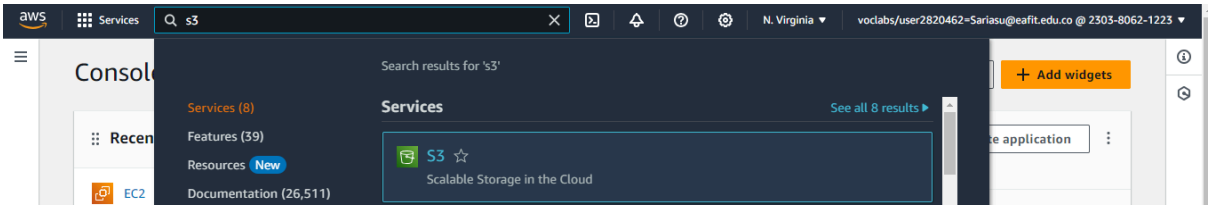
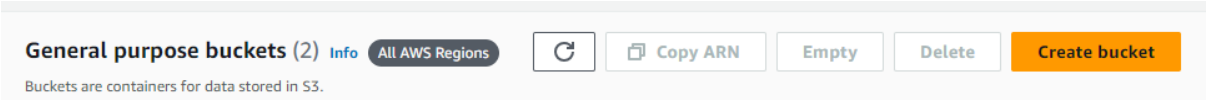
# Sebastián Arias Usma.
# C.C 1017932811
# Evidencias laboratorios 3 big data
# st0263-241 Tópicos de telematica

**Reto # 5: Laboratorio EMR y Laboratorio de archivos por HDFS**

**Paso #1:** Debemos crear un bucket S3 en amazon por lo que debemos buscar en el aws academy este espacio de la siguiente manera.

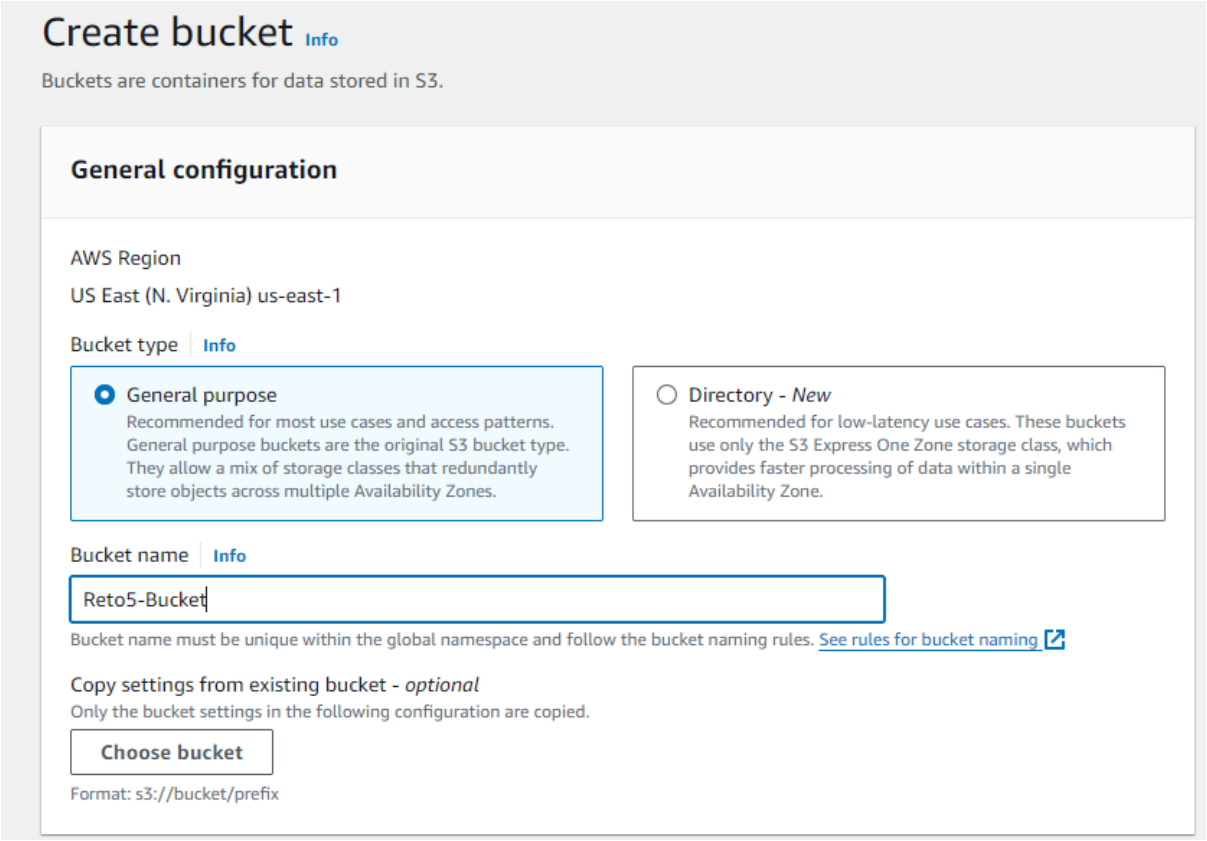



**Paso #2:** Debemos crear nuestro bucket de s3 para el reto #5 el cual luego conectaremos con nuestro EMR, seleccionamos el botón de Create bucket.

