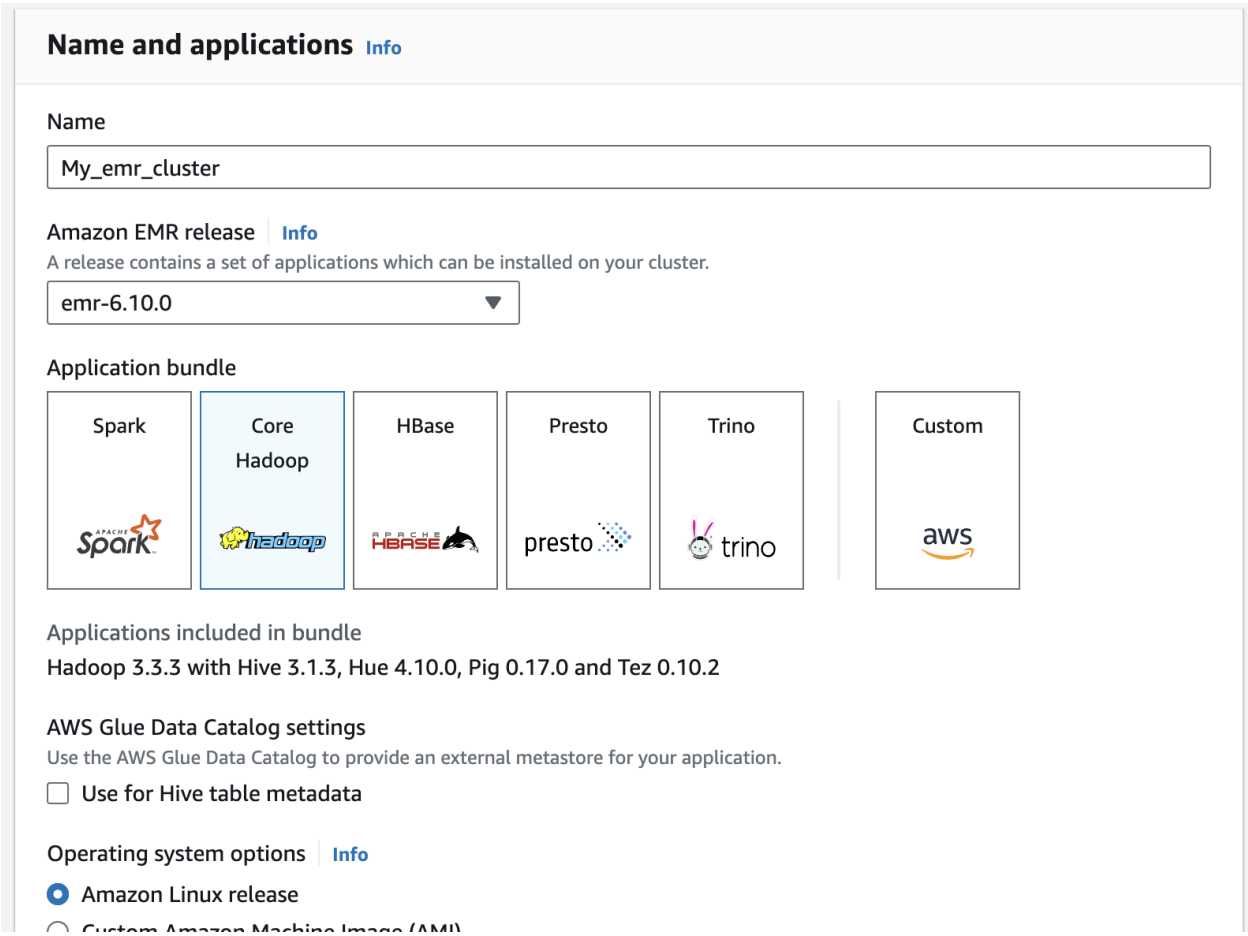
Big Data Wrangling With Google Books Ngrams

The scope of data processing and analysis is outlined in steps and screenshots for the questions below. This file showcases screenshots of steps necessary for completing the tasks: 1 to 6. Parts of step 4 are in the jupyterhub notebook with PySpark. Parts of step 6-8 are located in the second jupyter notebook on my local machine.

1. Spin up a new EMR cluster on AWS for using Spark and EMR notebooks - follow the same instructions as for the Spark Lab.



Cluster configuration Info

Choose a configuration method for the primary, core, and task node groups for your cluster.

- (●) **Instance groups**
  Choose one instance type per node group
- ( ) **Instance fleets**
  Choose any combination of instance types within each node group

Instance groups

Primary

Choose EC2 instance type

m5.xlarge
4 vCore 16 GiB memory EBS only storage
On-Demand price: $0.192 per instance/hour
Lowest Spot price: $0.053 (us-east-2b)

Actions

[ ] Use multiple primary nodes
To improve cluster availability, use 3 primary nodes with the same configuration and bootstrap actions. You can not use multiple primary nodes with instance fleets.

