploaded:

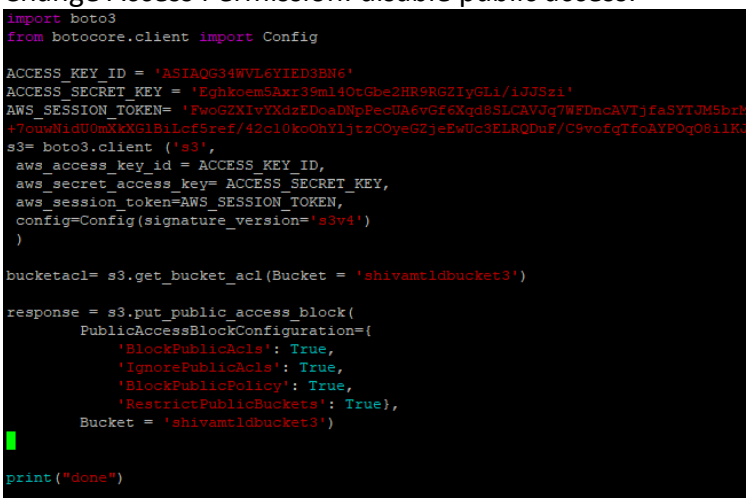


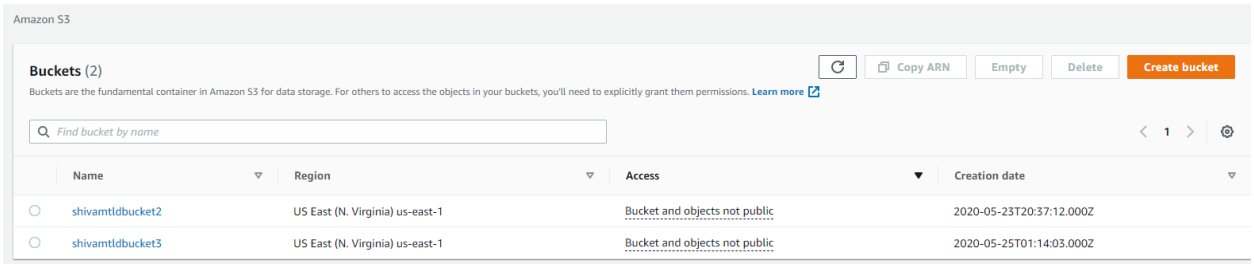
e) Code from creating new bucket:

New bucket created named shivamtldbucket3:



Change Access Permission: disable public access:



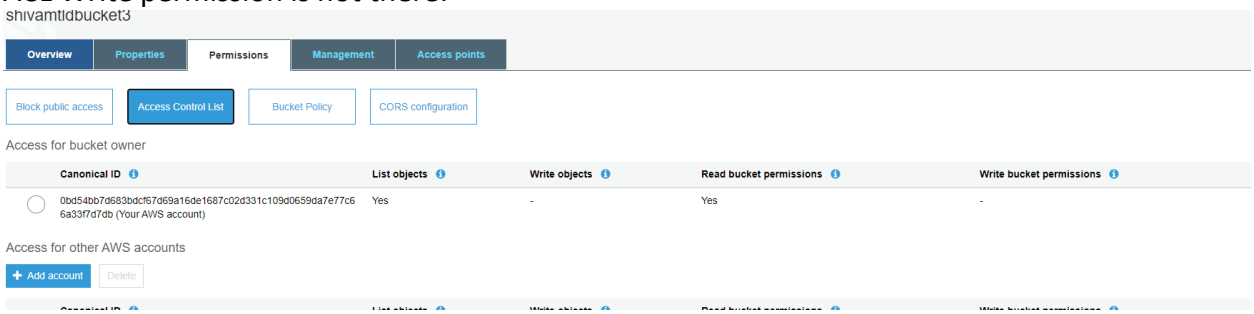


Change ACL write to No means changing it to Read.

