Data Engineering Lab

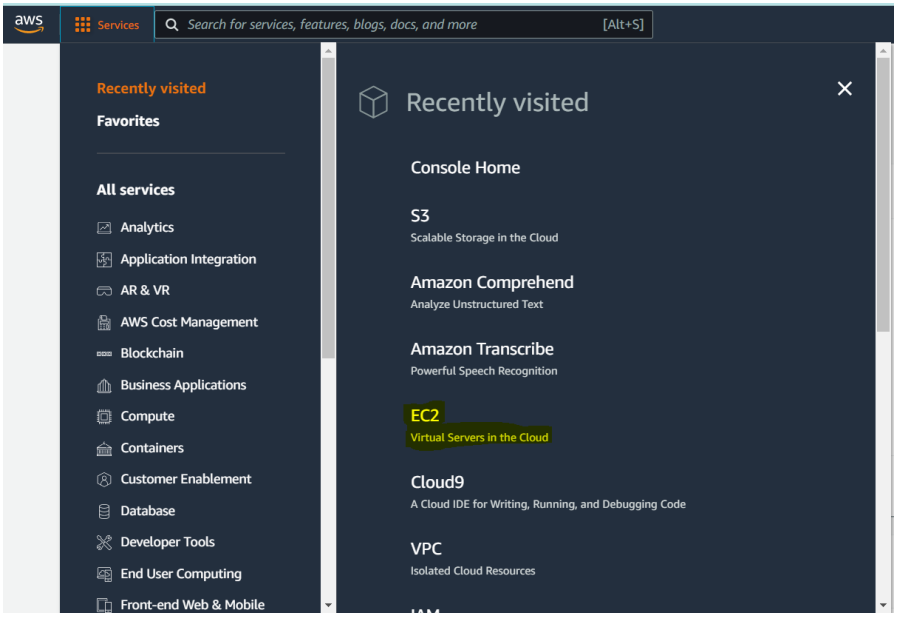
Gain hands-on experience on Scalable Data Engineering on AWS Cloud!

Kinesis to S3

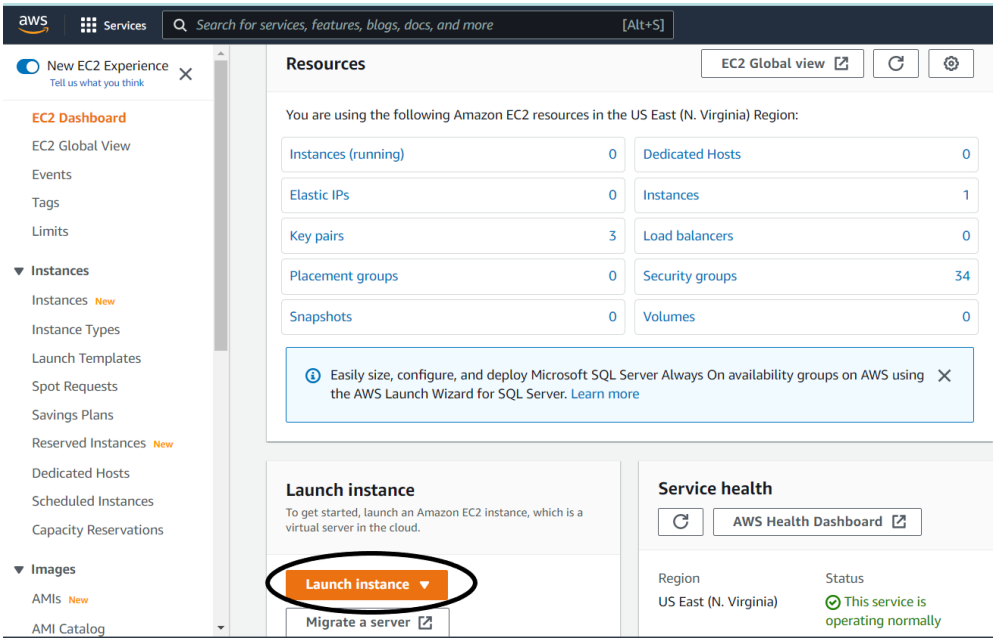
Step 1: Launch EC2 Instance

1. Go to the AWS console and click on “Services”.

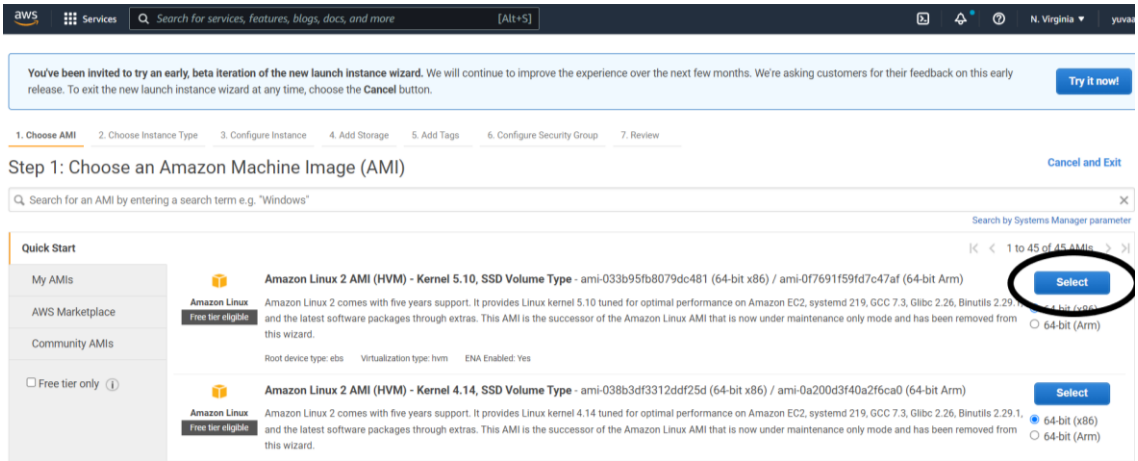
   Then choose EC2 from the dropdown.



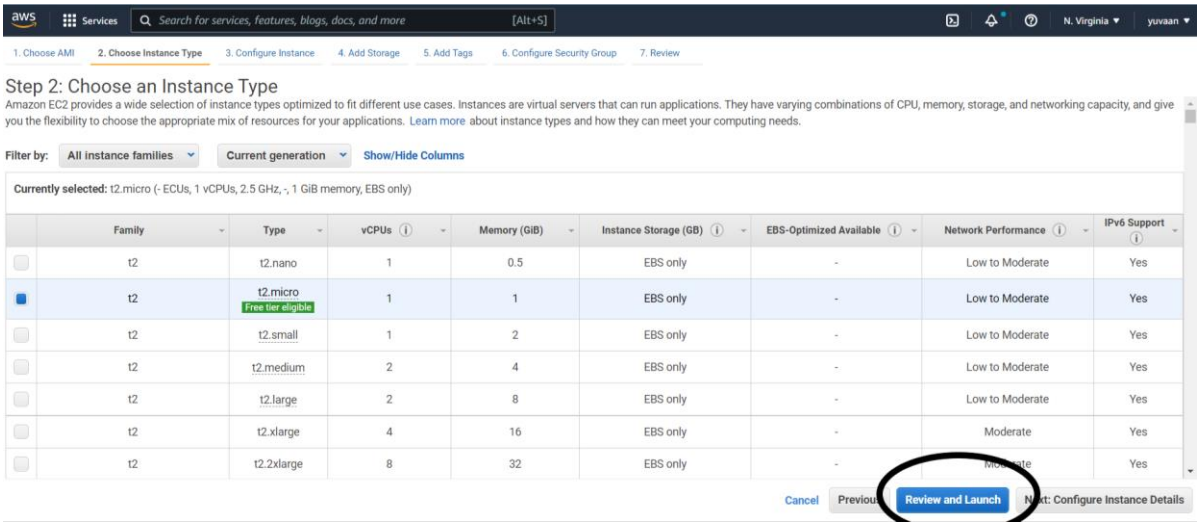
2. On the EC2 Dashboard, Click on the “**Launch Instance**” button



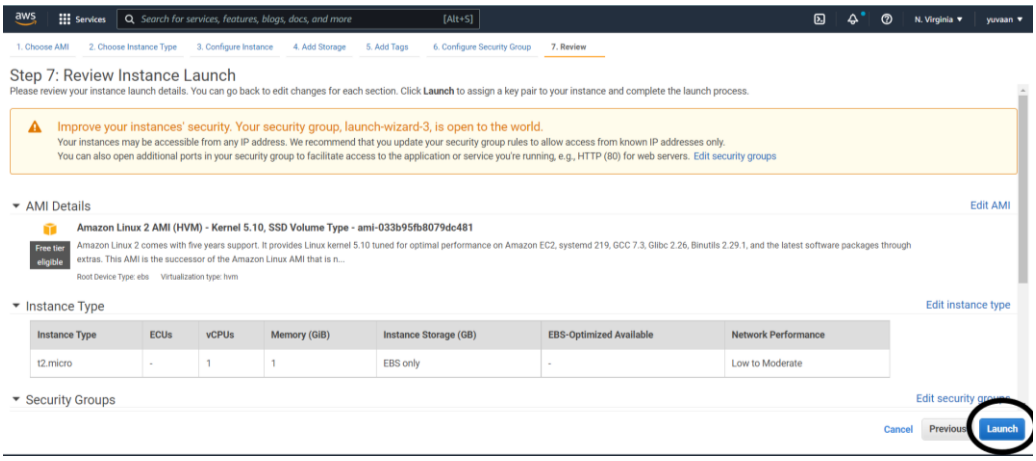
3. On the next screen of the launch wizard, select an AMI (Amazon Linux 2 AMI)



4. On the next screen of the launch wizard, select an instance family. For this demo, the default selected instance family "t2.micro" will work fine. Then click on the 'Review and launch' button.

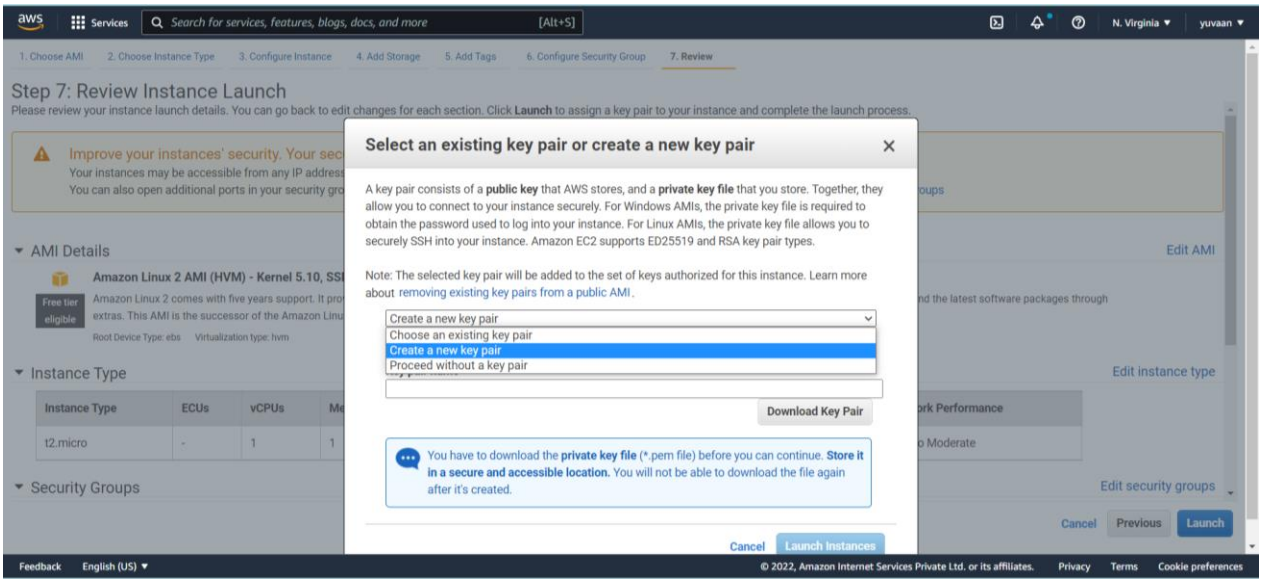


5. On the next screen, click on the 'Launch' button to start an EC2 instance.

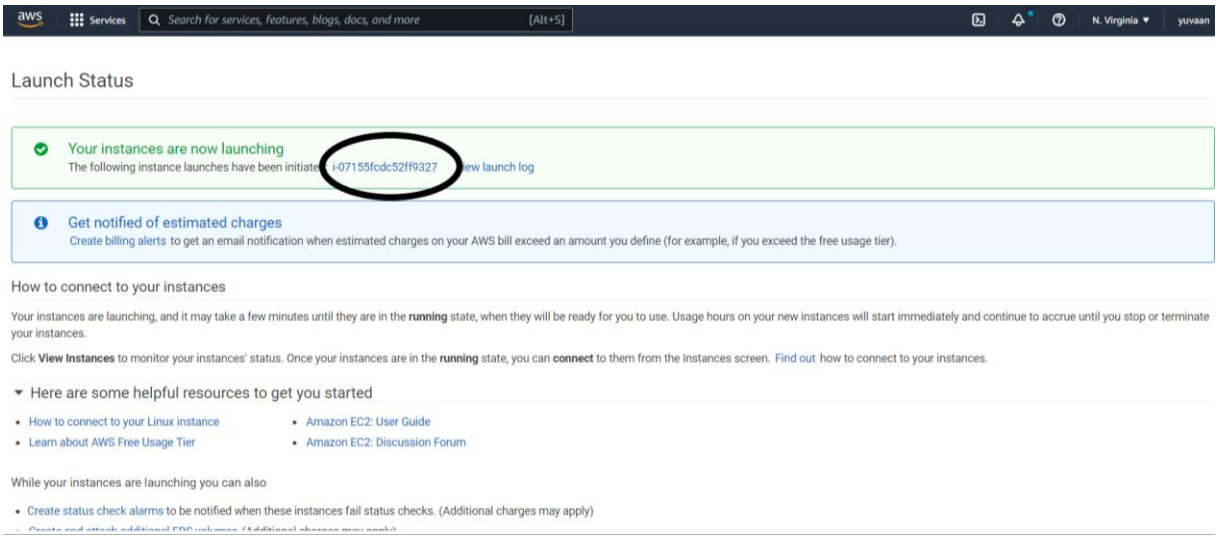


6. On the next Screen, select "Create a new key-pair" from the dropdown and click on

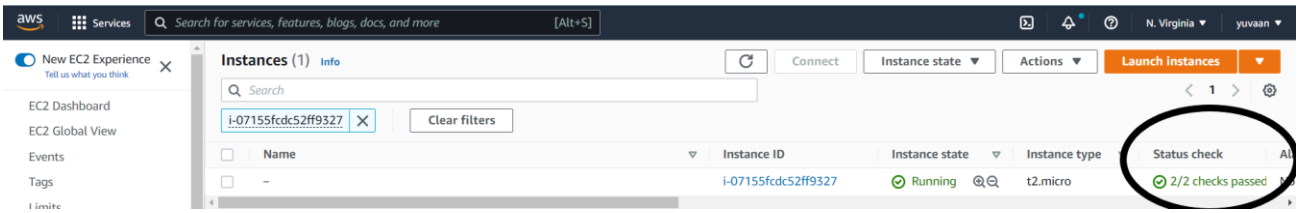
the “Download Key Pair” button. After the download finishes, click on the “Launch Instance” button.