▶ **Node configuration** - *optional*

Core

Remove instance group

Choose EC2 instance type

m5.xlarge
4 vCore 16 GiB memory EBS only storage
On-Demand price: $0.192 per instance/hour
Lowest Spot price: $0.053 (us-east-2b)

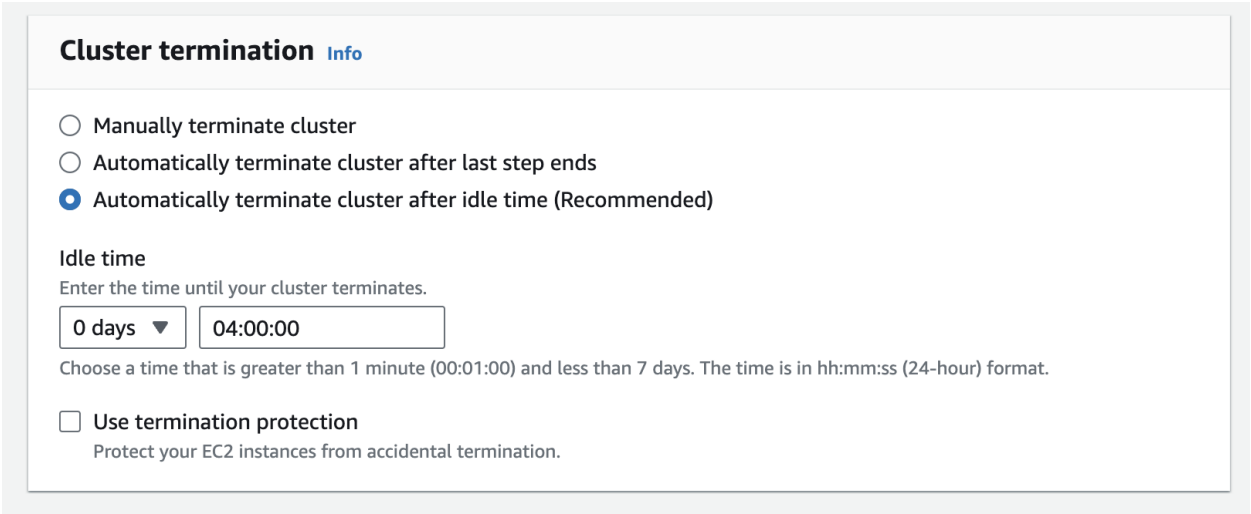
Actions

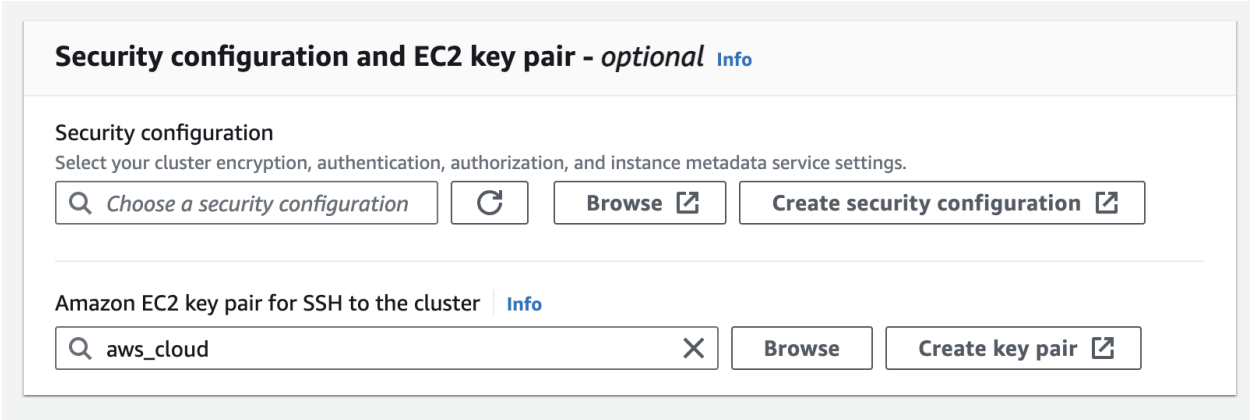
▶ **Node configuration** - *optional*

Add task instance group

You can add up to 48 more task instance groups.

▶ **EBS root volume** - *optional*





2. Connect to the head node of the cluster using SSH.
   - Use your file path for your pemkey on your local machine. To connect to the Primary Node of the cluster, and access JupyterHub in a browser window, edit the following bash command:
   - `ssh -i mykey.pem -L 9995:localhost:9443 hadoop@xxxxxxxxxxxxxx.compute.amazonaws.com`
   - Replace the 'xxxxxxxxx' with your 'Primary node public DNS' which you can find on the overview page of your cluster.
   - If the connection is successful, you should be greeted by the EMR ASCII banner:
   - Here is how I connected to my SSH cluster: ~/desktop/pemkey/aws_cloud.pem -L 9995:localhost:9443 hadoop@ec2-3-131-97-241.us-east-2.compute.amazonaws.com

Amazon EMR > EMR on EC2: Clusters > My_emr_cluster

My_emr_cluster

Updated 1 minute ago | Actions

Summary

Cluster info

Cluster ID
j-WMC80NAWA6TJ

Cluster configuration
Instance groups

Capacity
1 Primary | 2 Core | 0 Task

Applications

Amazon EMR version
emr-6.10.0

Installed applications
Hadoop 3.3.3, Hive 3.1.3, Hue 4.10.0, JupyterHub 1.5.0, Livy 0.7.1, Spark 3.3.1

Cluster management

Log destination in Amazon S3
Logging not configured

Persistent application UIs
Spark History Server
YARN timeline server
Tez UI

Primary node public DNS
ec2-3-131-97-241.us-east-2.compute.amazonaws.com
Connect to the Primary Node using SSH