Code continuation:

```
import boto3
from botocore.client import Config

ACCESS_KEY_ID = 'ASIAQG34WVL6YIED3BN6'
ACCESS_SECRET_KEY = 'Eghkoem5Axr39ml4OtGbe2HR9RGZIyGLi/iJJSzi'
AWS_SESSION_TOKEN= 'FwoGZXIvYXdzEDoaDNpPecUA6vGf6Xqd8SLCAVJq7WFDncAVTjfaSYTJM5brM9W5Zo0Z4sXQUQxCEWmK4IoV5a
+7ouwNidU0mXkXGlBiLcf5ref/42cl0koOhYljtzCOyeGZjeEwUc3ELRQDuF/C9vofqTfoAYPOqO8ilKJCurPYFMilYerlGJ/f95gyGlpR
s3= boto3.client ('s3',
 aws_access_key_id = ACCESS_KEY_ID,
 aws_secret_access_key= ACCESS_SECRET_KEY,
 aws_session_token=AWS_SESSION_TOKEN,
 config=Config(signature_version='s3v4')
 )

response = s3.put_public_access_block(
        PublicAccessBlockConfiguration={
            'BlockPublicAcls': True,
            'IgnorePublicAcls': True,
            'BlockPublicPolicy': True,
            'RestrictPublicBuckets': True},
        Bucket = 'shivamtldbucket3')
s4= boto3.resource ('s3',
 aws_access_key_id = ACCESS_KEY_ID,
 aws_secret_access_key= ACCESS_SECRET_KEY,
 aws_session_token=AWS_SESSION_TOKEN,
 config=Config(signature_version='s3v4')
 )
bucket_acl=s4.BucketAcl('shivamtldbucket3')

response=bucket_acl.put(
        AccessControlPolicy= {
            'Owner':{
                'DisplayName': 'awslabsc0w747645t1587490972',
                'ID': '0bd54bb7d683bdcf67d69a16de1687c02d331c109d0659da7e77c66a33f7d7db'},
            'Grants':[
                {'Grantee':{
                    'Type': 'CanonicalUser',
                    'DisplayName': 'awslabsc0w747645t1587490972',
                    'ID': '0bd54bb7d683bdcf67d69a16de1687c02d331c109d0659da7e77c66a33f7d7db'},
                    'Permission': 'READ_ACP'},
                {'Grantee':{
                    'Type': 'CanonicalUser',
                    'DisplayName': 'awslabsc0w747645t1587490972',
                    'ID': '0bd54bb7d683bdcf67d69a16de1687c02d331c109d0659da7e77c66a33f7d7db'},
                    'Permission': 'READ'
                }]})

print("done")

~
```

ACL Write permission is not there:



f) Move file from bucket shivamtldbucket2 to bucket shivamtldbucker3:
Python program:

```
import boto3
from botocore.client import Config

ACCESS_KEY_ID = 'ASIAQG34WVL6YIED3BN6'
ACCESS_SECRET_KEY = 'Eghkoem5Axr39ml4OtGbe2HR9RGZIyGLi/iJJSzi'
AWS_SESSION_TOKEN= 'FwoGZXIvYXdzEDoaDNpPecUA6vGf6Xqd8SLCAVJq7WFDncAVTjfaSYTJM5brM9W5Zo0Z4sXQUQxCEWmK4IoV5a
+7ouwNidU0mXkXGlBiLcf5ref/42c10koOhYljtzCOyeGZjeEwUc3ELRQDuF/C9vofqTfoAYPOqO8ilKJCurPYFMilYerlGJ/f95gyGlpR
s3= boto3.client ('s3',
 aws_access_key_id = ACCESS_KEY_ID,
 aws_secret_access_key= ACCESS_SECRET_KEY,
 aws_session_token=AWS_SESSION_TOKEN,
 config=Config(signature_version='s3v4')
 )
s4= boto3.resource ('s3',
 aws_access_key_id = ACCESS_KEY_ID,
 aws_secret_access_key= ACCESS_SECRET_KEY,
 aws_session_token=AWS_SESSION_TOKEN,
 config=Config(signature_version='s3v4')
 )

bucket1= s4.Bucket('shivamtldbucket2')
bucket2=s4.Bucket('shivamtldbucket3')


for i in bucket1.objects.all():
    fname= i.key.split('/')[-1]
    bucket2.put_object(Key='fromshivamtldbucket2/'+ fname, Body= i.get()["Body"].read())

print("Done")

~
~
~
```

Program Script:

File upload:

```
import boto3
from botocore.client import Config

ACCESS_KEY_ID = 'ASIAQG34WVL6QEH7PP54'
ACCESS_SECRET_KEY = 'UrctVnBQWQYuSF12j/PeI6nDchg1uTXrkc3Wfrw5'
AWS_SESSION_TOKEN='FwoGZXIvYXdzEDUaDP0rAFFph2DUvuMPVyLCAQCONKUIvCF4jVXRpoU
oD0yU+4pDApLplGlLY875vjT783P1LKwpe+Nd+JH/XDZ2HU8u4Bi/3fF3uyi/prsjBvuj0wlN1RYFt7Ls
EAuy0rTCWbfJESrmkrDCpxbrj1Mi7vGo6nitHxMgVQTpHxqvKLnx6wFr9RuZzZberhtlYGHJTNrtnZe
PeCeZiVEGb7PJIJt3NTHGBXKhmftn2r5NAx2keNk2EmtiveZdmYo3J+S4PboRQbB9shtyTJ5Fk0qGIJ
/jKN+Mq/YFMi0n6saXMvIhvzhQ6qUPzh1zgjTUd/UlbtH1UgQfEf2Tixwmv+oV9ECmug75bG0'
BUCKET_NAME = 'shivamtldbucket2'

data = open('Shivam.txt', 'rb')

s3= boto3.resource ('s3',
 aws_access_key_id = ACCESS_KEY_ID,
 aws_secret_access_key= ACCESS_SECRET_KEY,
 aws_session_token=AWS_SESSION_TOKEN,
 config=Config(signature_version='s3v4')
 )
s3.Bucket(BUCKET_NAME).put_object(Key='Shivam.txt', Body= data)

print("done")
```