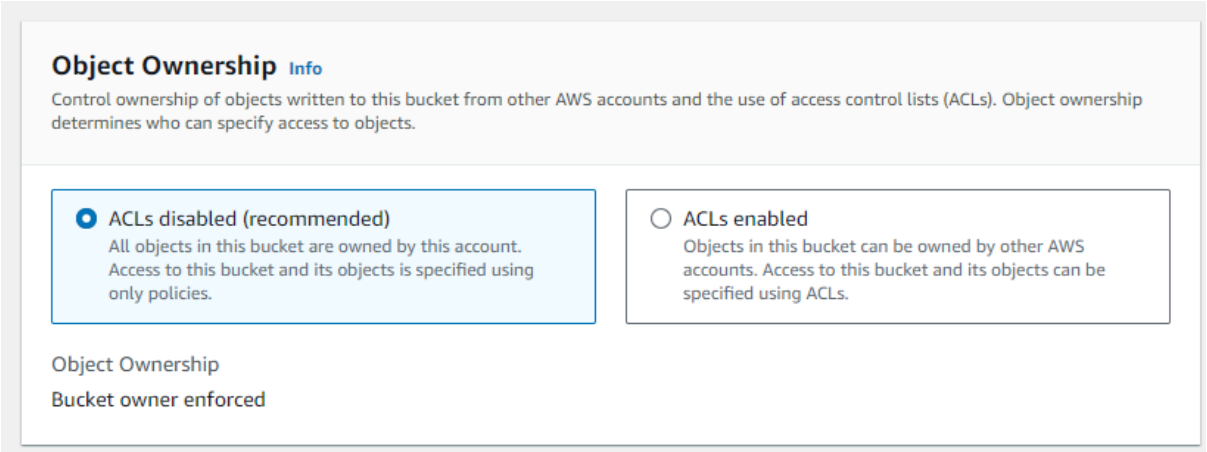



**Paso#3:** Debemos configurar nuestro bucket de la siguiente manera.

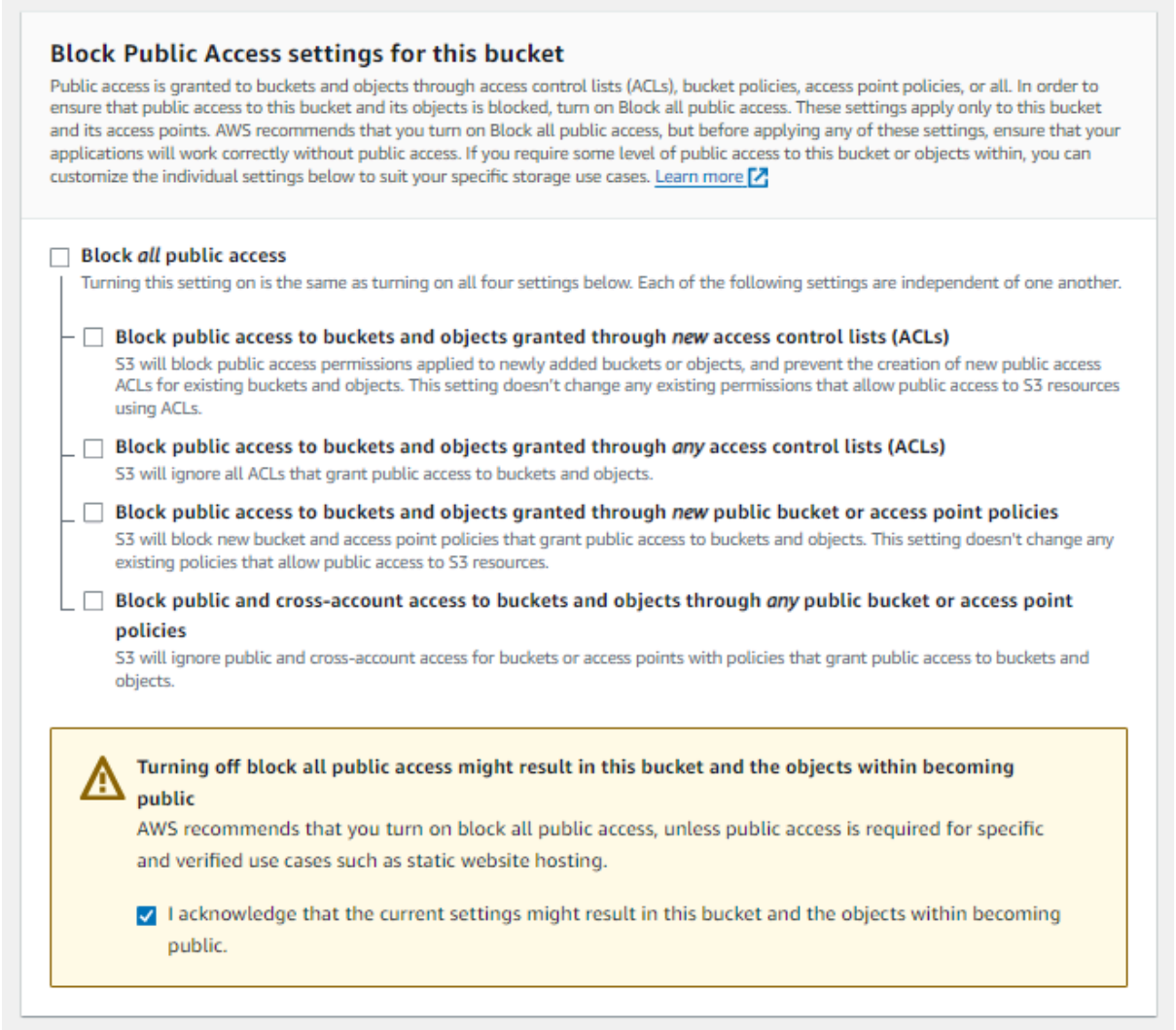
- Seleccionamos el tipo de bucket