Status and time

Status
Waiting

Creation time
August 22, 2023, 11:00 (UTC-04:00)

Elapsed time
1 day, 15 hours

Connect to the primary node using SSH

You can connect to the Amazon EMR primary node using SSH to perform actions like running interactive queries, examining log files, submit Linux commands, and view web interfaces hosted on Amazon EMR clusters. Learn more

Windows | Mac/Linux

1. Open a terminal window. On Mac OS X, choose Applications > Utilities > Terminal. On other Linux distributions, terminal is typically found at Applications > Accessories > Terminal.
2. To establish a connection to the primary node, enter the following command. Replace `~/aws_cloud.pem` with the location and filename of the private key file (.pem) that you used to launch the cluster.

```
ssh -i ~/aws_cloud.pem hadoop@ec2-3-131-97-241.us-east-2.compute.amazonaws.com
```

3. Enter yes to dismiss the security warning.

View web interfaces hosted on Amazon EMR clusters

Close

```
Last login: Tue Aug 22 11:04:43 on ttys004
(base) markbenhaim@Marks-MBP ~ % ssh -i ~/desktop/pemkey/aws_cloud.pem -L 9995:localhost:9443 hadoop@ec2-3-131-97-241.us-east-2.compute.amazonaws.com
Last login: Tue Aug 22 15:20:43 2023

       __|  __|_  )
       _|  (     /   Amazon Linux 2 AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-2/
14 package(s) needed for security, out of 16 available
Run "sudo yum update" to apply all updates.

EEEEEEEEEEEEEEEEEEEE MMMMMMMM           MMMMMMMM RRRRRRRRRRRRRRR
E::::::::::::::::::E M::::::::         M::::::::M R::::::::::::::R
EE:::::EEEEEEEEE:::E M::::::::M       M::::::::M R:::::RRRRRR:::::R
  E::::E       EEEEE M:::::::::M     M:::::::::M RR:::R      R:::R
  E::::E             M::::::M:::M   M:::M::::::M   R:::R      R:::R
  E:::::EEEEEEEEEE   M:::::M M::::M M::::M M:::::M   R:::RRRRRR:::::R
  E::::::::::::::E   M:::::M  M:::M:::M  M:::::M   R::::::::::::RR
  E:::::EEEEEEEEEE   M:::::M   M:::::M   M:::::M   R:::RRRRRR:::R
  E::::E             M:::::M    M:::M    M:::::M   R:::R      R:::R
  E::::E       EEEEE M:::::M     MMM     M:::::M   R:::R      R:::R
EE:::::EEEEEEEE::::E M:::::M             M:::::M   R:::R      R:::R
E::::::::::::::::::E M:::::M             M:::::M RR:::R      R:::R
EEEEEEEEEEEEEEEEEEEE MMMMMMM             MMMMMMM RRRRRRR      RRRRRR

[hadoop@ip-172-31-28-187 ~]$ hadoop distcp s3://brainstation-dsft/eng_1M_1gram.csv /user/hadoop/eng_1M_1gram/
```

3. Copy the data folder from the S3 bucket *directly* into a directory on the Hadoop File System (HDFS) named `/user/hadoop/eng_1M_1gram`.

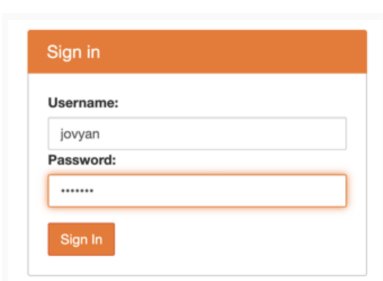
- Connect to the Brainstation public S3 bucket to access the dataset within the data folder by typing this code: hadoop distcp s3://brainstation-dsft/eng_1M_1gram.csv /user/hadoop/eng_1M_1gram/
-

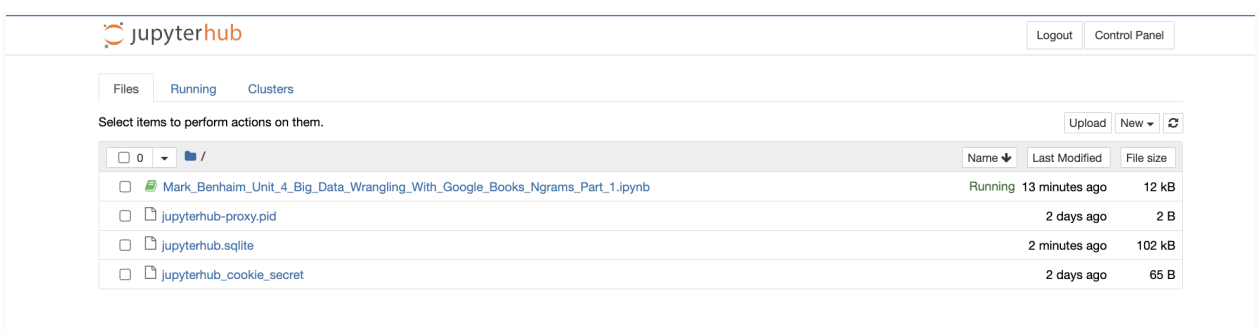
```
EEEEEEEEEEEEEEEEEEEE MMMMMMMM           MMMMMMMM RRRRRRRRRRRRRRR
E::::::::::::::::::E M:::::::M         M:::::::M R::::::::::::::R
EE:::::EEEEEEEEE:::E M::::::::M       M::::::::M R:::::RRRRRR:::::R
  E::::E       EEEEE M:::::::::M     M:::::::::M RR::::R      R::::R
  E::::E             M::::::M:::M   M:::M::::::M   R:::R      R::::R
  E::::::::::::::::  M:::::M M:::M M:::M M:::::M   R:::RRRRRR:::::R
  E::::::::::::::::  M:::::M  M:::M:::M  M:::::M   R::::::::::::RR
  E::::::::::::::::  M:::::M   M:::::M   M:::::M   R:::RRRRRR::::R
  E::::E             M:::::M    M:::M    M:::::M   R:::R      R::::R
  E::::E       EEEEE M:::::M     MMM     M:::::M   R:::R      R::::R
EE:::::EEEEEEEE:::E M:::::M             M:::::M   R:::R      R::::R
E::::::::::::::::::E M:::::M             M:::::M RR::::R      R::::R
EEEEEEEEEEEEEEEEEEEE MMMMMMM             MMMMMMM RRRRRRR      RRRRRR

[hadoop@ip-172-31-28-187 ~]$ hadoop distcp s3://brainstation-dsft/eng_1M_1gram.csv /user/hadoop/eng_1M_1gram/
```