Create bucket:

```
import boto3
from botocore.client import Config

ACCESS_KEY_ID = 'ASIAQG34WVL6YIED3BN6'
ACCESS_SECRET_KEY = 'Eghkoem5Axr39ml4OtGbe2HR9RGZIyGLi/iJJSzi'
AWS_SESSION_TOKEN=
'FwoGZXIvYXdzEDoaDNpPecUA6vGf6Xqd8SLCAVJq7WFDncAVTjfaSYTJM5brM9W5Zo0Z4sXQUQ
xCEWmK4IoV5aTxKUjStzDqy6xq/v8cFS5liZrqYbarfPLVRhI8rHv4kNeizBB5wqeiWcKBflOKlmpHPL/
fXQraPEjsV4bmEnWvOMEUnmzwT1zr7hlDSckDLAbyKyXeJY2Y0meR1pMU1pW8D+7ouwNidU0
mXkXGlBiLcf5ref/42c10koOhYljtzCOyeGZjeEwUc3ELRQDuF/C9vofqTfoAYPOqO8ilKJCurPYFMi1Y
er1GJ/f95gyG1pR3HTrNexyVEmdHqSIbq265a4W7BcG1rtn4G58MNcEZV0w='
s3= boto3.client ('s3',
 aws_access_key_id = ACCESS_KEY_ID,
 aws_secret_access_key= ACCESS_SECRET_KEY,
```

```
 aws_session_token=AWS_SESSION_TOKEN,
 config=Config(signature_version='s3v4')
 )
s3.create_bucket_Bucket='shivamtldbucket3'
response = s3.put_public_access_block(
       PublicAccessBlockConfiguration={
          'BlockPublicAcls': True,
          'IgnorePublicAcls': True,
          'BlockPublicPolicy': True,
          'RestrictPublicBuckets': True},
       Bucket = 'shivamtldbucket3')
s4= boto3.resource ('s3',
 aws_access_key_id = ACCESS_KEY_ID,
 aws_secret_access_key= ACCESS_SECRET_KEY,
 aws_session_token=AWS_SESSION_TOKEN,
 config=Config(signature_version='s3v4')
 )
bucket_acl=s4.BucketAcl('shivamtldbucket3')

response=bucket_acl.put(
       AccessControlPolicy= {
          'Owner':{
             'DisplayName': 'awslabsc0w747645t1587490972',
             'ID': '0bd54bb7d683bdcf67d69a16de1687c02d331c109d0659da7e77c66a33f7d7db'},
          'Grants':[
             {'Grantee':{
                'Type': 'CanonicalUser',
                'DisplayName': 'awslabsc0w747645t1587490972',
                'ID':
'0bd54bb7d683bdcf67d69a16de1687c02d331c109d0659da7e77c66a33f7d7db'},
                'Permission': 'READ_ACP'},
             {'Grantee':{
                'Type': 'CanonicalUser',
                'DisplayName': 'awslabsc0w747645t1587490972',
                'ID':
'0bd54bb7d683bdcf67d69a16de1687c02d331c109d0659da7e77c66a33f7d7db'},
                'Permission': 'READ'
             }]})

print("done")
```

Transfer file to another bucket:

```
import boto3
from botocore.client import Config

ACCESS_KEY_ID = 'ASIAQG34WVL6YIED3BN6'
ACCESS_SECRET_KEY = 'Eghkoem5Axr39ml4OtGbe2HR9RGZIyGLi/iJJSzi'
AWS_SESSION_TOKEN=
'FwoGZXIvYXdzEDoaDNpPecUA6vGf6Xqd8SLCAVJq7WFDncAVTjfaSYTJM5brM9W5Zo0Z4sXQUQ
xCEWmK4IoV5aTxKUjStzDqy6xq/v8cFS5liZrqYbarfPLVRhI8rHv4kNeizBB5wqeiWcKBflOKlmpHPL/
fXQraPEjsV4bmEnWvOMEUnmzwT1zr7hlDSckDLAbyKyXeJY2Y0meR1pMU1pW8D+7ouwNidU0
mXkXGlBiLcf5ref/42c10koOhYljtzCOyeGZjeEwUc3ELRQDuF/C9vofqTfoAYPOqO8ilKJCurPYFMi1Y
er1GJ/f95gyG1pR3HTrNexyVEmdHqSIbq265a4W7BcG1rtn4G58MNcEZV0w='
s3= boto3.client ('s3',
 aws_access_key_id = ACCESS_KEY_ID,
 aws_secret_access_key= ACCESS_SECRET_KEY,
 aws_session_token=AWS_SESSION_TOKEN,
 config=Config(signature_version='s3v4')
 )
s4= boto3.resource ('s3',
 aws_access_key_id = ACCESS_KEY_ID,
 aws_secret_access_key= ACCESS_SECRET_KEY,
 aws_session_token=AWS_SESSION_TOKEN,
 config=Config(signature_version='s3v4')
 )

bucket1= s4.Bucket('shivamtldbucket2')
bucket2=s4.Bucket('shivamtldbucket3')


for i in bucket1.objects.all():
   fname= i.key.split('/')[-1]
   bucket2.put_object(Key='fromshivamtldbucket2/'+ fname, Body= i.get()["Body"].read())

print("Done")

~
```