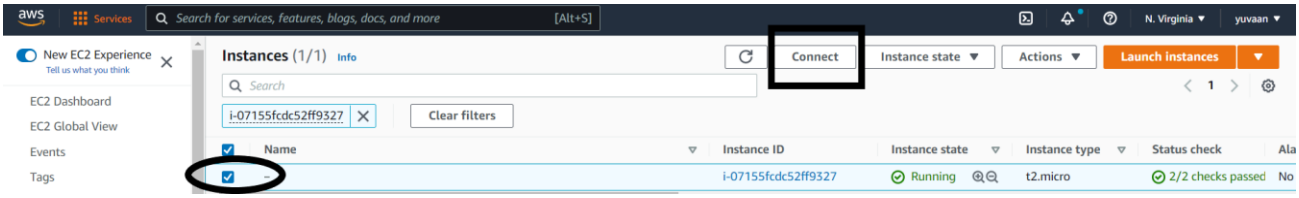
7. After you launch the instance, “Launch Status” will appear/ click on the instance id(a hexadecimal string) to view the running instance.



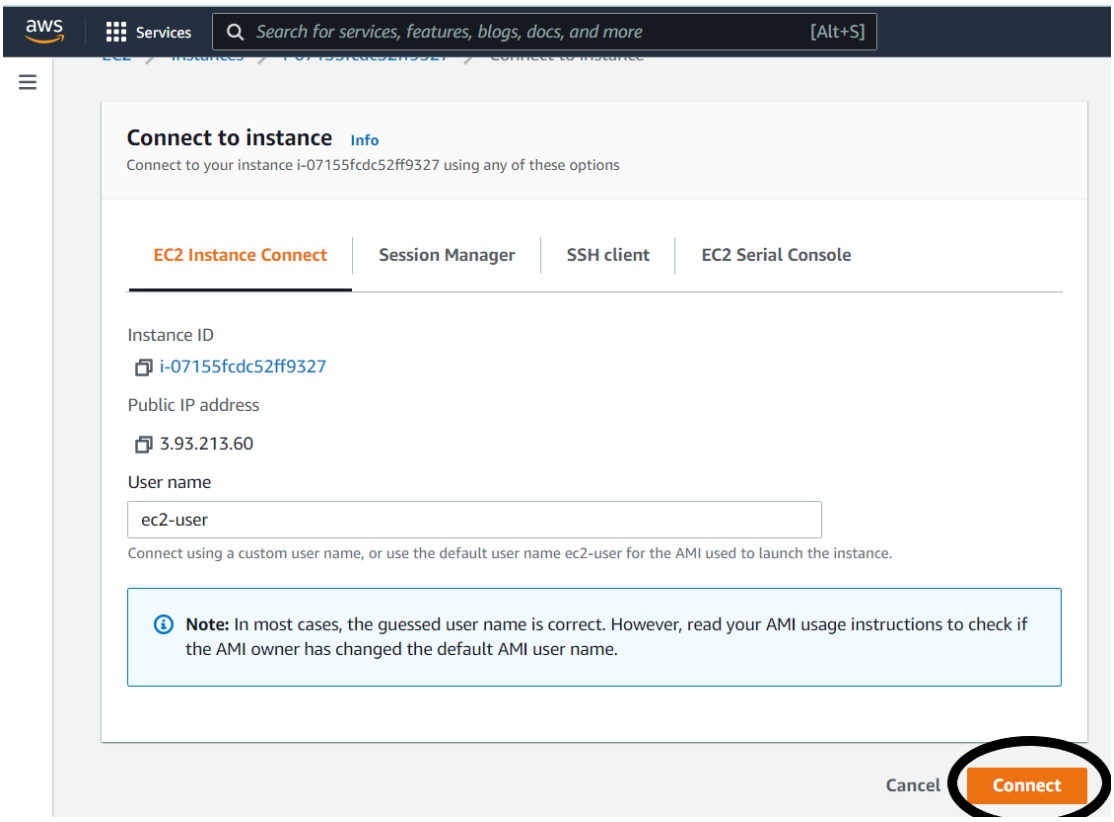
8. This screen will show you the instance that you just launched. Before connecting to the instance, wait for instance status to show passed.



9. Now select the checkbox next to the instance name and then click on the ‘Connect’ button.

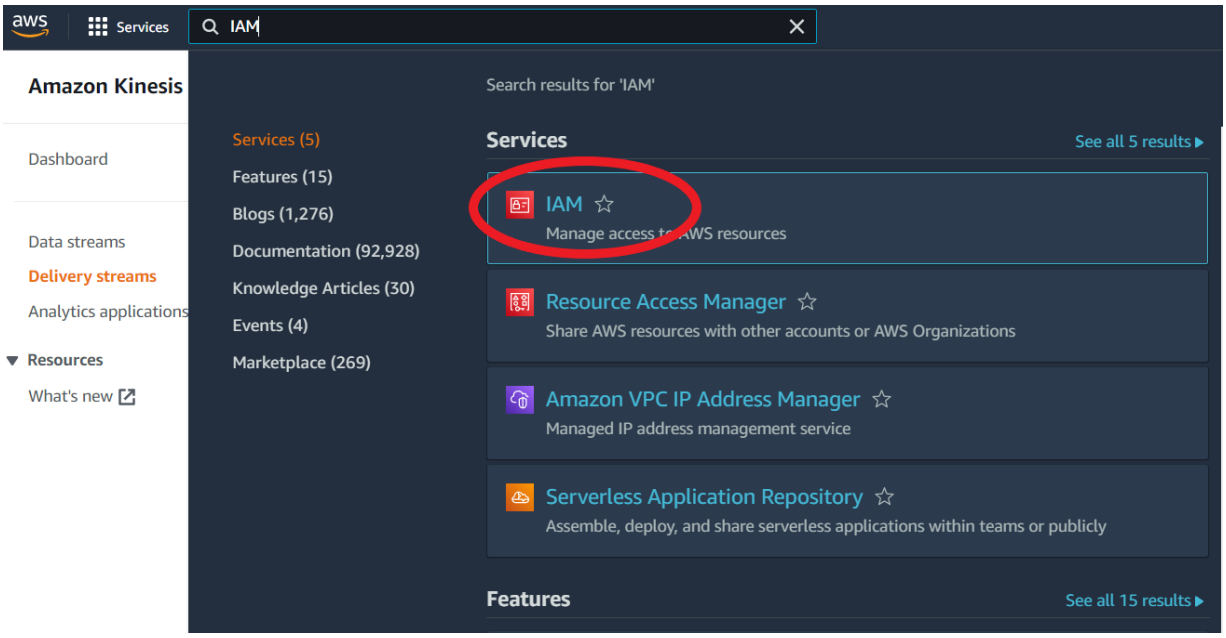


10. On the next screen, click on 'Connect" once again.

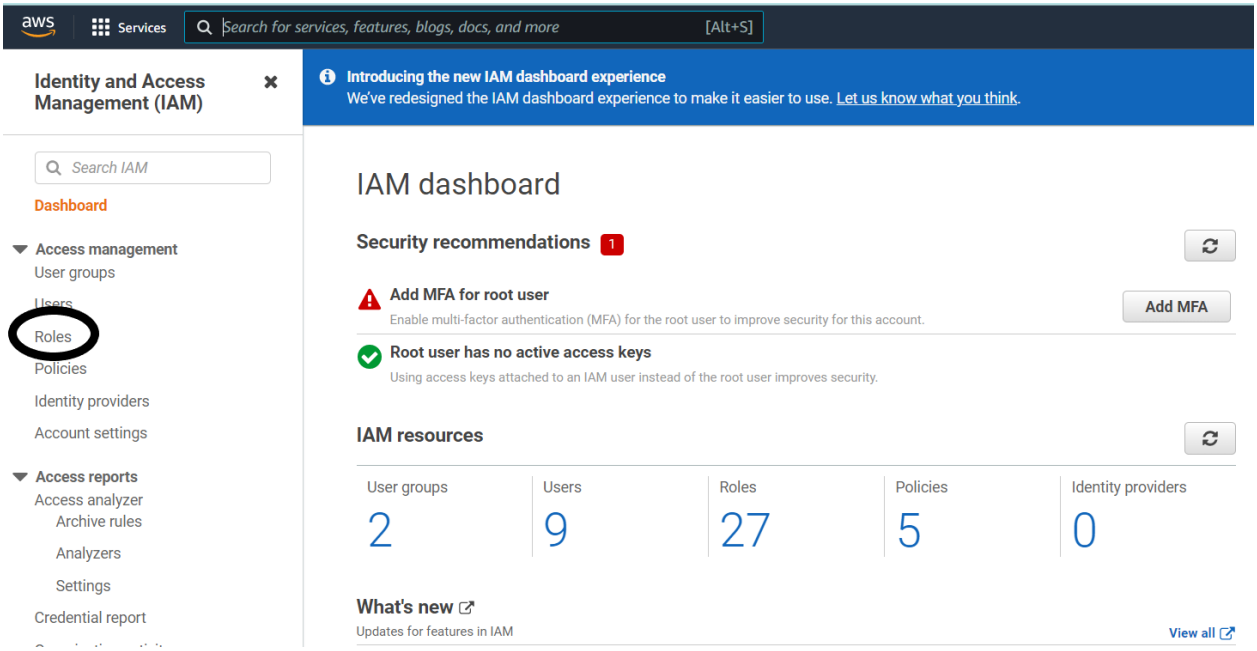


Step 2: Create an IAM role for EC2 instance

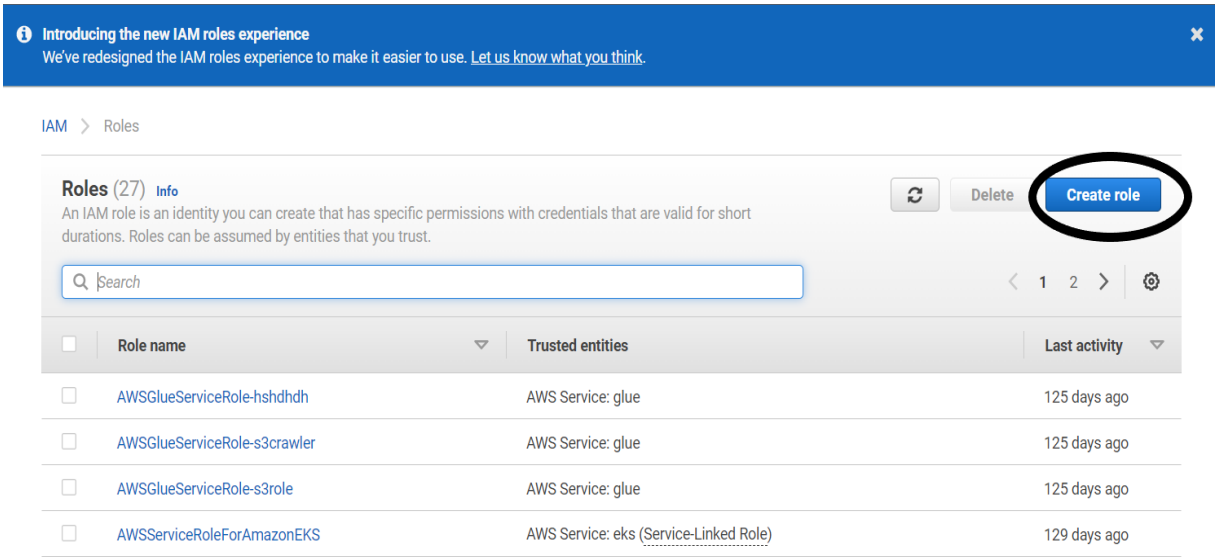
1. Go to “Services” and search for IAM.



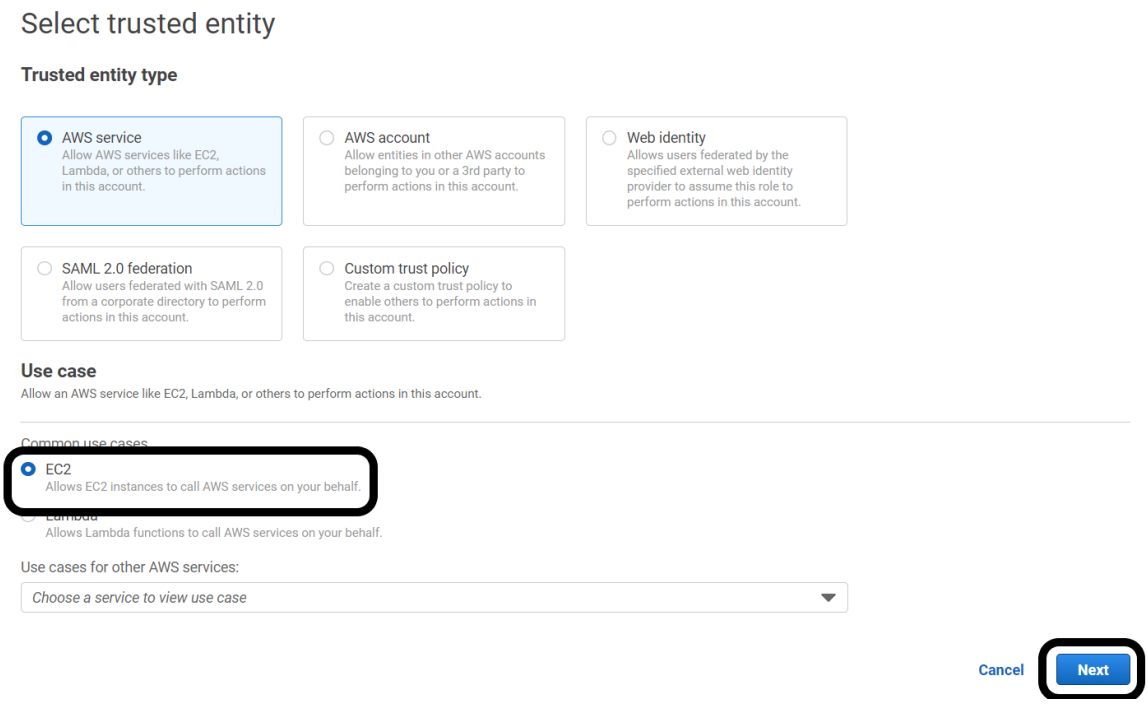
2. On the IAM Dashboard, select “Roles” from the left hand side navigation menu.