4. Using pyspark, read the data you copied into HDFS in Step 3. You may either use Jupyterhub on EMR (the default user and password are `jovyan` and `jupyter`) or work from pyspark in the terminal if you prefer.

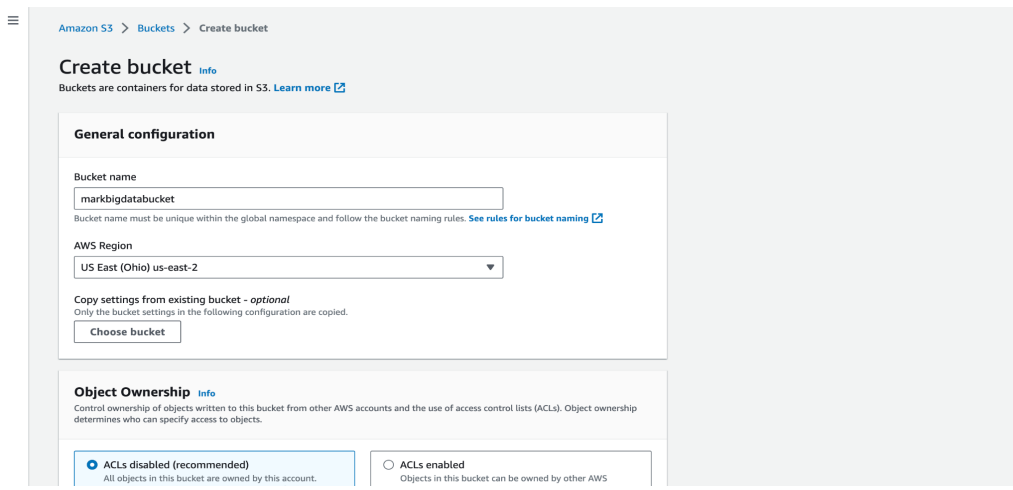
- Use this link to access jupyterhub in yoru browser: https://localhost:9995/
- Make sure your cluster is in the waiting stage and rerun the ssh command
- Login to Jupyterhub using the login info then get to this page and create a notebook

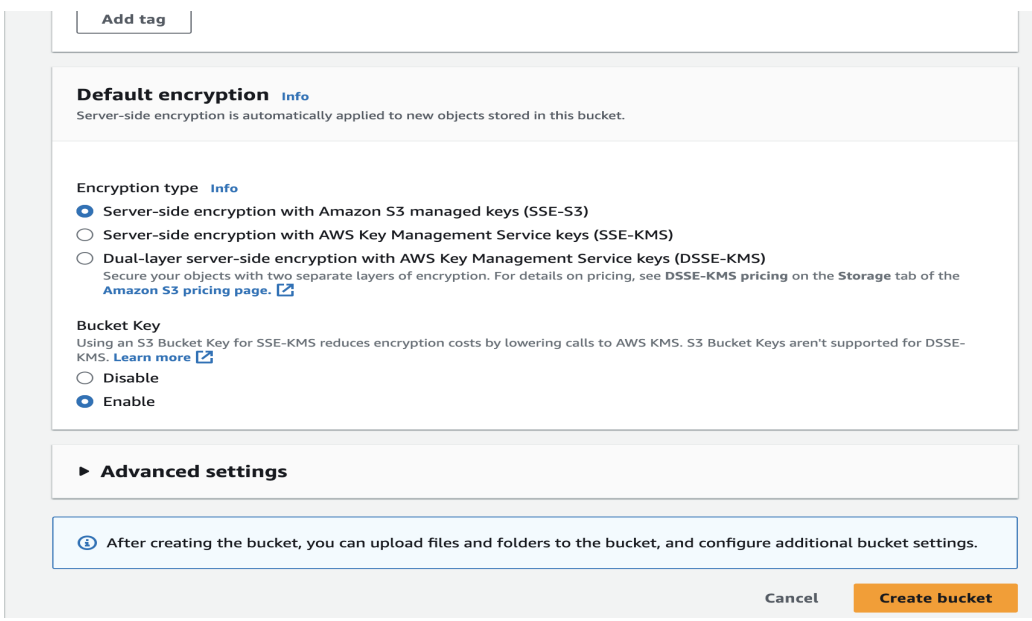




5. Collect the contents of the directory into a single file on the local drive of the head node using `getmerge` and move this file into a S3 bucket in your account.
    - Input code in bash:
    - Note: filtered_1gram is the name of my file. Use your file name that you gave from question 4.
    - You'll also need to create your own s3 bucket in your AWS account under the S3 page.
    - Use the same region as your cluster. Name your bucket
    - Create the bucket