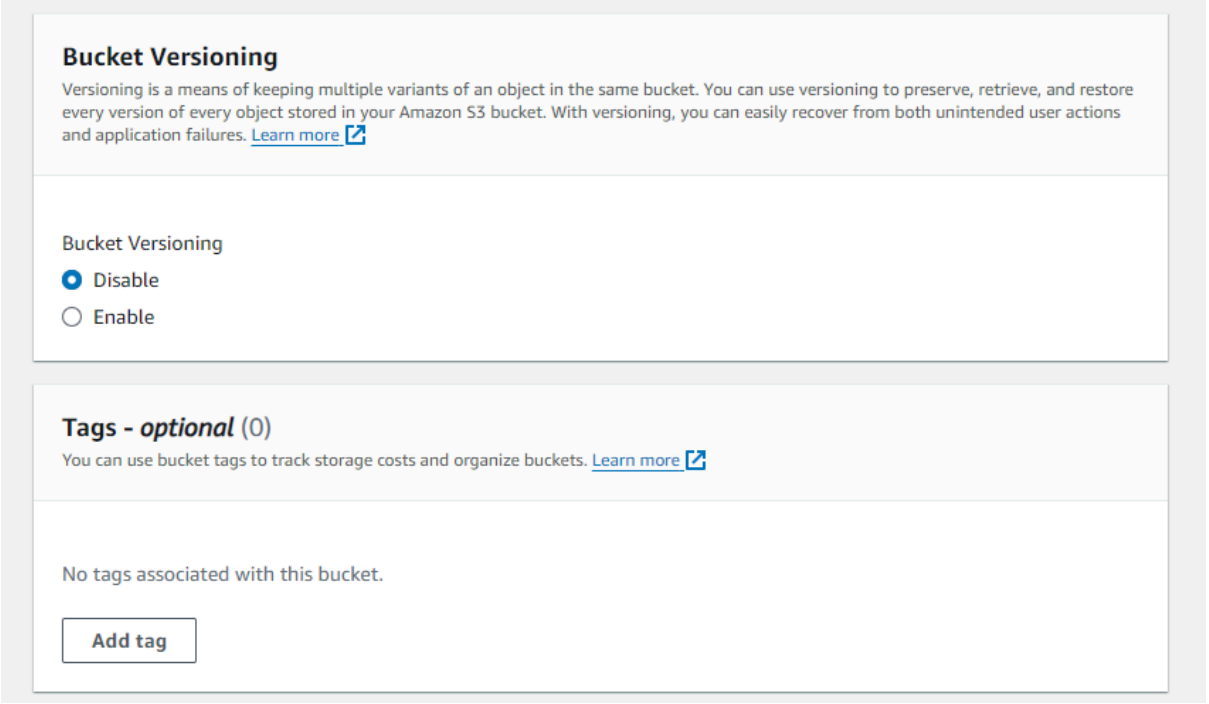
- Seleccionamos la manera en como se va a comportar el bucket.

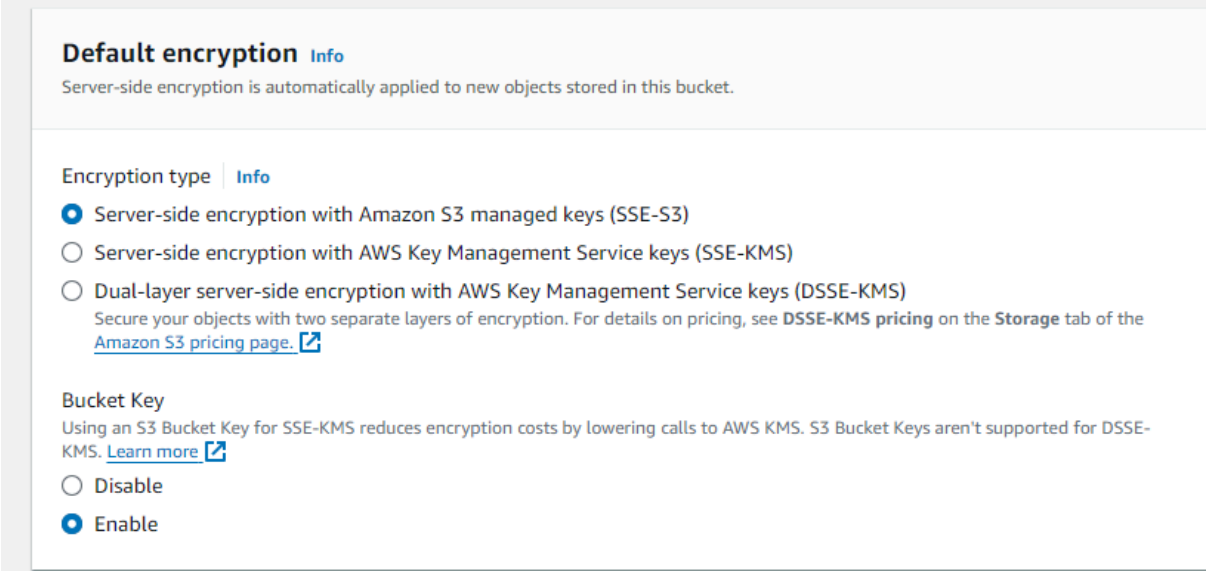



- Permitimos que sea de acceso publico.