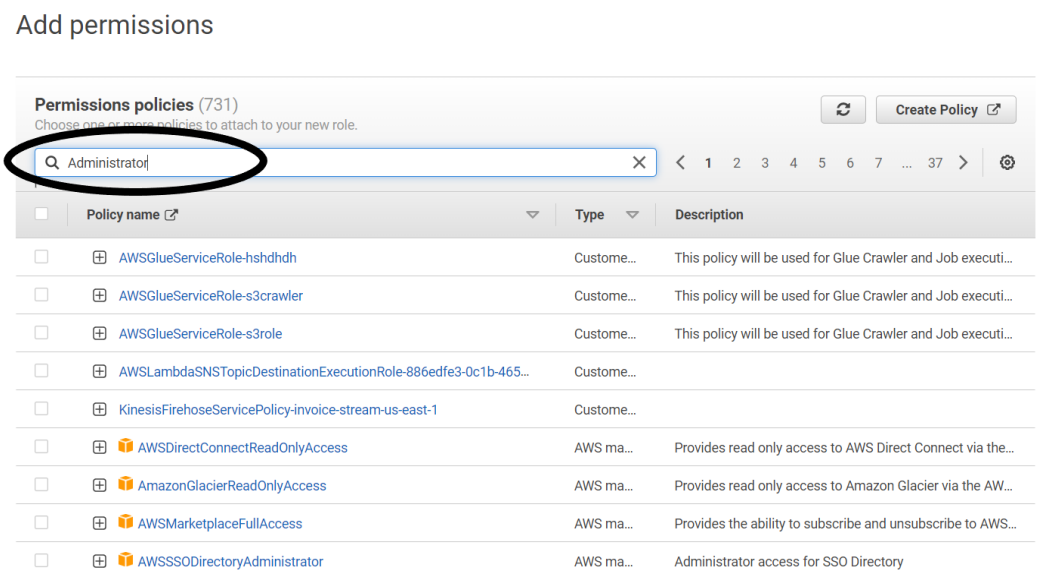
3. Now click on the”Create Role” button.



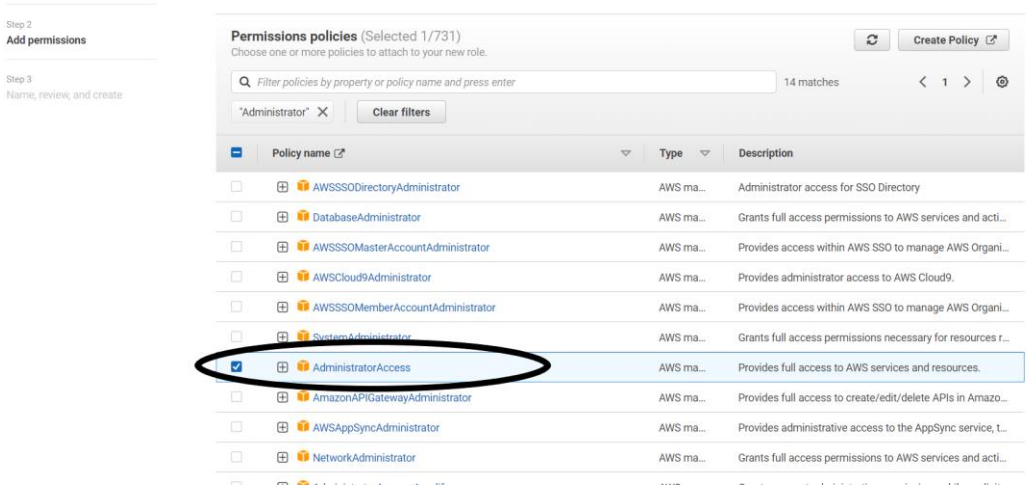
4. On the next screen, check the "EC2" radio button and click on Next.



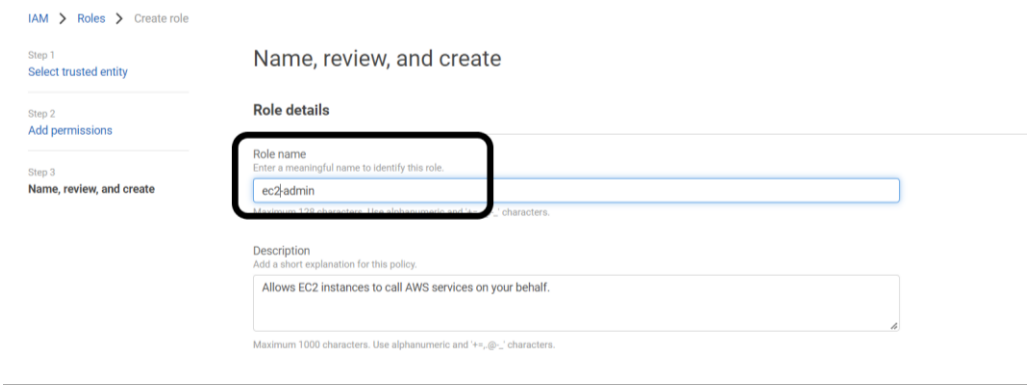
5. In the "Permissions policies" window, search for "Administrator."



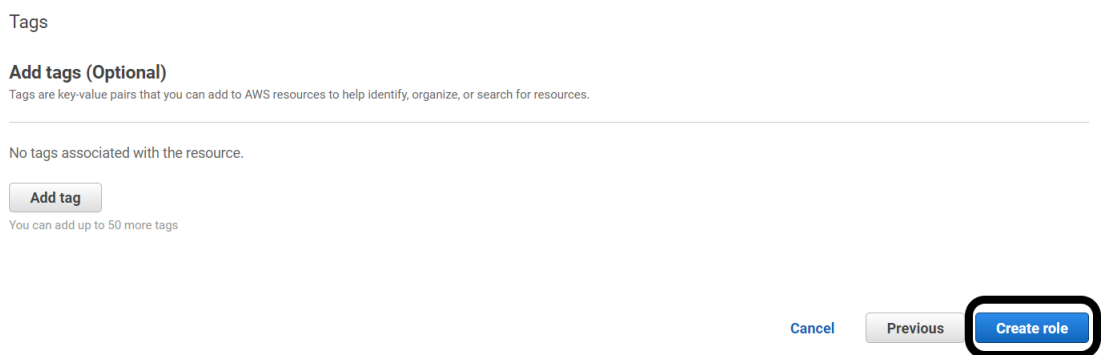
6. Select the “Administrator Access” from the list and click Next



7. On the next screen, you have to provide a role name. You can provide any name.

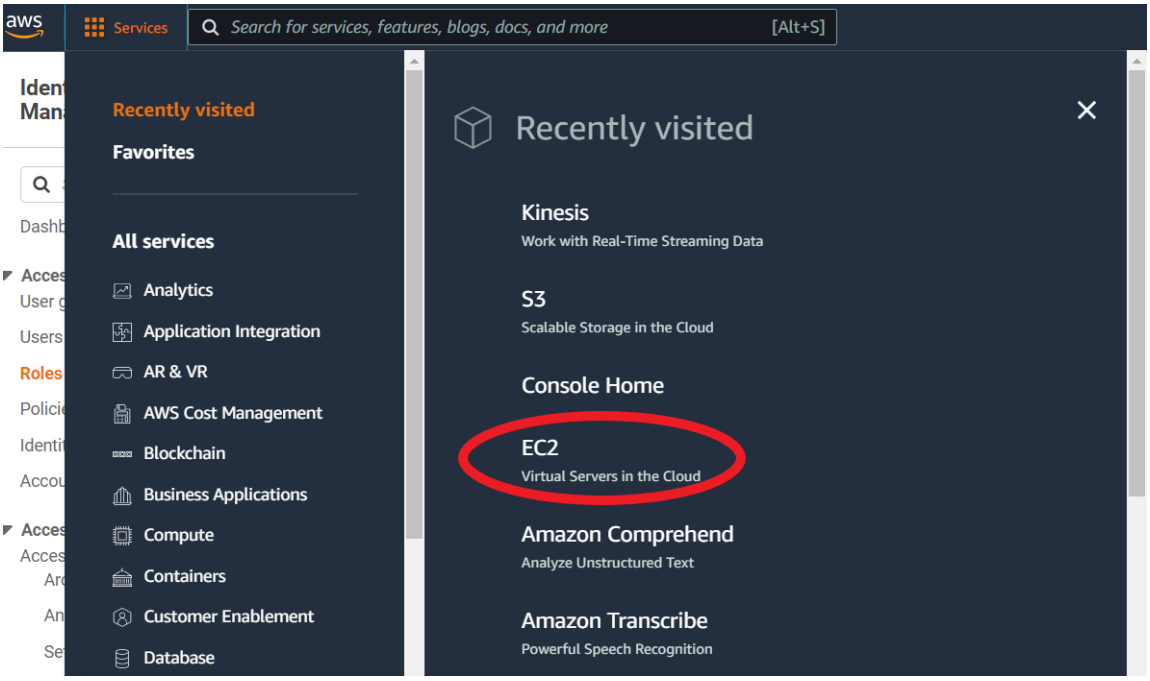


8. Scroll down to the bottom of the screen and click on the “Create role” button.

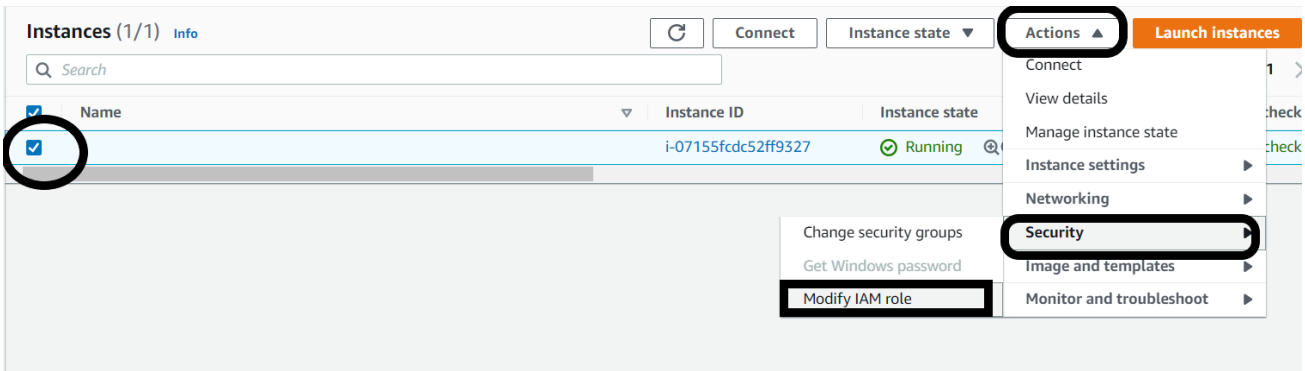


Step 3: Attach IAM role to EC2

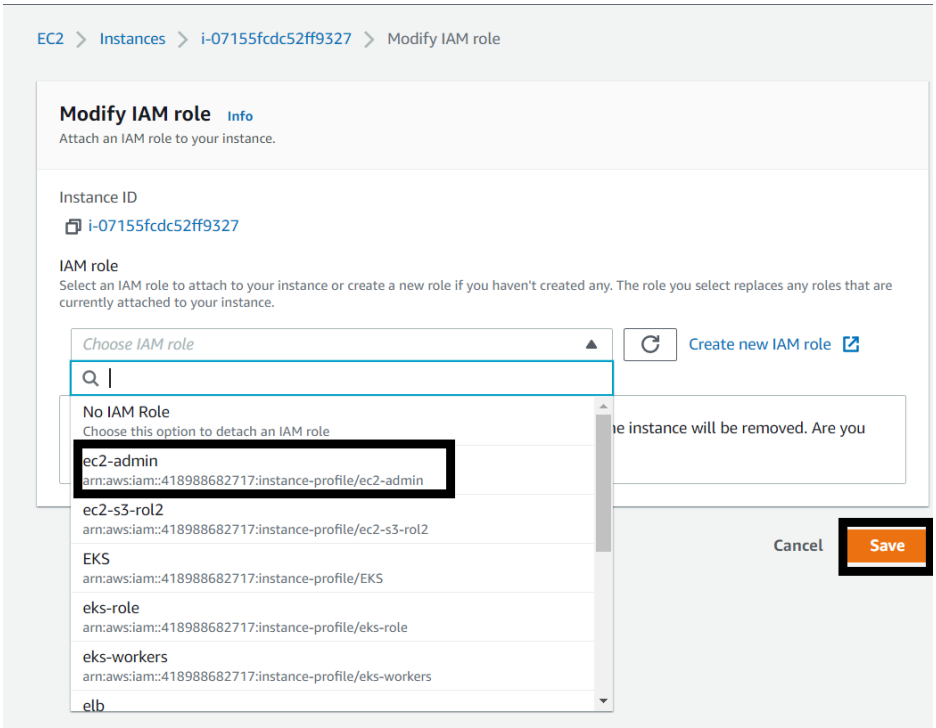
1. Choose "EC2" from the services and go to the EC2 Dashboard.



2. Select your EC2 instance and click on "Actions". Under actions, click on "Security" and choose "Modify IAM Role" from the dropdown.