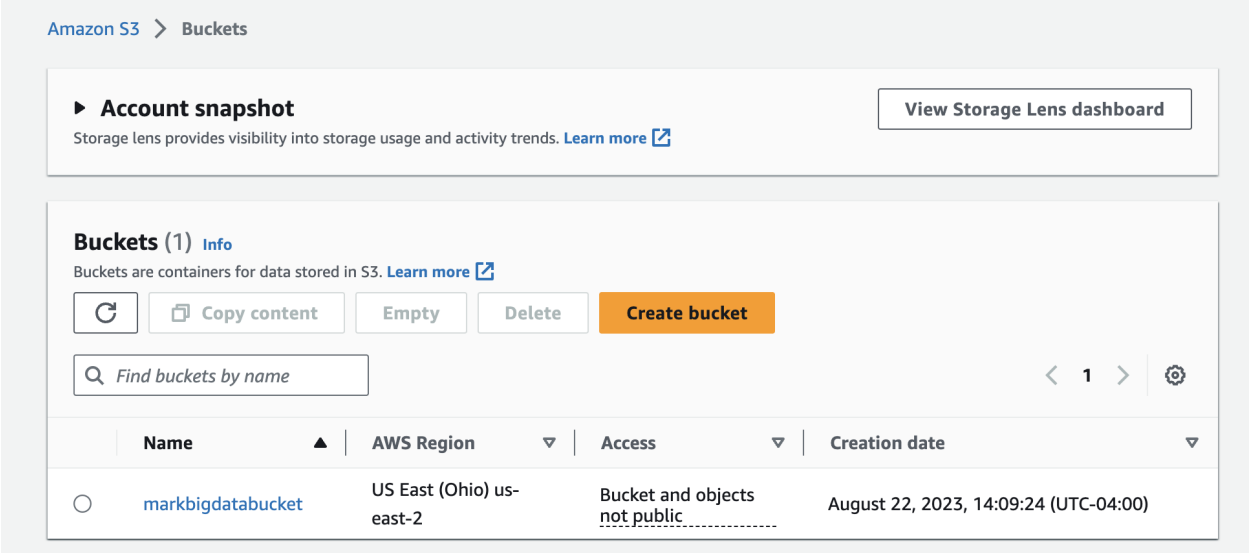
```
[hadoop@ip-172-31-28-187 ~]$ hadoop fs -getmerge /user/hadoop/filtered_1gram/ ~/filtered_1gram.csv
```