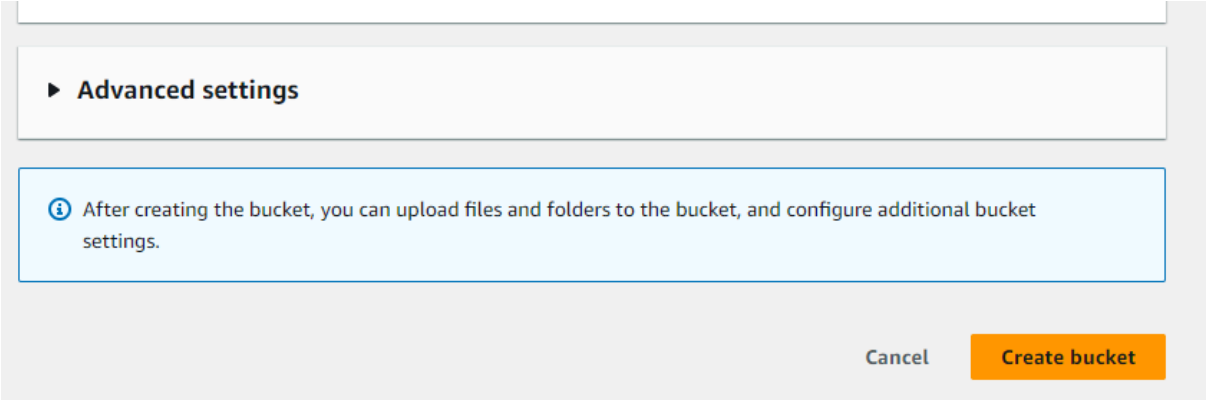


- Dejamos estas configuraciones por defecto.



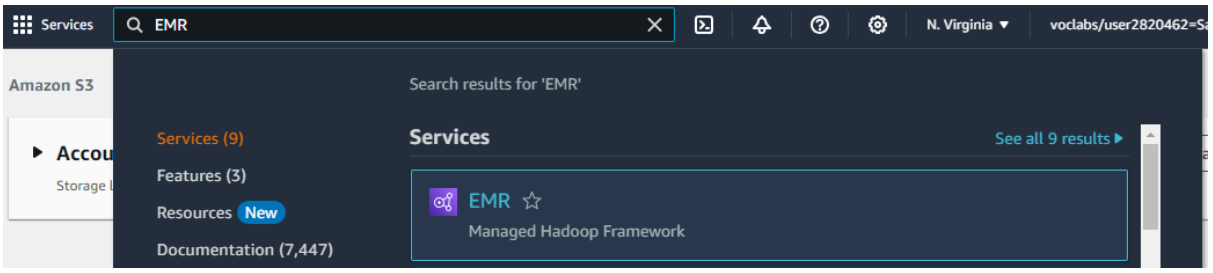




- Y ahora creamos el nuestro s3 buket.

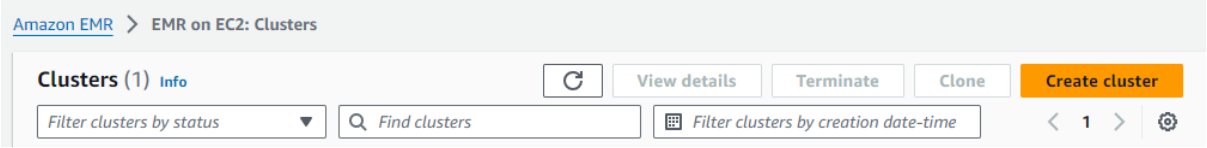



**Paso #4:** Ahora debemos ir al apartado de EMR en nuestro AWS y crear uno de la siguiente manera.

- Accedemos al apartado de EMR de la siguiente manera.




- Ahora vamos a darle click al botón de crear cluster.




- Crearemos el cluster con la siguiente configuración.
- Seleccionamos que version de EMR deseamos, en este caso la 6.15.0 y las aplicaciones que necesitaremos en el EMR

Name

EMR-reto5

Amazon EMR release | Info

A release contains a set of applications which can be installed on your cluster.

emr-6.15.0

Application bundle

Spark Interactive | Core Hadoop | Flink | HBase | Presto | Trino | Custom

- [x] Flink 1.17.1
- [x] HCatalog 3.1.3
- [x] Hue 4.11.0
- [x] Livy 0.7.1
- [ ] Phoenix 5.1.3
- [x] Spark 3.4.1
- [x] Tez 0.10.2
- [x] ZooKeeper 3.5.10
- [ ] Ganglia 3.7.2
- [x] Hadoop 3.3.6
- [x] JupyterEnterpriseGateway 2.6.0
- [ ] MXNet 1.9.1
- [ ] Pig 0.17.0
- [x] Sqoop 1.4.7
- [ ] Trino 426
- [ ] HBase 2.4.17
- [x] Hive 3.1.3
- [x] JupyterHub 1.5.0
- [ ] Oozie 5.2.1
- [ ] Presto 0.283
- [ ] TensorFlow 2.11.0
- [x] Zeppelin 0.10.1

AWS Glue Data Catalog settings

Use the AWS Glue Data Catalog to provide an external metastore for your application.

- [x] Use for Hive table metadata
- [x] Use for Spark table metadata

Operating system options | Info

- (•) Amazon Linux release
- ( ) Custom Amazon Machine Image (AMI)

- [x] Automatically apply latest Amazon Linux updates

- Ahora configuraremos los atributos del cluster EMR de la siguiente manera

## Cluster configuration - *required* Info

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

(•) **Uniform instance groups**
Choose the same EC2 instance type and purchasing option (On-Demand or Spot) for all nodes in your node group. Learn more

( ) **Flexible instance fleets**
Choose from the widest variety of provisioning options for the EC2 instances in your cluster. Diversify instance types and purchasing options, and use an allocation strategy. Learn more

### Uniform instance groups

**Primary**

Choose EC2 instance type

m5.xlarge
4 vCore 16 GiB memory EBS only storage
On-Demand price: - Lowest Spot price: -