3. On the next screen, choose the IAM role created in Step 2 from the dropdown and click on the "Save" button.



Step 4: Download Simulation APP on EC2

1. After connecting to EC2, download the simulation application on EC2.

   Run the command given below to download the code.

```
wget https://invoice-generator-
di.s3.amazonaws.com/InvoiceGenerator.zip
```

2. Extract the files using the "unzip" command. The unzip command given below will create a folder named "InvoiceGenerator" containing two files (customer_retails.csv and InvoiceGenerator.py)

```
unzip InvoiceGenerator.zip
```

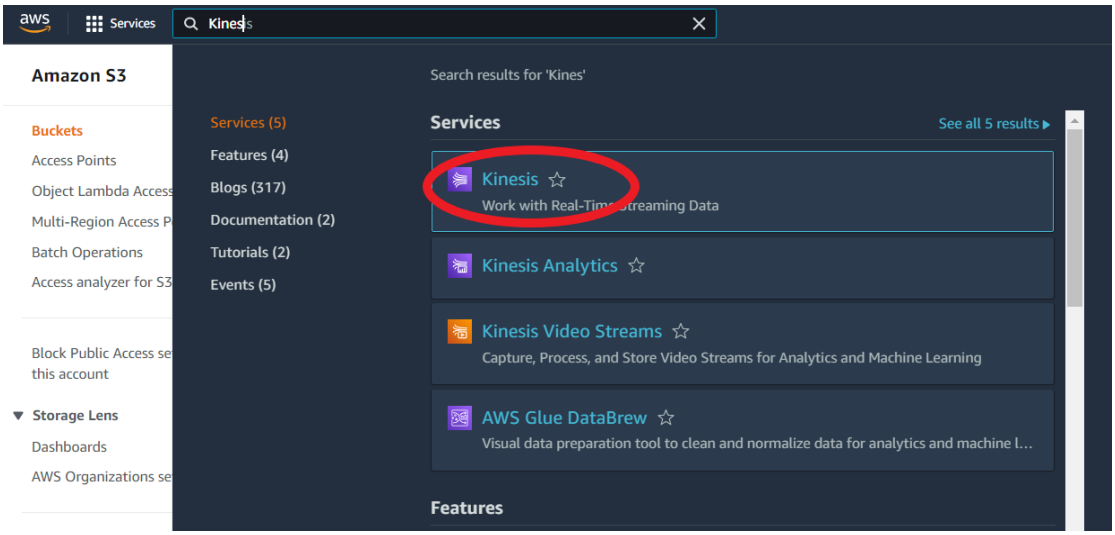
3. On running, the InvoiceGenerator.py will produce invoices logs at the path ***"/var/logs/invoices"***. So, you must give execute permission to the file InvoiceGenerator.py and create the logs directory as below:

```
chmod +x InvoiceGenerator.py
mkdir /var/logs/invoices
```

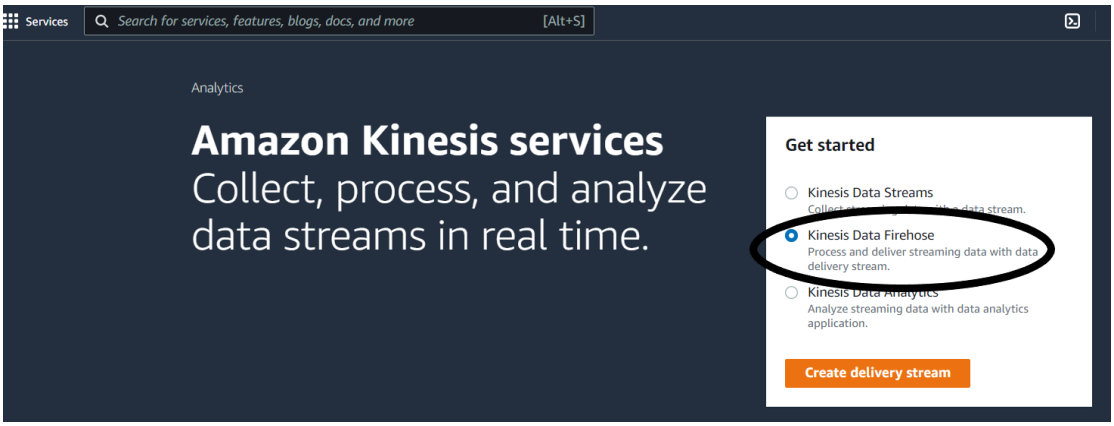
Step 4: Create Kinesis Firehose on AWS

Amazon Kinesis Data Firehose is an extract, transform, and load (ETL) service that reliably captures, transforms, and delivers streaming data to data lakes, data stores, and analytics services.

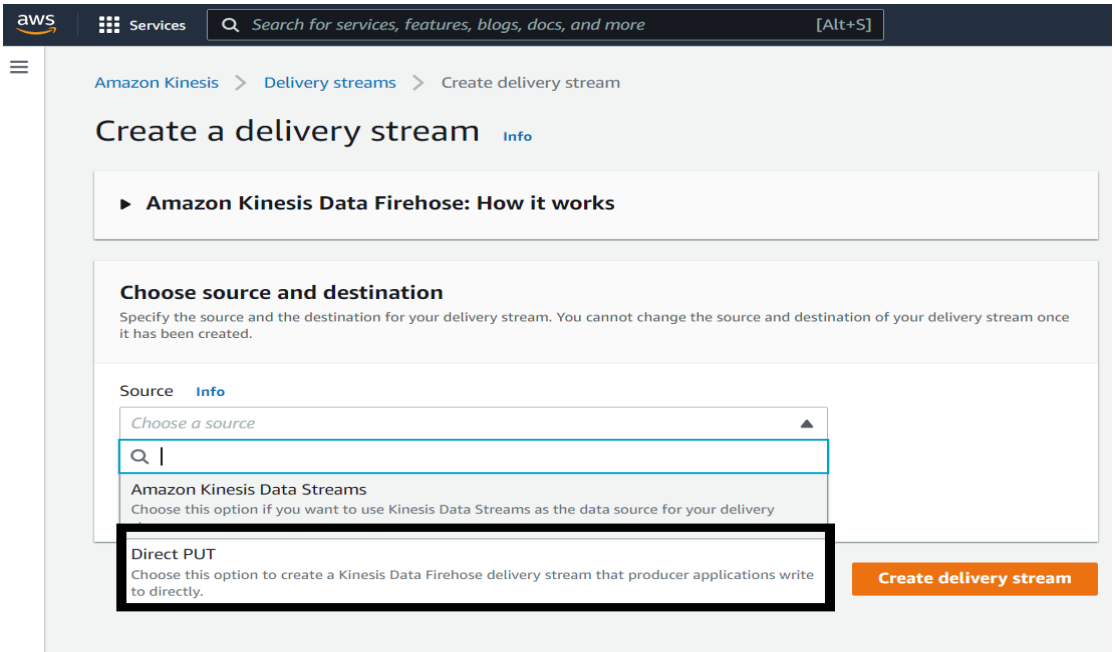
1. First search for Kinesis in AWS Services and double click on “Kinesis” from the dropdown



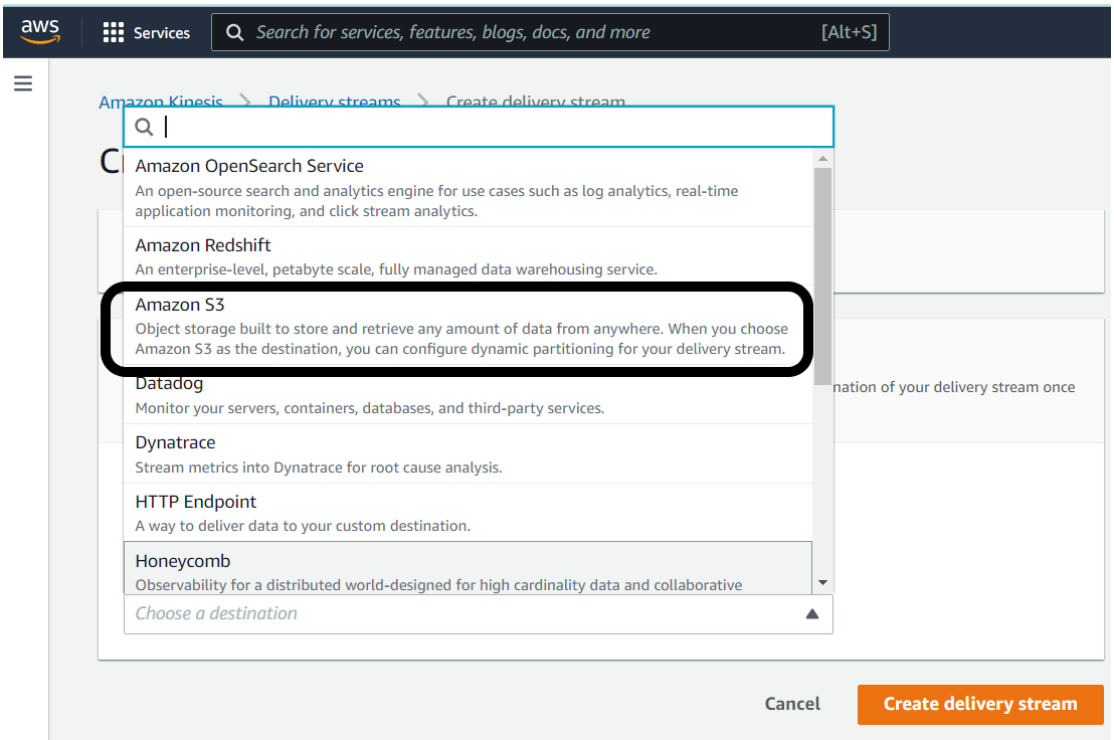
2. Select the “Kinesis Firehose” radio button and click on the “Create Delivery Stream” button.



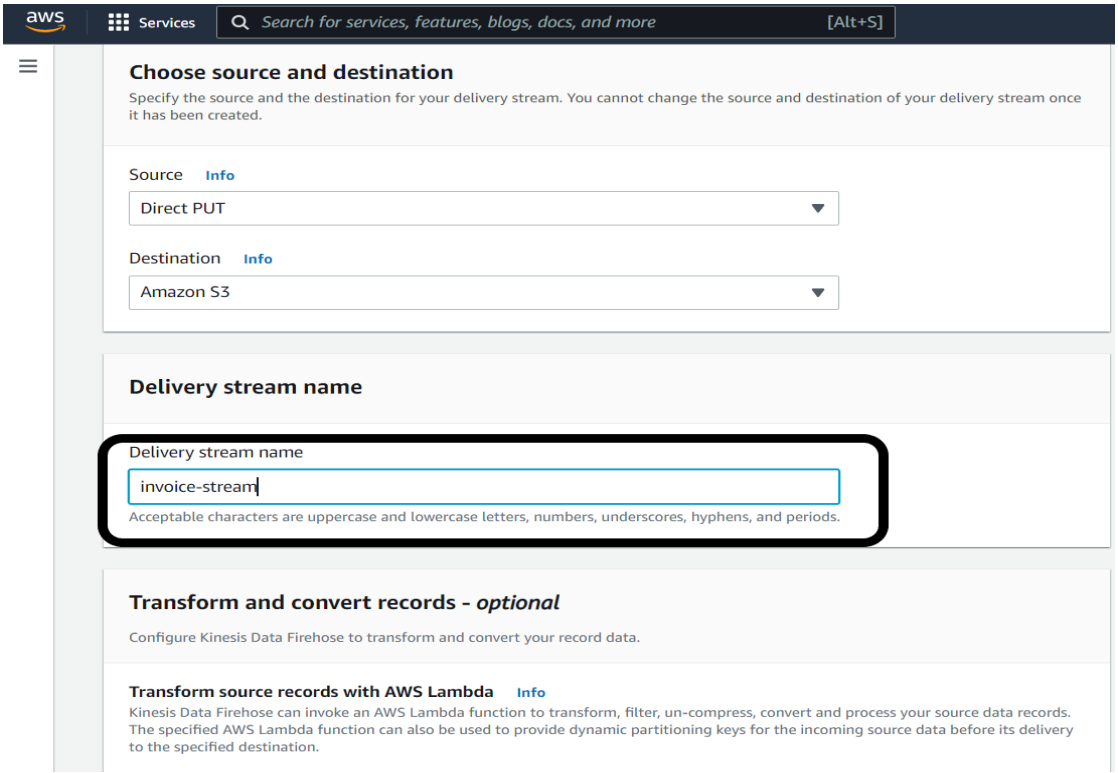
3. On the next screen ,select “Direct PUT” in the source. This is because in this exercise, we will directly put data from the EC2 instance into this Kinesis Firehose using the Kinesis Agent.



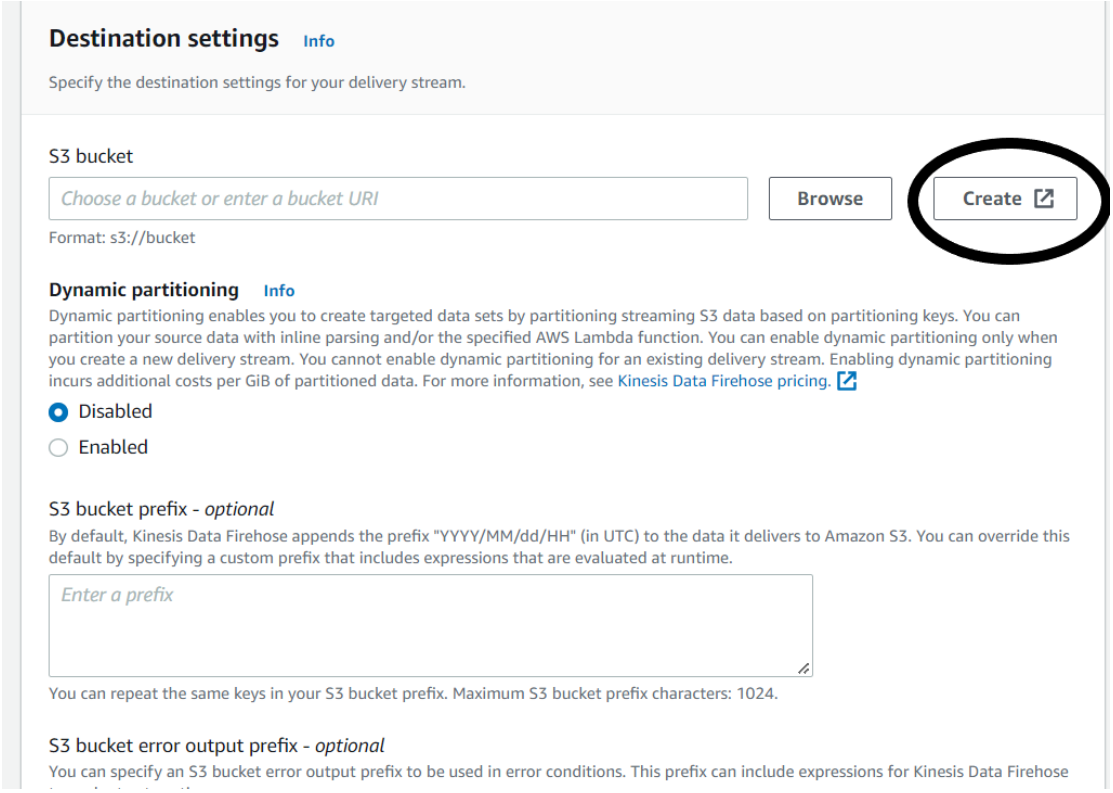
4. In the "Destination" dropdown, select "AWS S3." We will be pushing our raw logs there, that are generated by the Invoice Generator Simulation App on EC2.