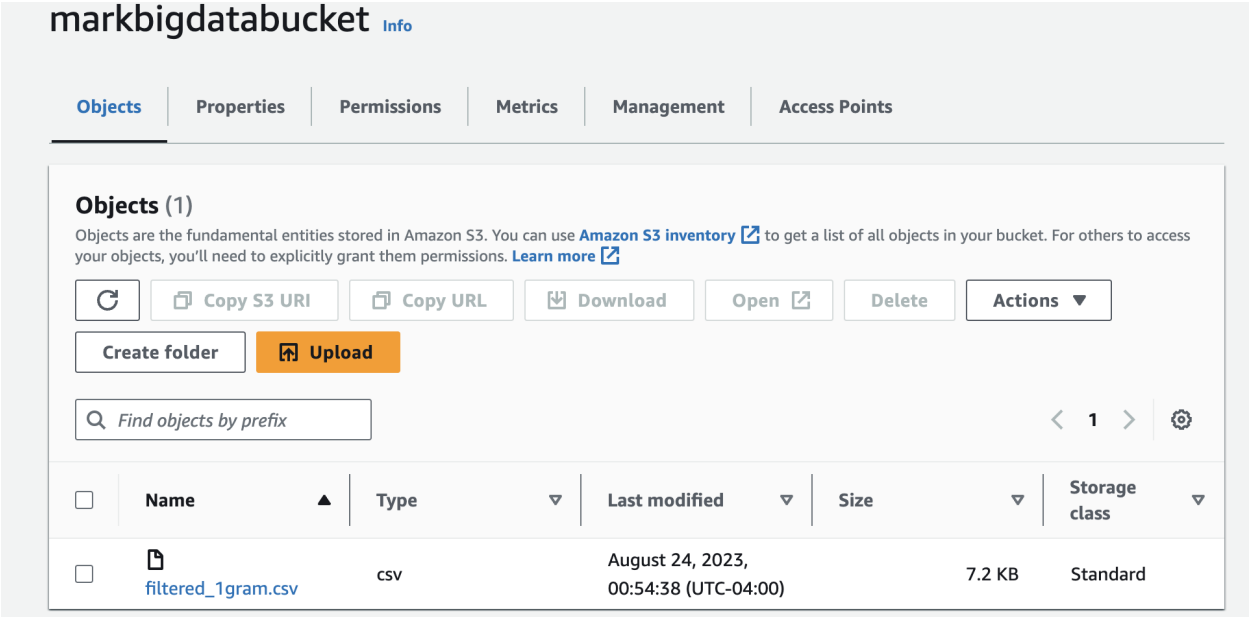


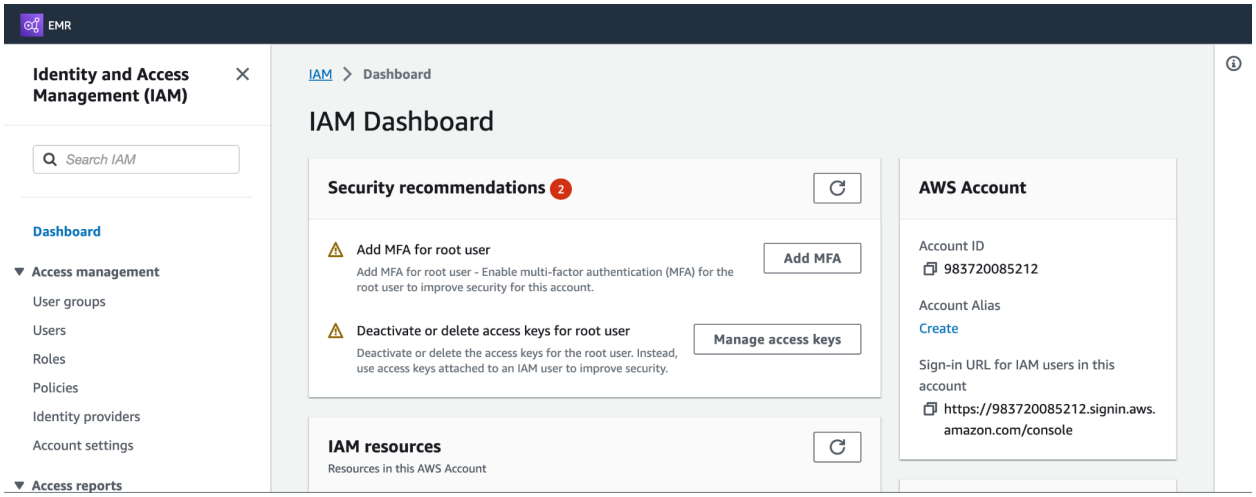


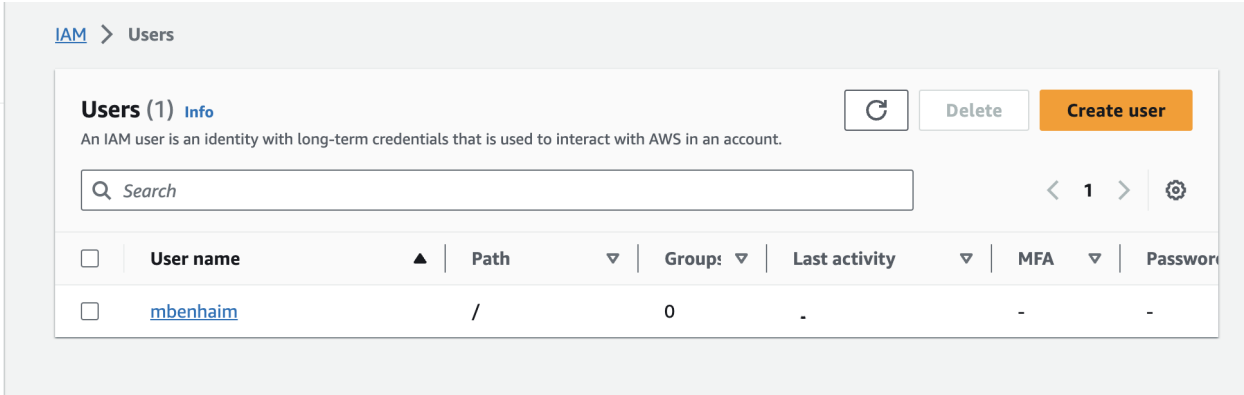
6. - Use ‘aws configure’ in bash and place your codes and default region:
   - First, get your codes from your aws account
   - Check your s3 bucket to see if your file is there by clicking on the name of your bucket
   - The file should be shown under objects
   - Go to your IAM dashboard and create a user and name it
   - Afterwards, go to the drop down add permissions and click create inline policy
   - Click S3 and all S3 actions then click next then create policy
   - Go to Security and Recommendations on your IAM dashboard and click manage access keys
   - Under access keys, click create access key then check I understand and click create access key
   - Download your access key and secret key
   - Now you can use ‘aws configure’ in bash and place your codes and default region:

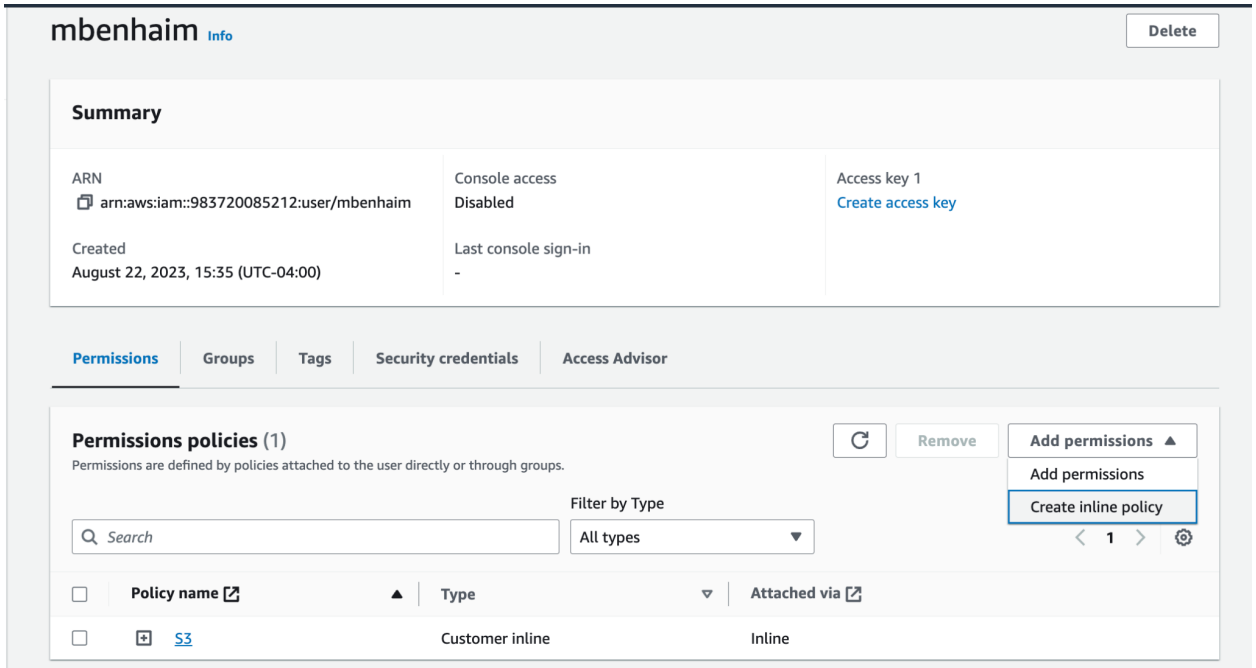
```
[[hadoop@ip-172-31-28-187 ~]$ aws configure
 AWS Access Key ID [****************LTUB]: AKIA6KCSJI3OLJHELTUB
 AWS Secret Access Key [****************hcVi]: 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
 Default region name [us-east-2]: us-east-2
 Default output format [None]:
```




markbigdatabucket Info
Objects
Properties
Permissions
Metrics
Management
Access Points
Objects (1)
Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more
Copy S3 URI
Copy URL
Download
Open
Delete
Actions
Create folder
Upload
Find objects by prefix
1
Name
Type
Last modified
Size
Storage class
filtered_1gram.csv
csv
August 24, 2023, 00:54:38 (UTC-04:00)
7.2 KB
Standard


EMR
Identity and Access Management (IAM)
Search IAM
Dashboard
Access management
User groups
Users
Roles
Policies
Identity providers
Account settings
Access reports
IAM > Dashboard
IAM Dashboard
Security recommendations 2
Add MFA for root user
Add MFA for root user - Enable multi-factor authentication (MFA) for the root user to improve security for this account.
Add MFA
Deactivate or delete access keys for root user
Deactivate or delete the access keys for the root user. Instead, use access keys attached to an IAM user to improve security.
Manage access keys
IAM resources
Resources in this AWS Account
AWS Account
Account ID
983720085212
Account Alias
Create
Sign-in URL for IAM users in this account
https://983720085212.signin.aws.amazon.com/console


IAM > Users
Users (1) Info
An IAM user is an identity with long-term credentials that is used to interact with AWS in an account.
Delete
Create user
Search
1
User name
Path
Groups
Last activity
MFA
Passwor
mbenhaim
/
0
-
-
-


mbenhaim Info
Delete
Summary
ARN
arn:aws:iam::983720085212:user/mbenhaim
Console access
Disabled
Access key 1
Create access key
Created
August 22, 2023, 15:35 (UTC-04:00)
Last console sign-in
-
Permissions
Groups
Tags
Security credentials
Access Advisor
Permissions policies (1)
Permissions are defined by policies attached to the user directly or through groups.
Remove
Add permissions
Add permissions
Create inline policy
Filter by Type
Search
All types
1
Policy name
Type
Attached via
S3
Customer inline
Inline

IAM > Users > mbenhaim > Create policy

Step 1
Specify permissions

Step 2
Review and create

Specify permissions Info

Add permissions by selecting services, actions, resources, and conditions. Build permission statements using the JSON editor.

Policy editor

Visual | JSON | Actions

S3

Allow 1 Actions

Specify what actions can be performed on specific resources in S3.

Actions allowed

Specify actions from the service to be allowed.

Filter Actions

Switch to deny permissions

Manual actions | Add actions

All S3 actions (s3:*)

Access level

Expand all | Collapse all

- List (Selected 10/10)
- Read (Selected 53/53)
- Write (Selected 42/42)
- Permissions management (Selected 15/15)
- Tagging (Selected 10/10)

Required permissions not selected.
To grant permissions for the selected resource actions, you must include additional required actions

- s3:CreateJob requires 1 more actions.
- s3:PutReplicationConfiguration requires 1 more actions.

Resources

Specify resource ARNs for these actions.

Specific | All

The all wildcard '*' may be overly permissive for the selected actions. Allowing specific ARNs for these service resources can improve security.

Request conditions - *optional*

Actions on resources are allowed or denied only when these conditions are met.