Actions

- [ ] Use high availability
Launch highly available, more resilient cluster with three primary nodes on On-Demand Instances. This configuration applies for the lifetime of your cluster. Learn more

▶ **Node configuration - *optional***

**Core**

Choose EC2 instance type

m5.xlarge
4 vCore 16 GiB memory EBS only storage
On-Demand price: - Lowest Spot price: -

Actions

▶ **Node configuration - *optional***

**Task 1 of 1**

Remove instance group

Name

Task - 1

Choose EC2 instance type

m5.xlarge
4 vCore 16 GiB memory EBS only storage
On-Demand price: - Lowest Spot price: -

Actions

▶ **Node configuration - *optional***

Add task instance group

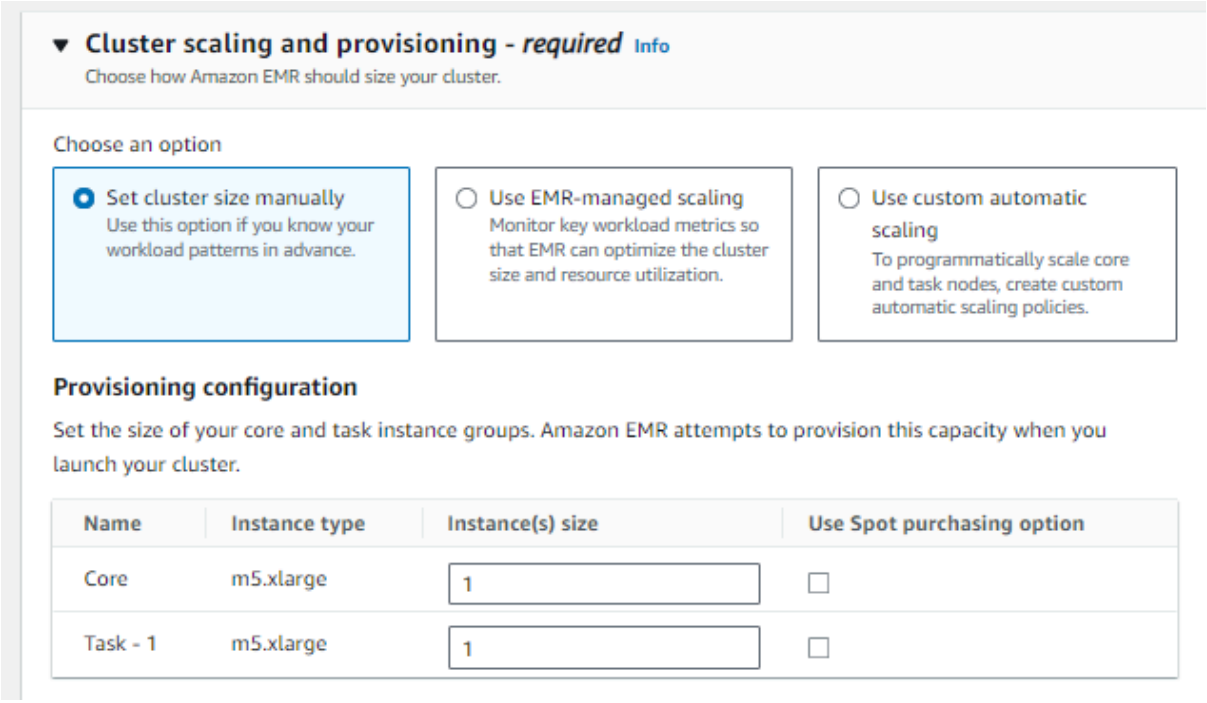
You can add up to 47 more task instance groups.

**EBS root volume**

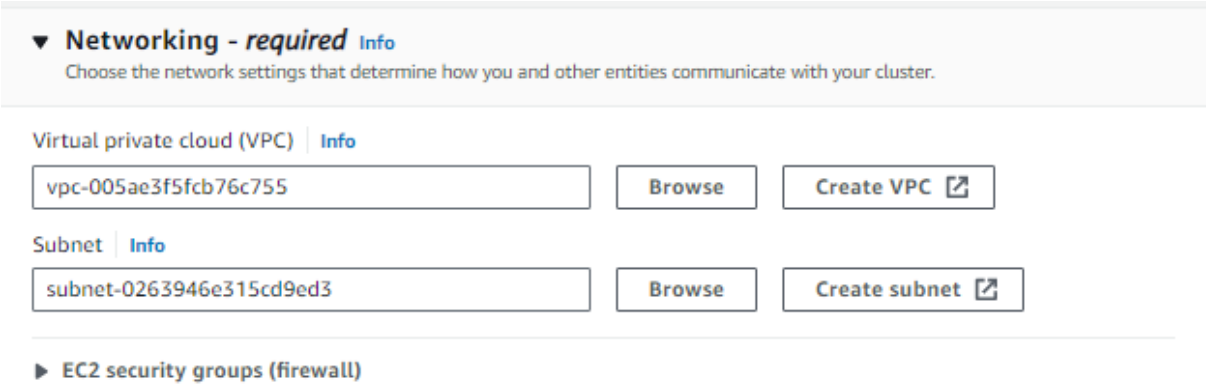
EBS root volume applies to the operating systems and applications that you install on the cluster. EBS root volume ratio constraints

| Size (GiB) | IOPS | Throughput (MiB/s) |
|---|---|---|
| 15 | 3000 | 125 |
| 15 - 100 GiB per volume General Purpose SSD (gp3) | 3000 - 16000 IOPS per volume. Choose a maximum ratio of 500:1 between IOPS and volume size. | 125 - 1000 MiB/s per volume. Choose a maximum ratio of 0.25:1 between throughput and IOPS. |

- Ahora ajustaremos la escalabilidad de nuestro cluster EMR de la siguente manera.