5. Provide a name to your delivery stream, like “invoice-stream”.

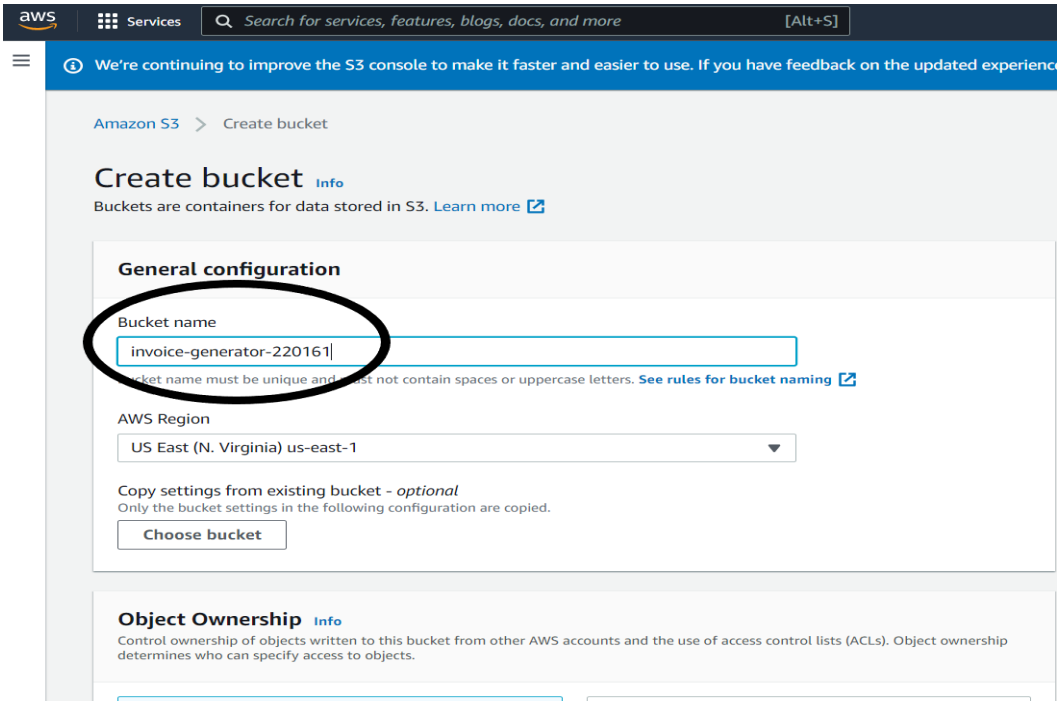


6. Scroll down and under “Destination settings”, click on the “Create” button. It will open an S3 dashboard to choose where data will be loaded on AWS.

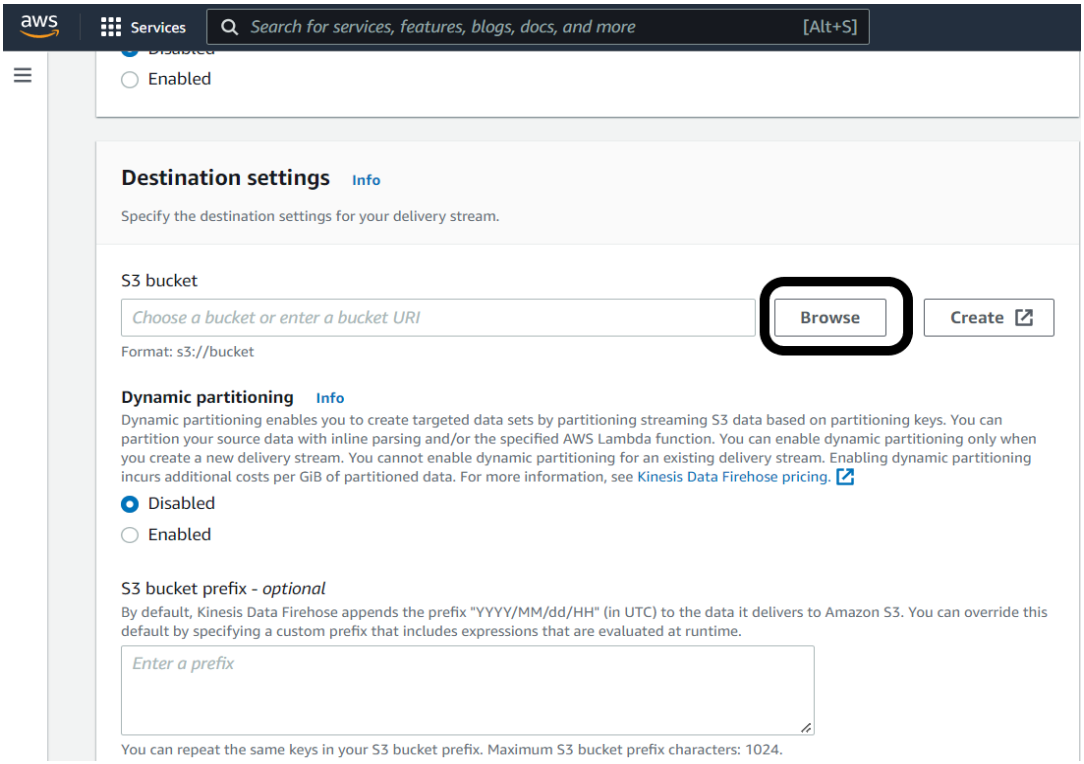


7. In the S3 Dashboard, provide a unique name to your S3 bucket. Keep the default

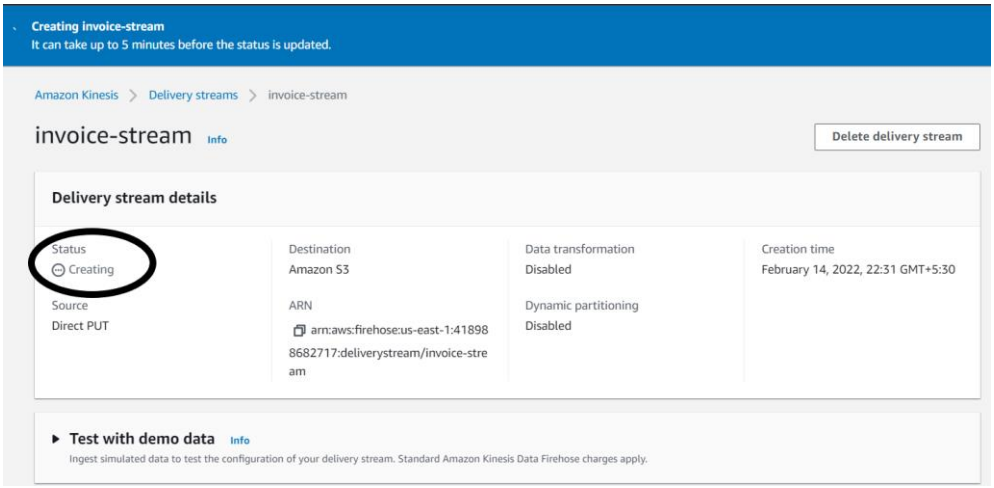
configuration, scroll down, and click on the “Create Bucket” button.



8. Now go back to the Kinesis Dashboard. Under the “Destination settings”, locate the “S3 bucket” path, click on “Browse”, and choose the S3 bucket you just created.. Finally, scroll down, and click on the “Create Delivery Stream” button.



9. You have successfully created a Kinesis Firehose stream. You may have to wait for stream status to change to “Active”.



Step 5: Run Kinesis Agent on EC2

Kinesis Agent is a **stand-alone Java software application** that offers an easy way to collect and send data to Kinesis data streams and Kinesis Firehose.

1. Before running the Simulation app, first download the AWS Kinesis Agent on EC2.

```
sudo yum install -y aws-kinesis-agent
```

2. Let us configure the Kinesis-agent to listen to the logs generated by our InvoiceGenerator simulation app at the location “/var/logs/invoices”.

```
cd /etc/aws-kinesis
sudo nano agent.json
```

```
{
  "cloudwatch.emitMetrics": true,
  "kinesis.endpoint": "",
  "firehose.endpoint": "firehose.us-east-1.amazonaws.com",

  "flows": [
    {
      "filePattern": "/var/logs/invoices/*.log",
      "deliveryStream": "invoice-stream"
    }
  ]
}
```

3. Start the Kinesis Agent and go back to the home directory to start the simulation application to generate logs.

```
sudo service aws-kinesis-agent start
sudo chkconfig aws-kinesis-agent on
cd ~
```

Step 6: Run InvoiceGenerator Simulation Application

1. Start the application and generate 10000 lines in the log.

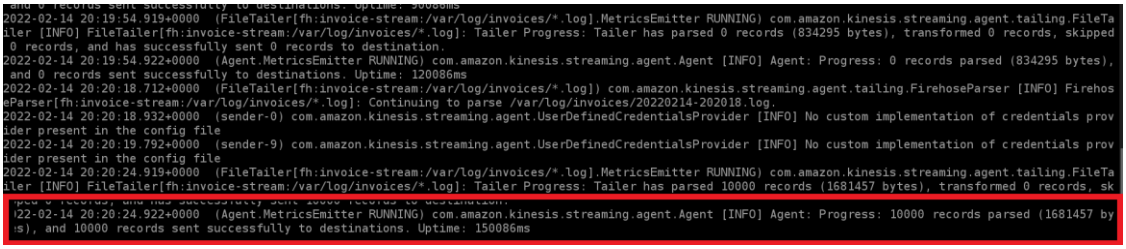
```
cd ~/InvoiceGenerator/
sudo ./InvoiceGenerator.py 10000
```

2. Check the logs directory

```
cd /var/logs/invoices
```

3. Check Kinesis logs

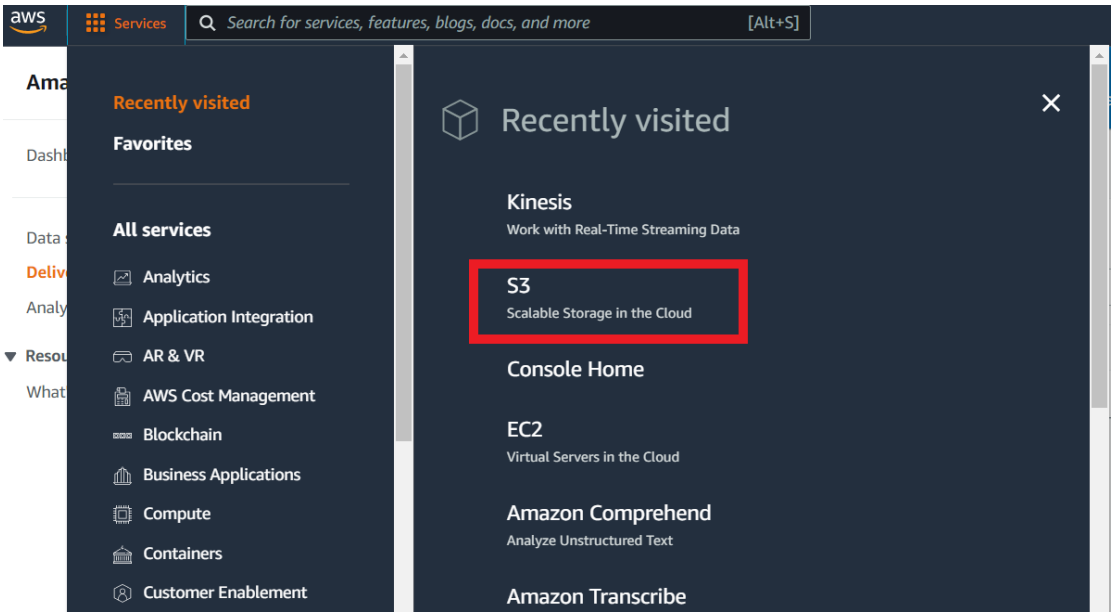
```
tail -f /var/log/aws-kinesis-agent/aws-kinesis-agent.log
```



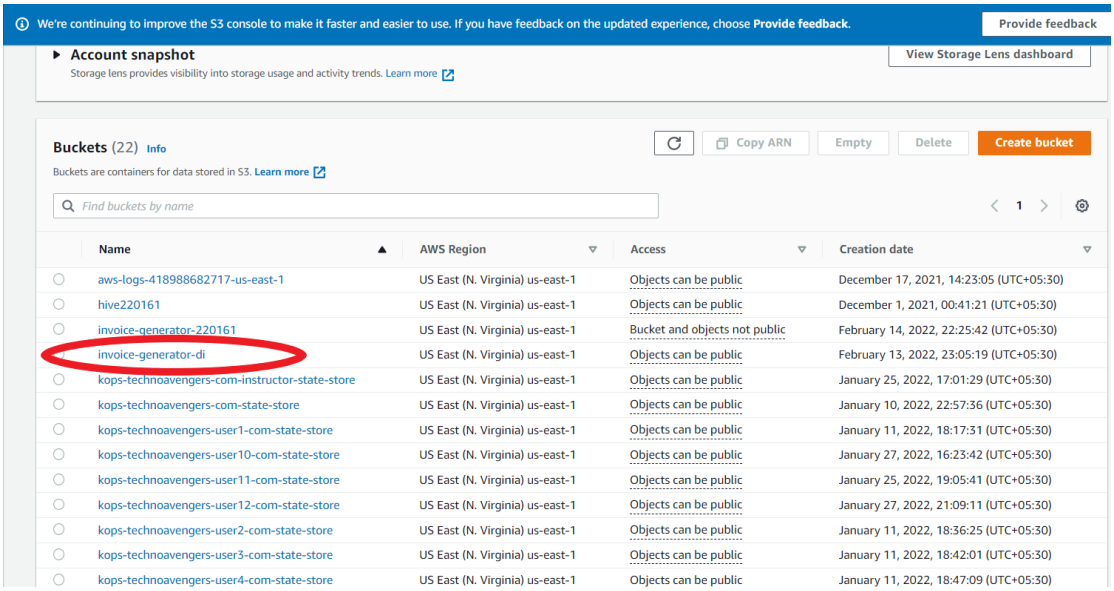
Step 7: Check S3

Please note that Firehose has a buffer time of minimum 1 minute to push data to the destination.

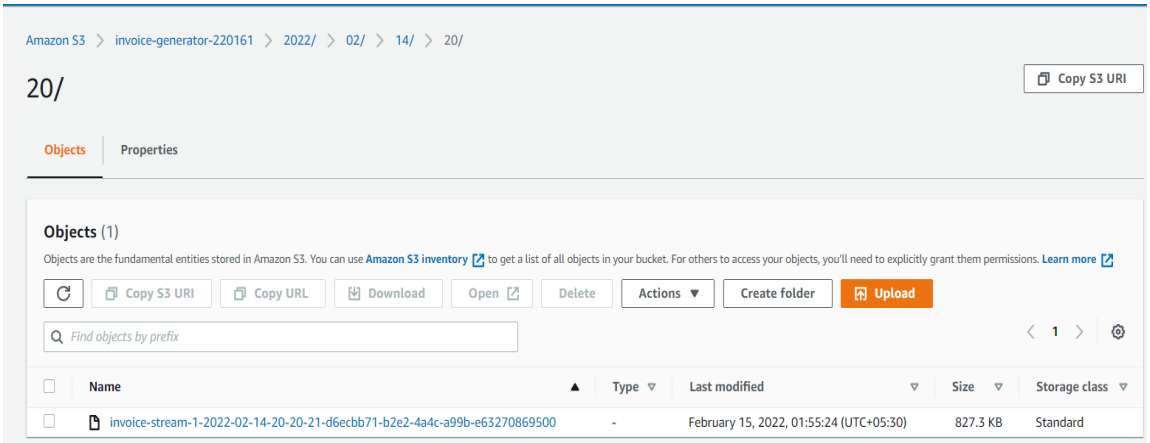
1. Search “S3” on the services dropdown.