Add more permissions

Security: 0 Errors: 0 Warnings: 0 Suggestions: 0

Cancel | Next

IAM > Users > mbenhaim > Create policy

Step 1
Specify permissions

Step 2
Review and create

Review and create

Review the permissions, specify details, and tags.

Policy details

Policy name
Enter a meaningful name to identify this policy.

s3

Maximum 128 characters. Use alphanumeric and '+=,.@-_' characters.

Permissions defined in this policy Info

Permissions in the policy document specify which actions are allowed or denied.

Edit

Search

Allow (1 of 385 services)

Show remaining 384 services

Service	Access level	Resource	Request condition
S3	Full access	All resources	None

Cancel | Previous | Create policy

Security recommendations 2

Add MFA for root user

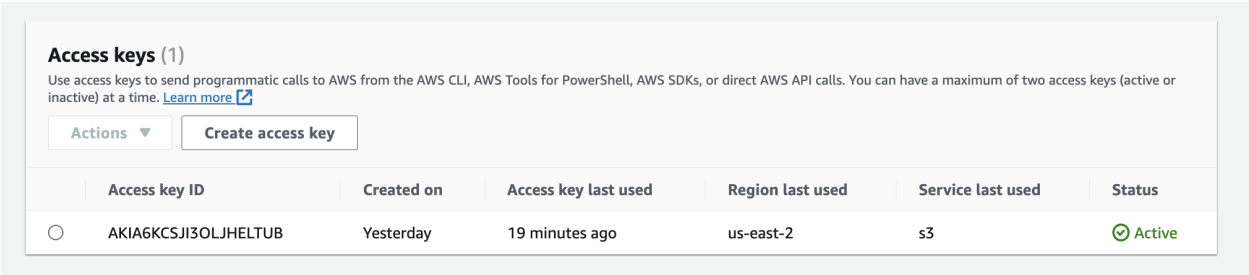
Add MFA for root user - Enable multi-factor authentication (MFA) for the root user to improve security for this account.

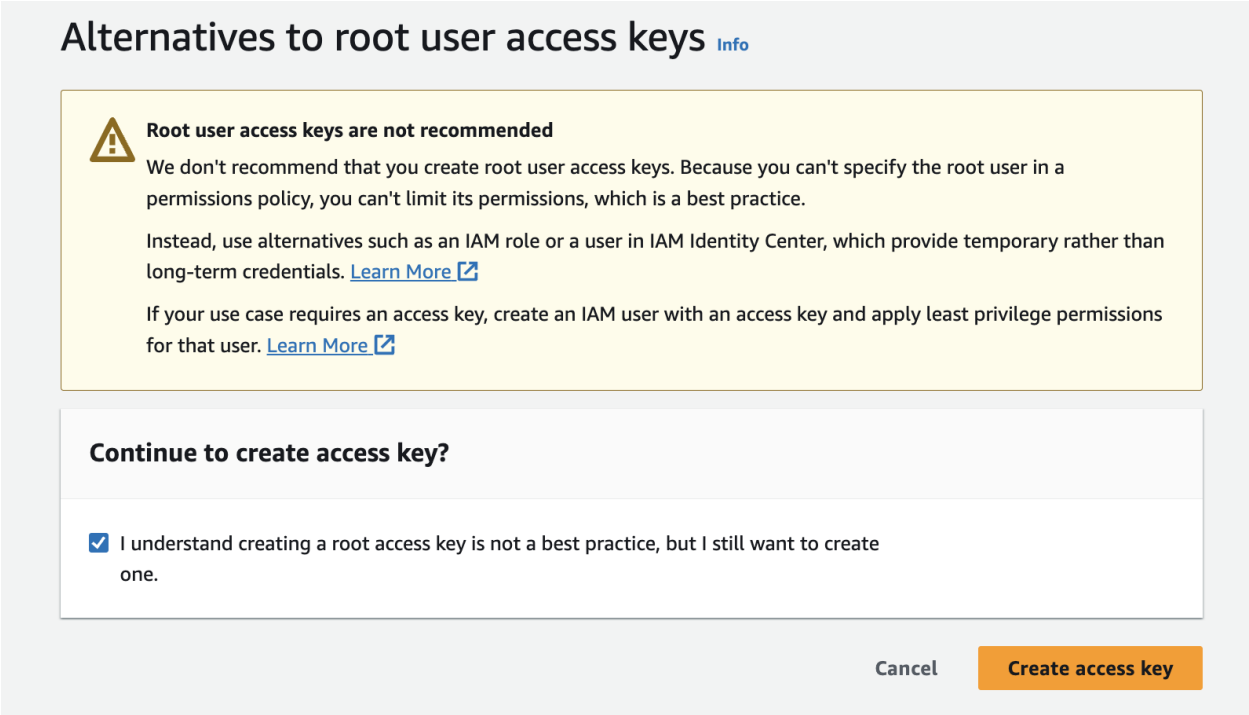
Add MFA

Deactivate or delete access keys for root user

Deactivate or delete the access keys for the root user. Instead, use access keys attached to an IAM user to improve security.

Manage access keys





Note: The answers corresponding to questions 1-8 are also located in within the 2 notebooks:

- Mark_Benhaim_Unit_4_Big_Data_Wrangling_With_Google_Books_Ngrams_Part_1
- Mark_Benhaim_Unit_4_Big_Data_Wrangling_With_Google_Books_Ngrams_Part_2