- Ahora configuaremos los volumenes de redes o las VP, VPC que necesitemos




- Determinaremos la manera en la que dejara de trabajar nuestro cluster que sera de manera manual.

## Cluster termination and node replacement Info

Choose termination settings and protect your cluster from accidental shutdown.

Termination option

- (●) Manually terminate cluster
- ( ) Automatically terminate cluster after last step ends
- ( ) Automatically terminate cluster after idle time (Recommended)

[ ] Use termination protection

Protects your cluster from accidental termination. If on, you must first turn off protection to terminate the cluster. We recommend turning on termination protection for your long running clusters.

Unhealthy node replacement - *new* Info

- (●) Turn on

  Amazon EMR gracefully stops processes on unhealthy nodes to minimize data loss and job interruptions. It quickly replaces unhealthy nodes with new EC2 instances to keep your jobs running smoothly.

- ( ) Turn off

  Amazon EMR adds unhealthy nodes to a denylist while keeping them in the cluster, allowing you continued access for troubleshooting.

- Luego debemos configurar el EMS en nuestro aws con este código en el apartado de configurar Software Settings para conectarlo con nuestro bucket s3.

## Software settings Info

Override the default configurations for specific applications on your cluster.

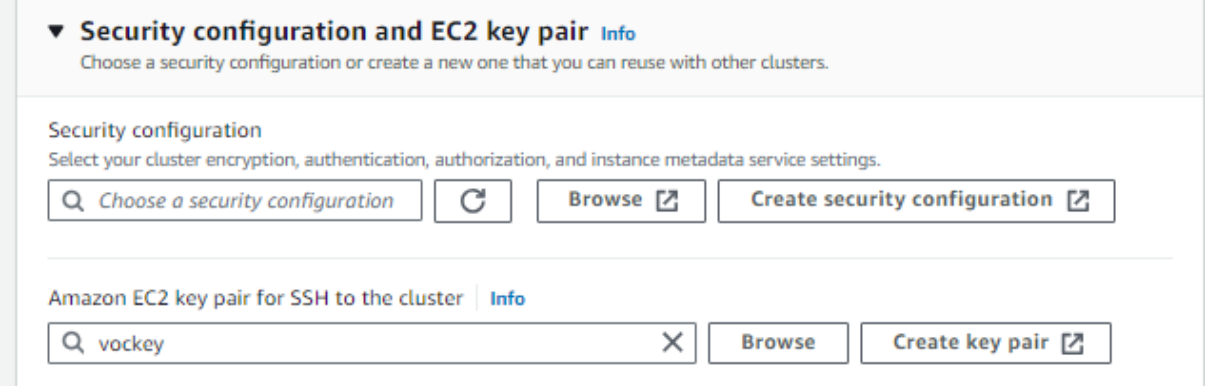
(●) Enter configuration | ( ) Load JSON from Amazon S3

```
[
  {
    "Classification": "jupyter-s3-conf",
    "Properties": {
      "s3.persistence.bucket": "reto5-bucket",
      "s3.persistence.enabled": "true"
    }
  }
]
```

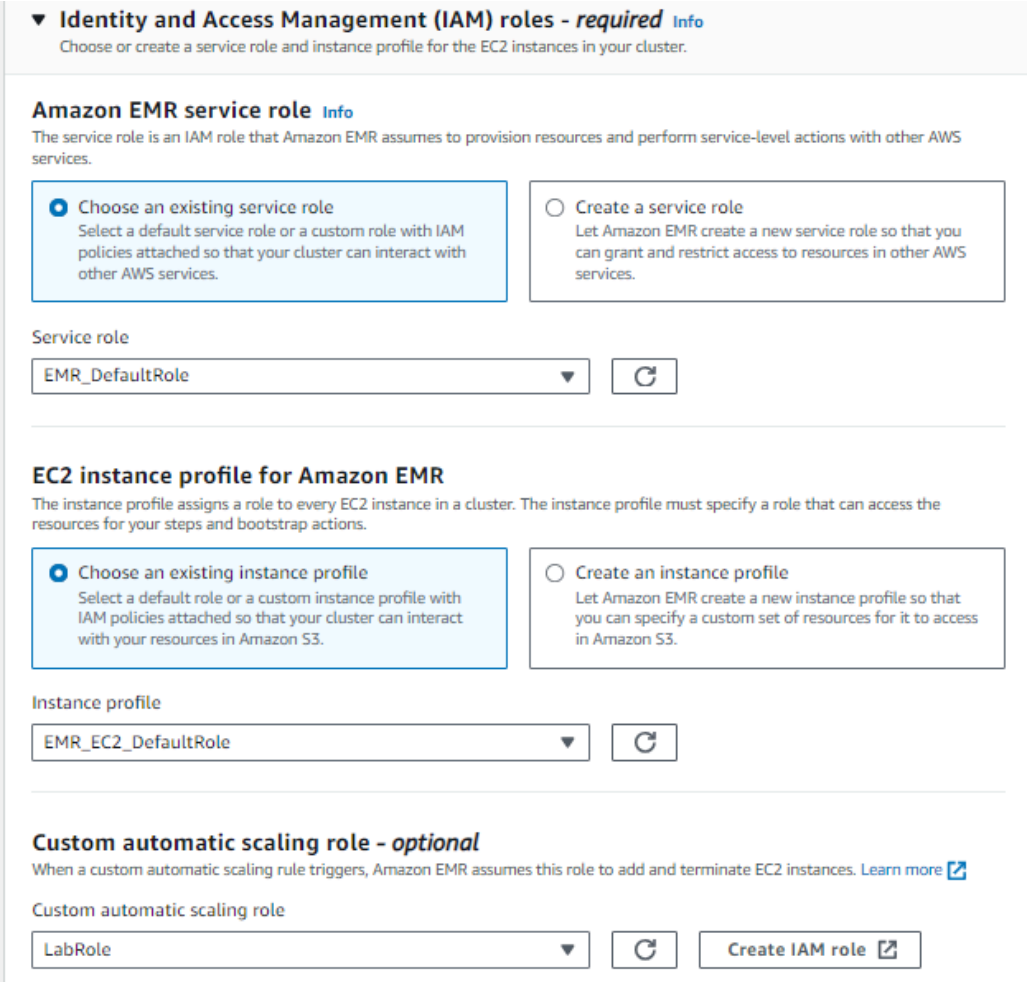
JSON Ln 1, Col 1

```
[
  {
    "Classification": "jupyter-s3-conf",
    "Properties": {
      "s3.persistence.enabled": "true",
      "s3.persistence.bucket": "--nombre de mi s3 bucket--  "
    }
  }
]
```