2. Choose the bucket that we have configured as “Destination” in our Firehose.



3. Finally check whether data is published in S3.

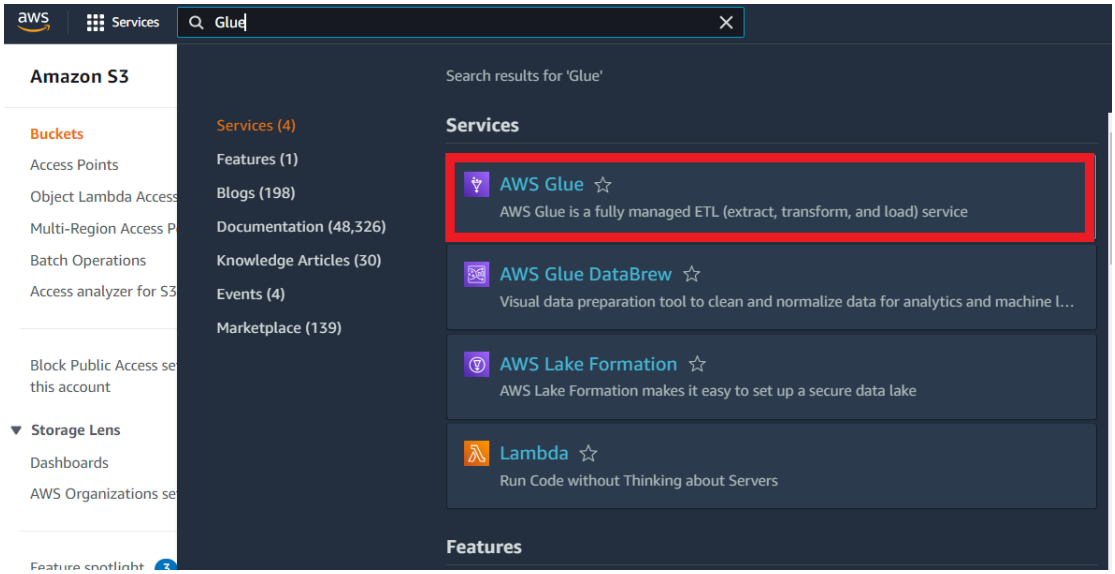


Data Catalog Using Glue

Once the data is in your data lake such as S3, you can create a Data Catalog from it using Glue.

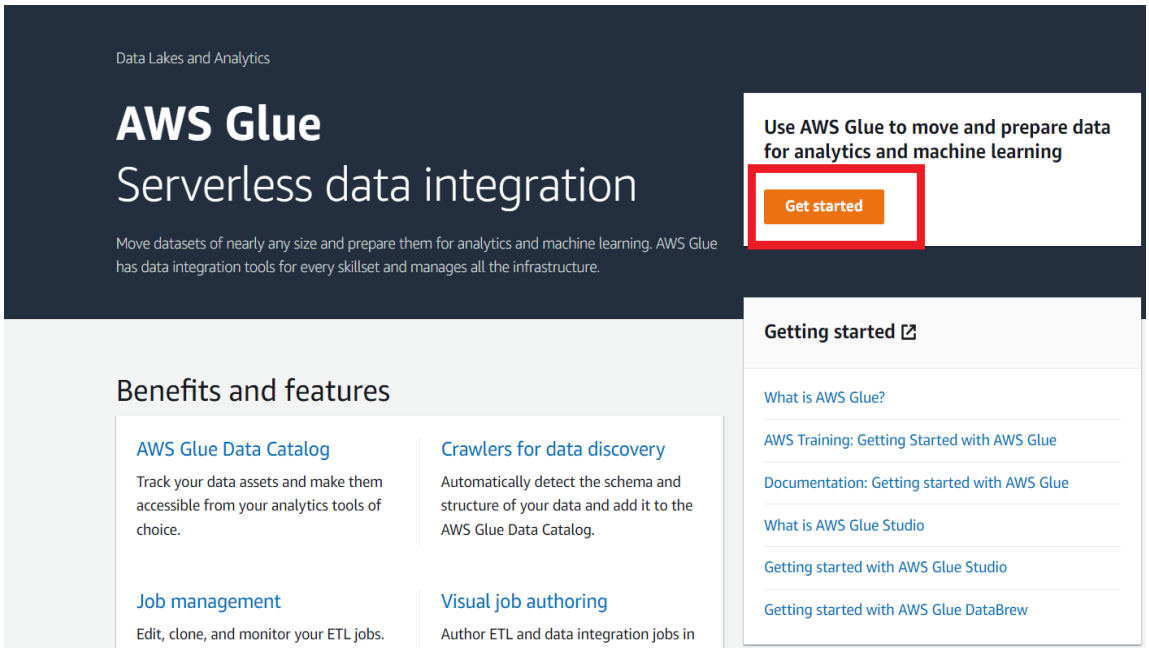
Step 1 : Start Glue

Type “Glue” in Services and select “AWS Glue” from the list.



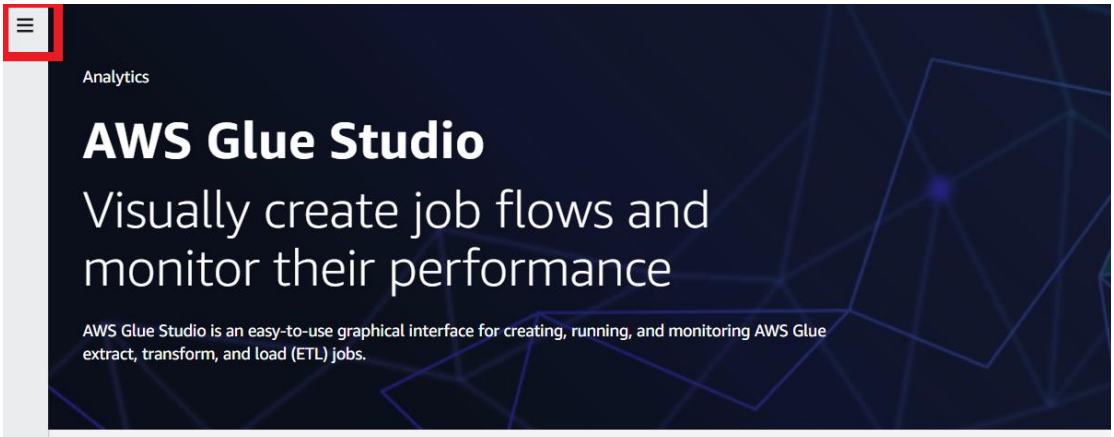
Step 2: Get

Click on “Get Started” to move forward.

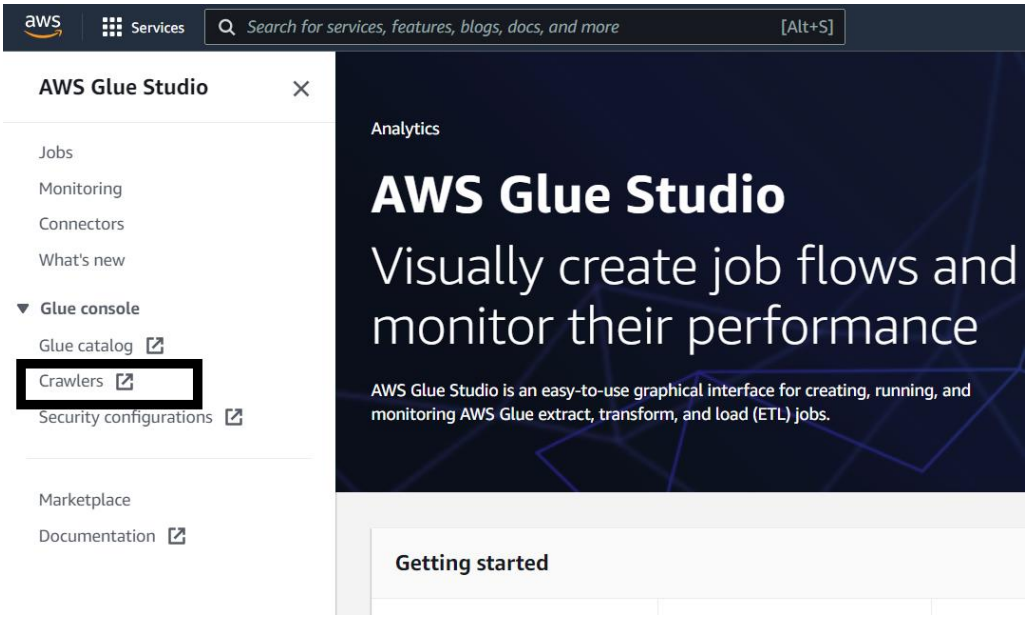


Step 3: Create Crawler

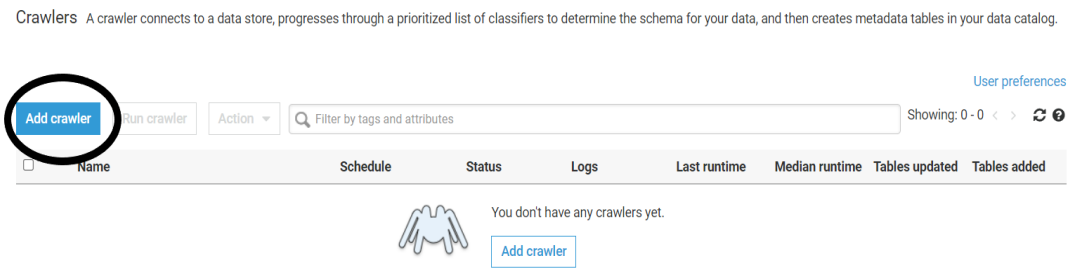
1. Click on the left hand side menu symbol to open the Glue Dashboard.



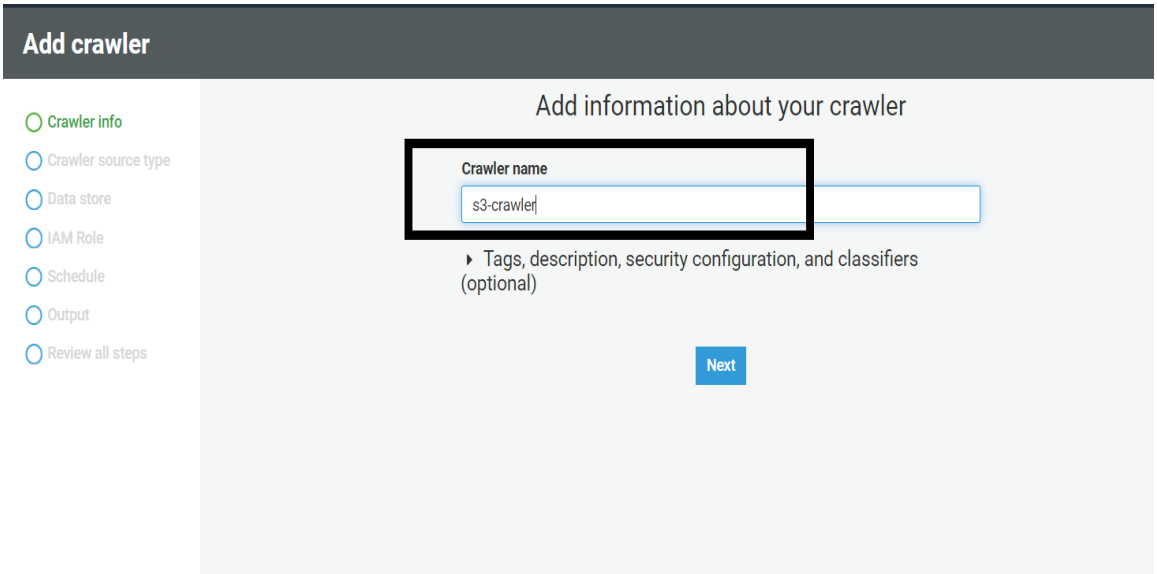
2. Click on "Crawlers" in the menu.



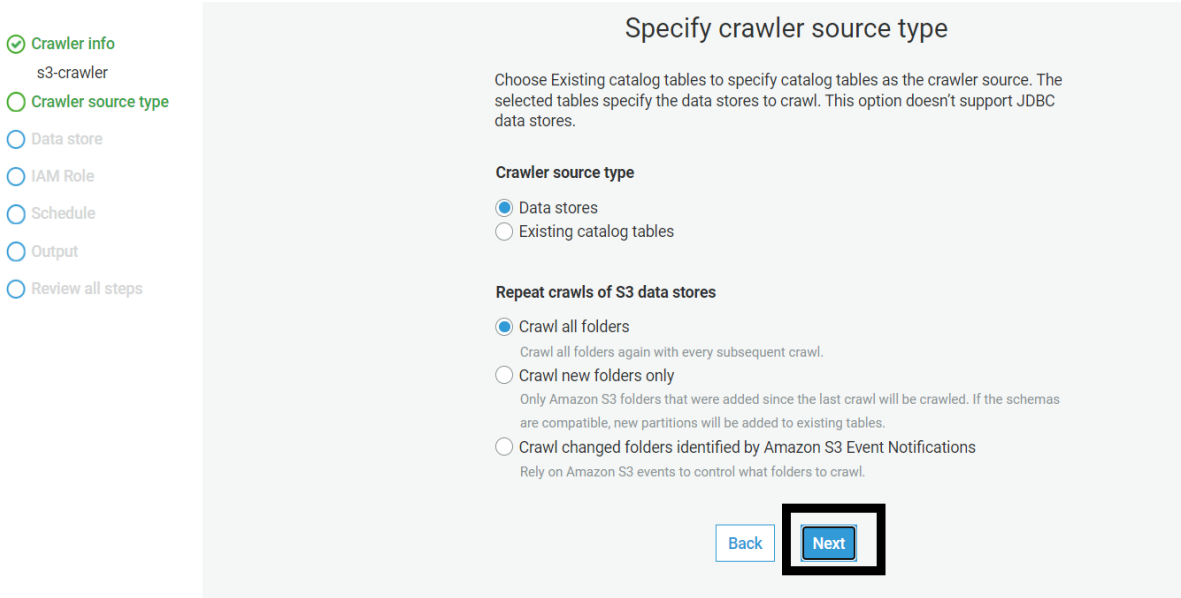
3. In the next screen, click on the "Add Crawler" button.



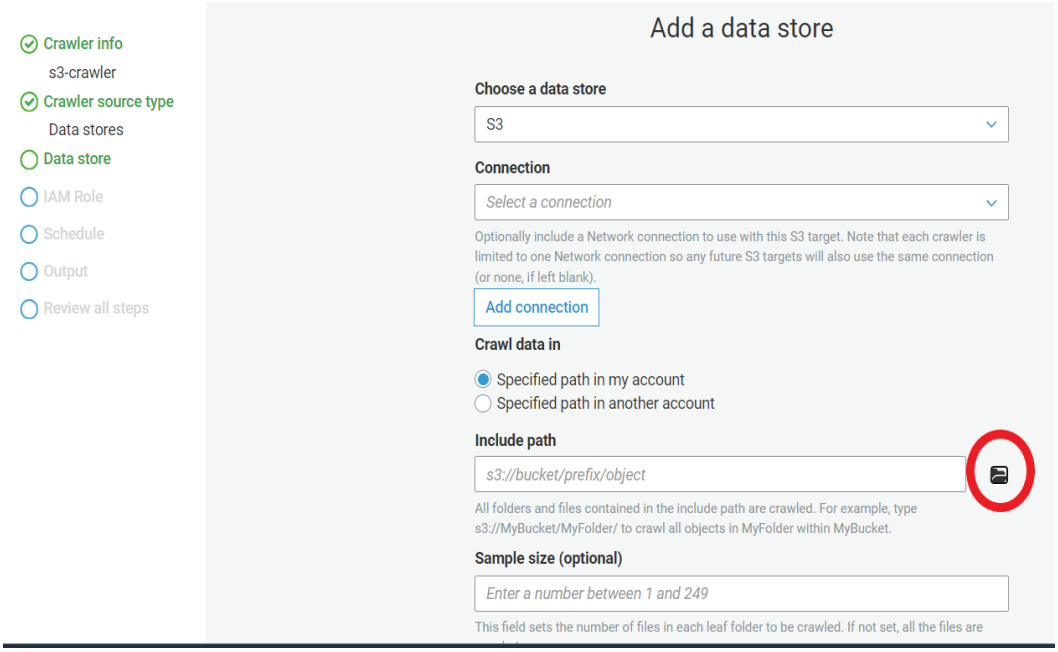
4. Give a name to your crawler (in this example *s3-crawler*) and click on Next.



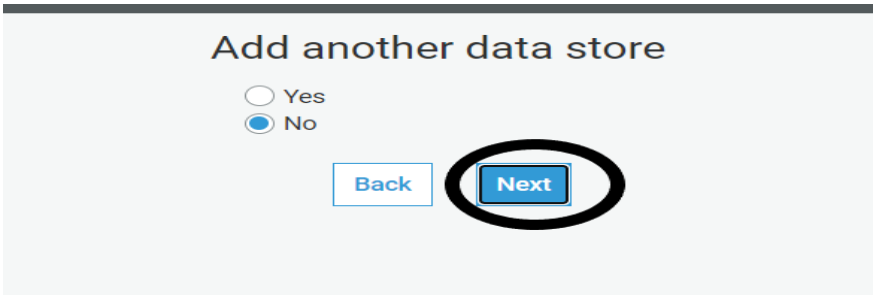
5. On the next screen, you have to define the crawler source type. Choose "Data Stores" under "Crawler source type". Keep the default option "Crawl all folders" under "Repeat crawls of S3 data stores". Click next.



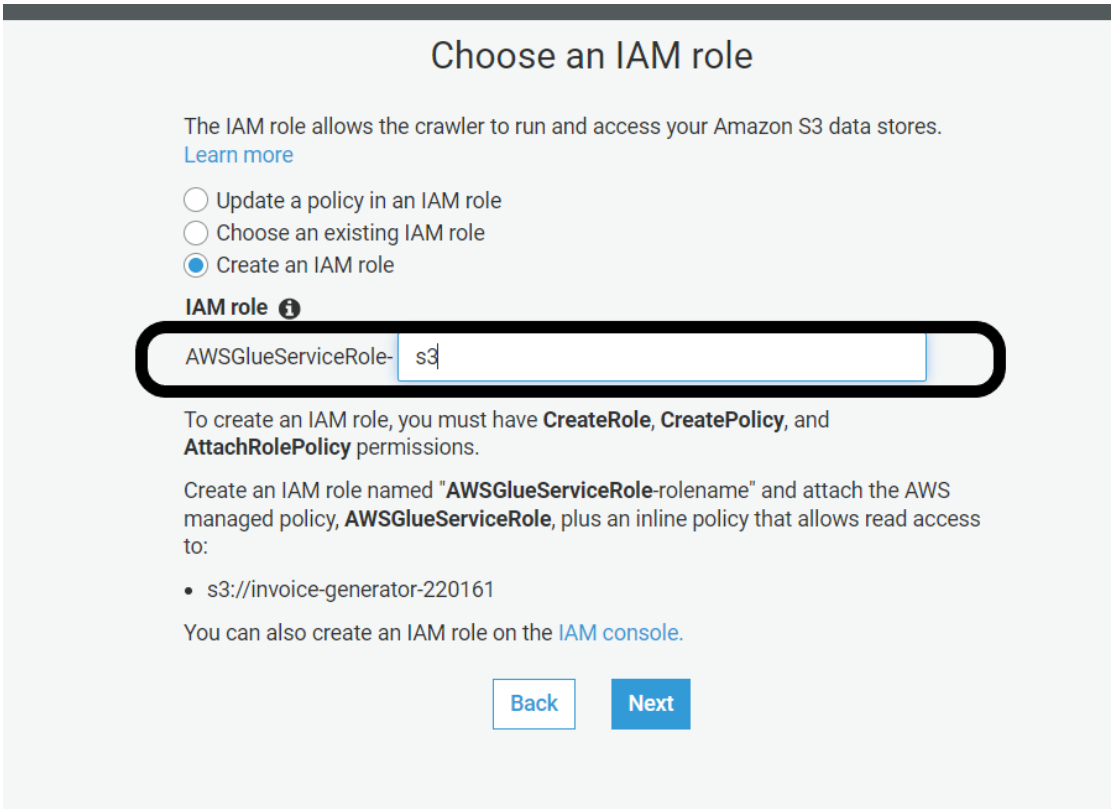
6. On the next screen, choose the S3 bucket from which you would like to create a data catalog in "Include path" and click next.



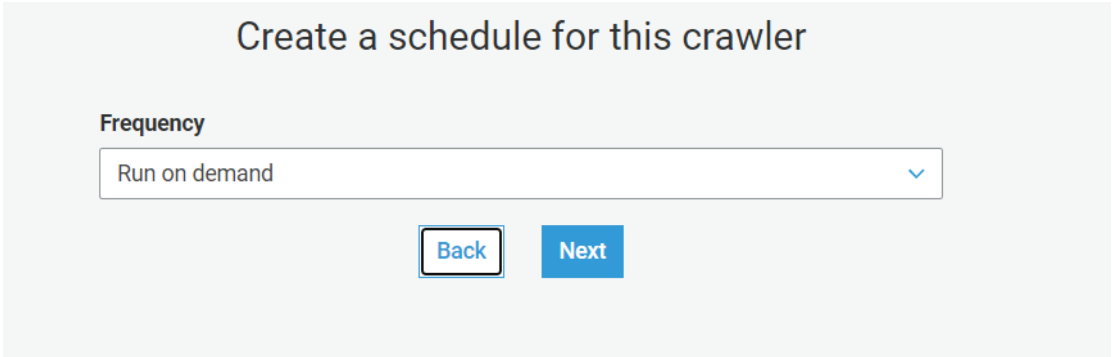
7. On next screen asking to add another data store, keep the default "No" option and click Next



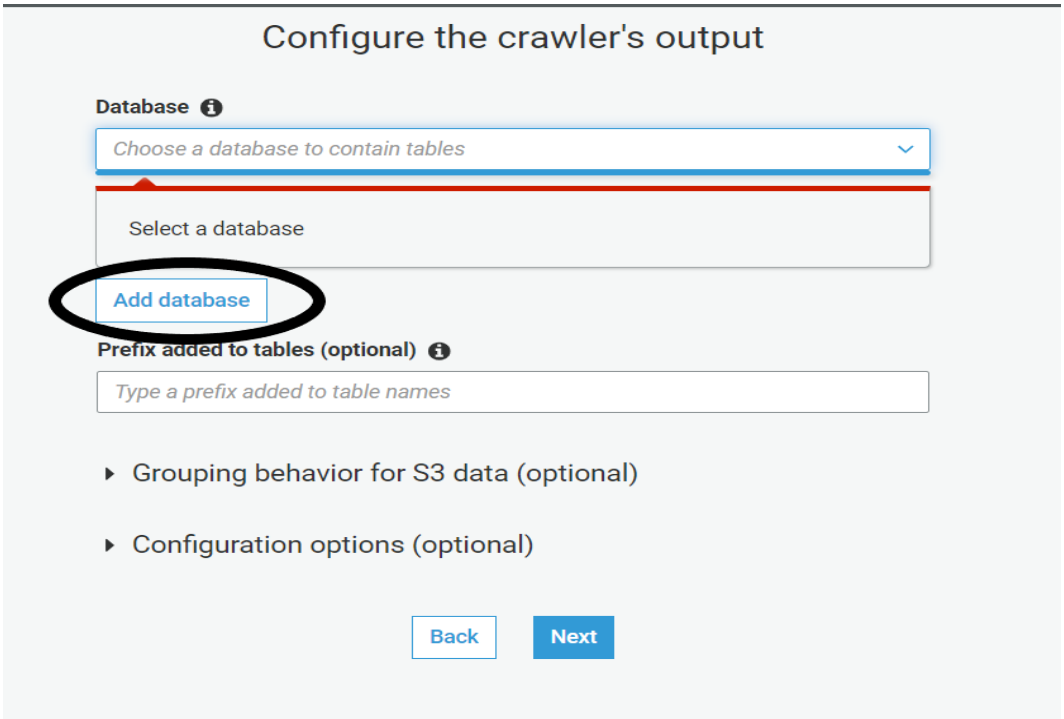
8. On the next screen, select "Create an IAM role" from the options available and provide a name to this role. Then click Next.



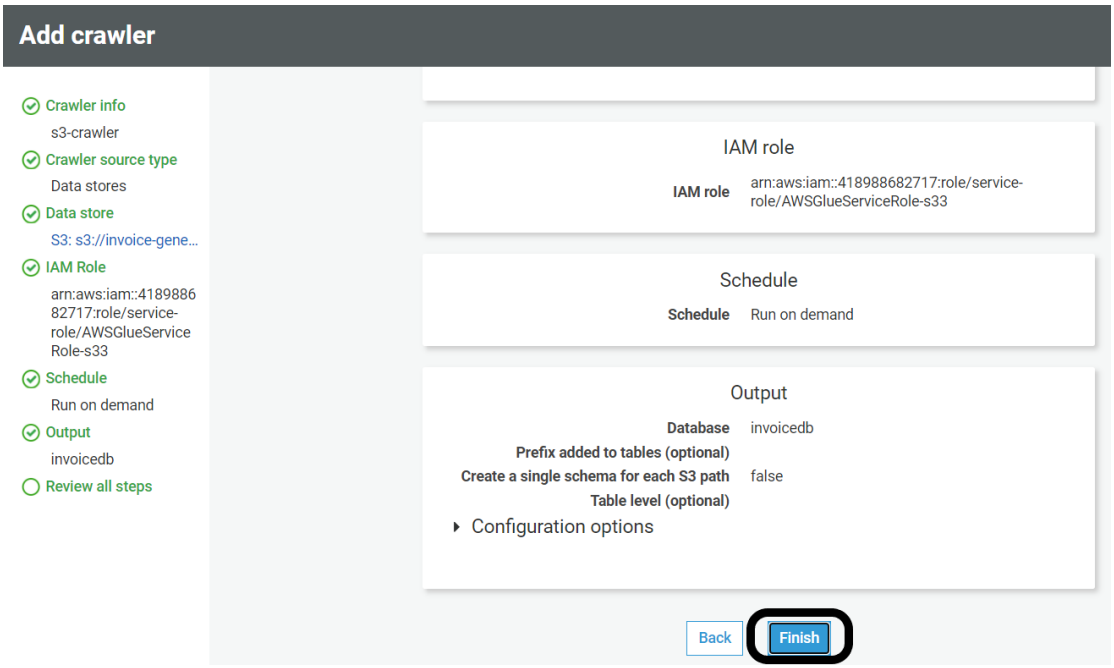
9. On the next screen, we need to define the frequency at which the crawler will crawl S3. For now, let's keep it "Run on demand" and click Next.



10. On the next screen, we need to define a database. Click on "Add database", to open a new prompt. Provide any name for your database and click on Next.

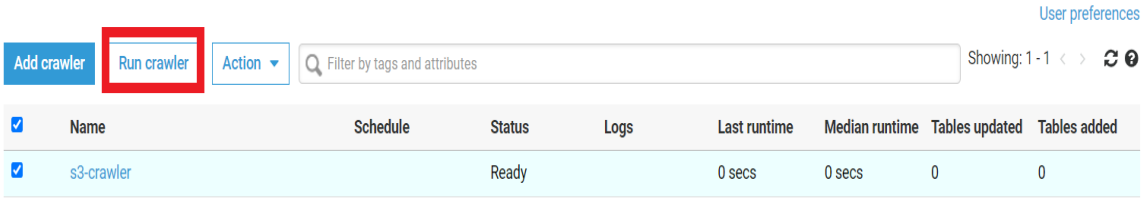


11. On the final screen, click on the “Finish” button to add this Crawler.

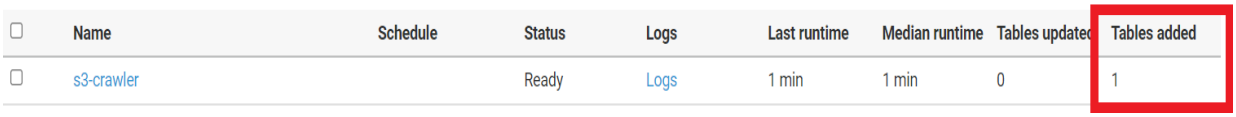


Step 4: Run Crawler

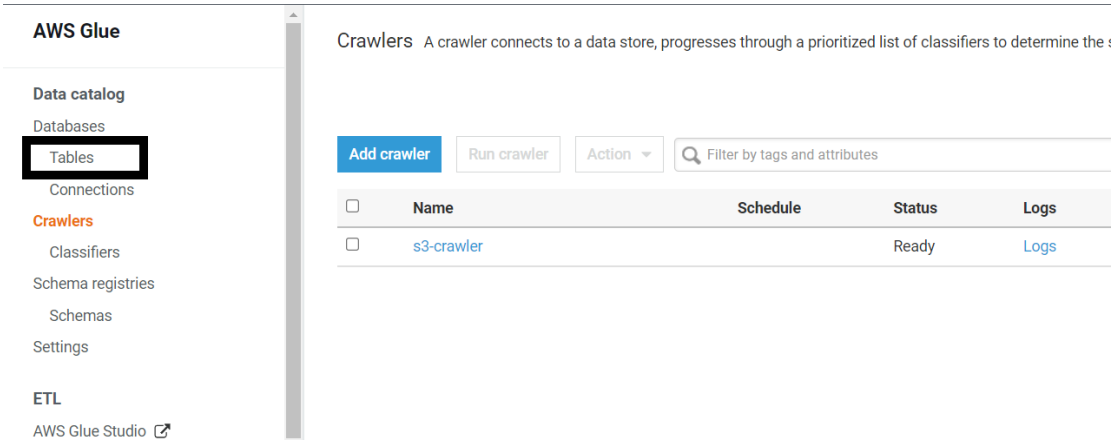
1. Now we have to run our crawler to crawl all folders defined under S3 bucket. This will create a table from the crawler.