- Ahora configuraremos nuestro grupo de seguridad.




- Y por ultimo dejamos que nuestro cluster cree las instancias necesarias para acceder a nuestro EMR.




- Con estos pasos ya hemos creado nuestro EMR cluster.

Luego modificamos el security group de nuestro EMR desde el E2C para obtener las Ips públicas de nuestro HUE, debemos agregar estos security groups.

- 8443
- 9870
- 8888
- 14000
- 9443
- 8890

Todas estas con ipv4

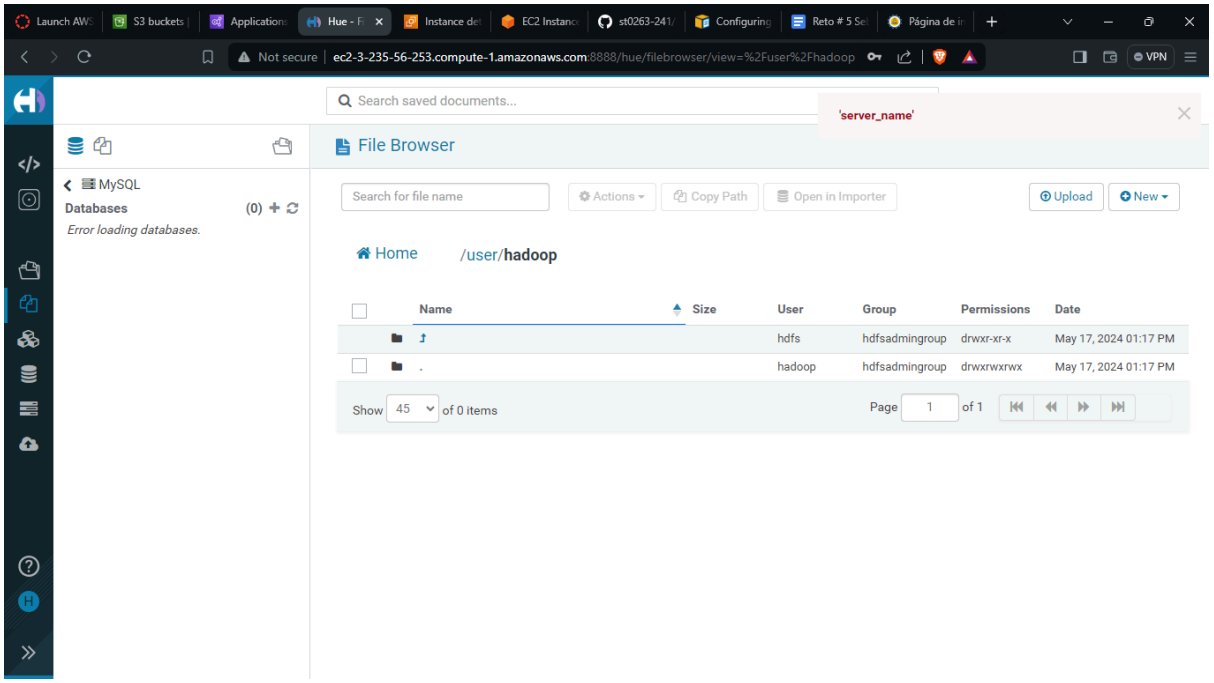
**Paso # 5:**

- Ahora accederemos a nuestro EMR desde nuestro SSH y haremos los siguientes comandos para poder acceder a nuestro HUE y subir nuestros archivos

- Cada vez que clonemos el cluster debemos copiar estos comandos en nuestro bash de nuestro EMR

```
[root@ip-172-31-73-11 ~]# sed -i 's/.ec2.internal:14000/.ec2.internal:9870/' /etc/hue/conf/hue.ini
[root@ip-172-31-73-11 ~]# systemctl restart hue.service
```

Ahora revisamos nuestra página de hue y debería salir de esta manera:




**Paso #6:** Luego hacemos yum update para instalar git y luego de este realizamos el yum install git, para clonar el repositorio del reto.

- git clone https://github.com/st0263eafit/st0263-241.git

Y cuando tenemos clonado el repo realizamos los comandos hdfs para el hadoop

- hdfs dfs -ls /
- hdfs dfs -ls /user
- hdfs dfs -ls /user/hadoop
- hdfs dfs -ls /user/hadoop/datasets

al ejecutar el ultimo comando saldra error ya que no tenemos la carpeta datasets por lo que debemos crearla con el siguiente comando

- hdfs dfs -mkdir /user/hadoop/datasets

Cuando creamos esta carpeta crearemos la primera carpeta de datasets la cual se llamara gutenberg-small.