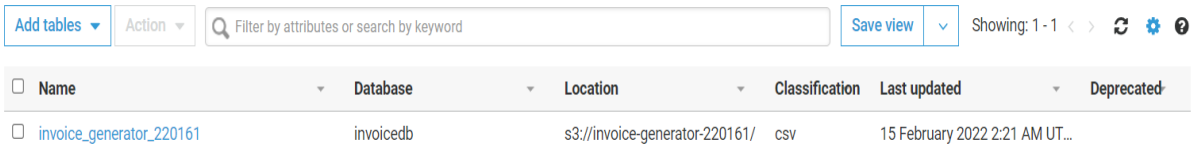
2. After the crawler is successfully executed, you may see 1 under the "Tables added" column.



3. To see what is added in the table, click on the "Tables" option in the left side navigation menu.



4. In the tables screen, you would see a table added. Click on the table to check more details.



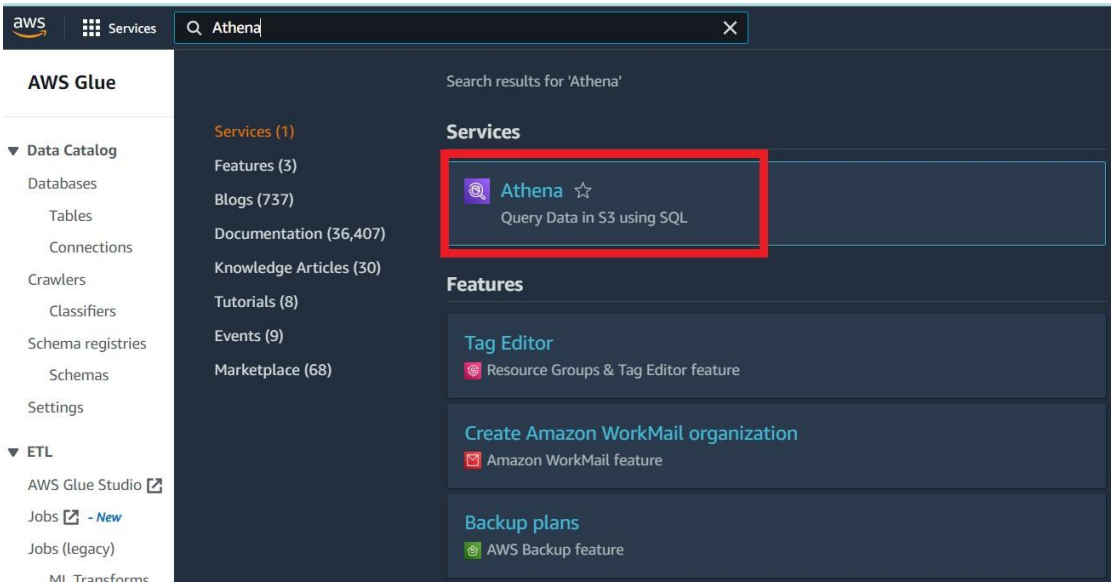
5. Upon clicking the table name , you will see metadata fetched by the crawler from the underlying S3 bucket.

Name	invoice_generator_220161
Description	
Database	invoicedb
Classification	csv
Location	s3://invoice-generator-220161/
Connection	
Deprecated	No
Last updated	Tue Feb 15 02:21:34 GMT+530 2022
Input format	org.apache.hadoop.mapred.TextInputFormat
Output format	org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat
Serde serialization lib	org.apache.hadoop.hive.serde2.lazy.LazySimpleSerDe
Serde parameters	field.delim ,
Table properties	sizeKey **847162** objectCount **1** UPDATED_BY_CRAWLER **s3-crawler** CrawlerSchemaSerializerVersion **1.0** recordCount **9012** averageRecordSize **94** CrawlerSchemaDeserializerVersion **1.0** compressionType **none** columnsOrdered **true** areColumnsQuoted **false** delimiter , typeOfData **file**

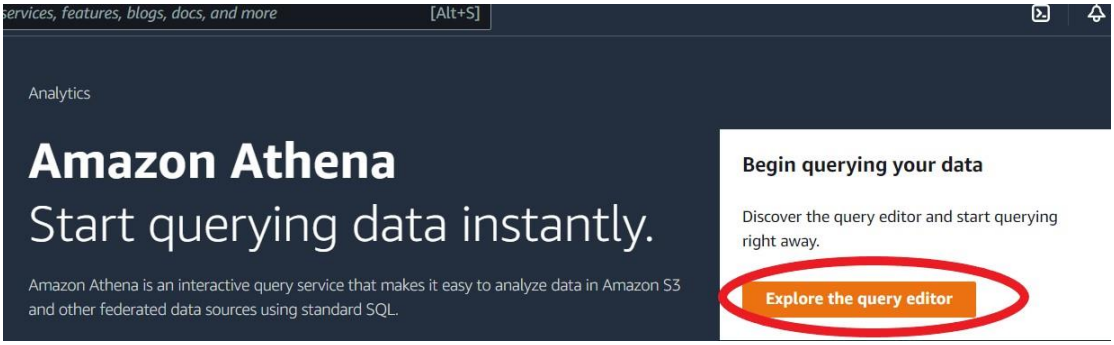
Analytics Using Athena

Amazon Athena is **an interactive query service** that makes it easy to analyze data in Amazon S3 using standard SQL. Athena is serverless, so there is no infrastructure to manage, and you pay only for the queries that you run. This makes it easy for anyone with SQL skills to quickly analyze large-scale datasets.

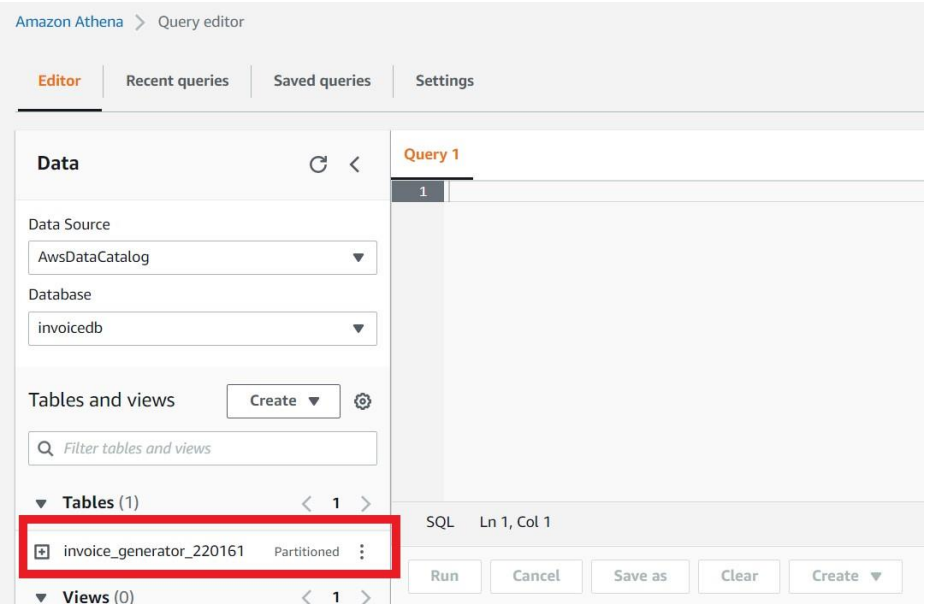
1. Search for "Athena" in Services.



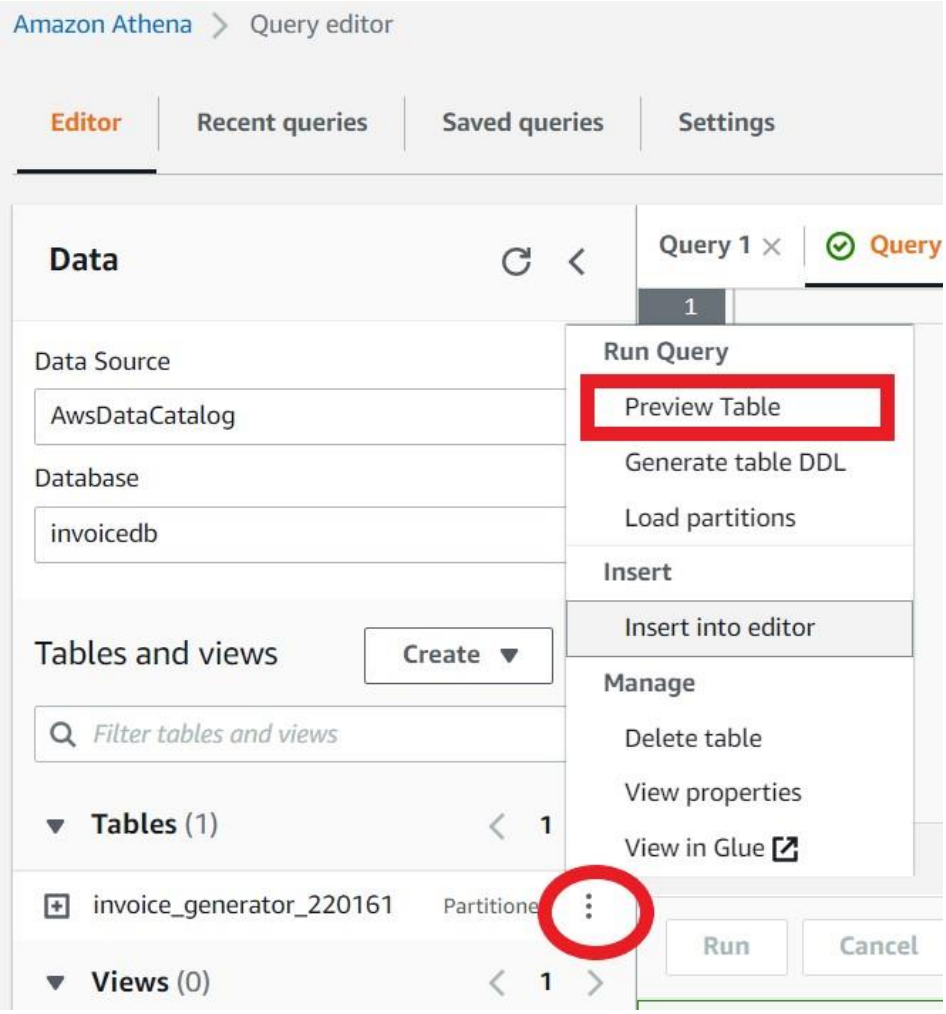
2. Click on the "Explore the query editor" button.



3. You will see the same table listed here that we created using Glue from S3 bucket.

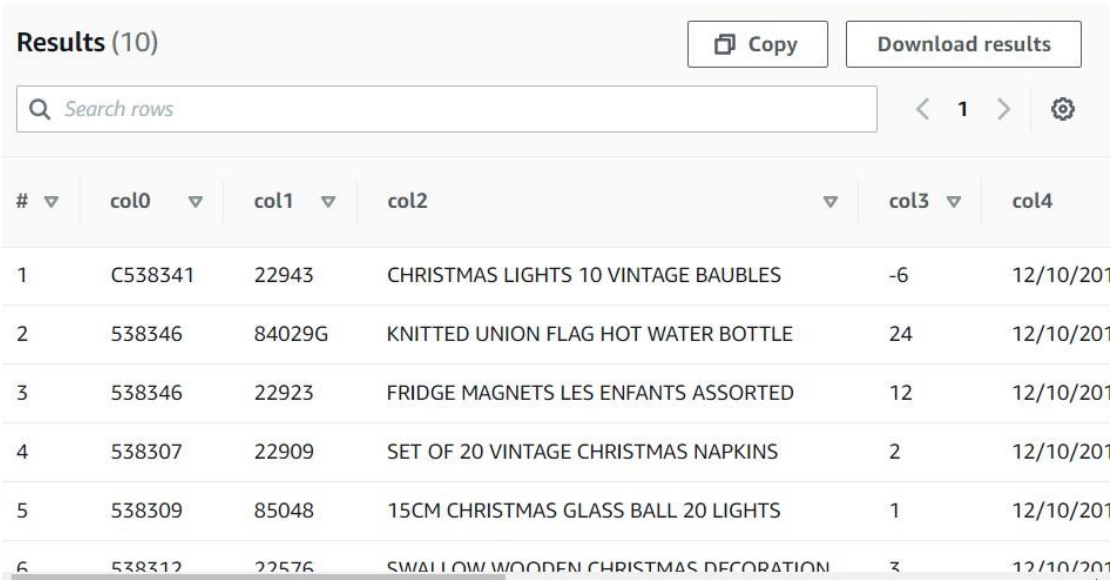


4. Click on the "Preview Table" option to see the contents of the table.



5. It will create a SELECT SQL query to run against the data saved on S3. Click on the "Run again" button to run this query.

6. Wonderful, here are the results.



#	col0	col1	col2	col3	col4
1	C538341	22943	CHRISTMAS LIGHTS 10 VINTAGE BAUBLES	-6	12/10/201
2	538346	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	24	12/10/201
3	538346	22923	FRIDGE MAGNETS LES ENFANTS ASSORTED	12	12/10/201
4	538307	22909	SET OF 20 VINTAGE CHRISTMAS NAPKINS	2	12/10/201
5	538309	85048	15CM CHRISTMAS GLASS BALL 20 LIGHTS	1	12/10/201
6	538312	22576	SWALLOW WOODEN CHRISTMAS DECORATION	3	12/10/201