- hdfs dfs -mkdir /user/hadoop/datasets/gutenberg-small

Cuando creamos esta carpeta procederemos a insertar los primeros archivos del dataset con este comando.

- hdfs dfs -put /home/ec2-home/st0263-241/bigdata/datasets/gutenberg-small/*.txt /user/hadoop/datasets/gutenberg-small/

y tendremos la carpeta creada en nuestro Hue

Home /user/hadoop/**datasets**

| Name | Size | User | Group | Permissions | Date |
|---|---|---|---|---|---|
| ↑ | | hadoop | hdfsadmingroup | drwxrwxrwx | May 17, 2024 01:55 PM |
| . | | root | hdfsadmingroup | drwxr-xr-x | May 17, 2024 01:58 PM |
| airlines.csv | 761.8 KB | hadoop | hdfsadmingroup | -rw-r--r-- | May 17, 2024 01:58 PM |
| all-news | | hadoop | hdfsadmingroup | drwxr-xr-x | May 17, 2024 01:57 PM |
| covid19 | | hadoop | hdfsadmingroup | drwxr-xr-x | May 17, 2024 01:58 PM |
| flights | | hadoop | hdfsadmingroup | drwxr-xr-x | May 17, 2024 01:58 PM |
| gutenberg | | hadoop | hdfsadmingroup | drwxr-xr-x | May 17, 2024 01:58 PM |
| gutenberg-small | | root | hdfsadmingroup | drwxr-xr-x | May 17, 2024 01:58 PM |
| onu | | hadoop | hdfsadmingroup | drwxr-xr-x | May 17, 2024 01:58 PM |
| otros | | hadoop | hdfsadmingroup | drwxr-xr-x | May 17, 2024 01:58 PM |
| retail_logs | | hadoop | hdfsadmingroup | drwxr-xr-x | May 17, 2024 01:58 PM |
| sample_data.csv | 567 bytes | hadoop | hdfsadmingroup | -rw-r--r-- | May 17, 2024 01:58 PM |
| spark | | hadoop | hdfsadmingroup | drwxr-xr-x | May 17, 2024 01:58 PM |

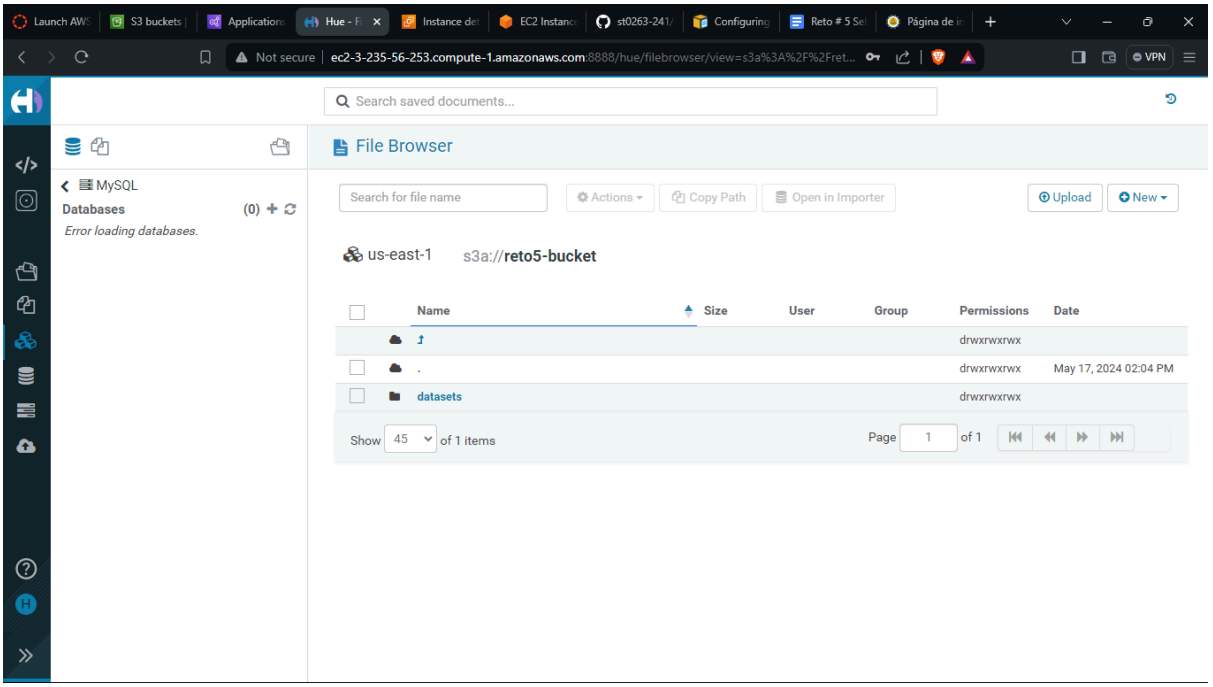
Por ultimo subiremos estas mismas carpetas a nuestro bucket de s3 en nuestro hue el cual deberia salir asi

us-east-1 s3a://

| Name | Size | User | Group | Permissions | Date |
|---|---|---|---|---|---|
| . | | | | drwxrwxrwx | |
| aws-logs-230380621223-us-east-1 | | | | drwxrwxrwx | |
| reto5-bucket | | | | drwxrwxrwx | |

Show 45 of 2 items Page 1 of 1

ingresamos a el “reto5-bucket”



y subiremos la carpeta de datasets que tambien se subio en el apartado de files para que se vea de la siguiente manera