AMAZON REDSHIFT

ABSTRACT

This document showcases a data pipeline project utilizing Amazon Redshift and Amazon S3. It demonstrates the architecture, management, and analysis of large datasets in the cloud, highlighting practical applications of cloud-based data warehousing and data lake integration.

By:
Kevin Ndungu.

REDSHIFT DATA WAREHOUSE DEMONSTRATION

INTRODUCTION

The project showcases the end-to-end process of creating and populating a database on Amazon Redshift.

This documentation provides insights into the setup of a cloud environment, including the creation of a VPC, subnets, S3 bucket, IAM roles, security groups, and Redshift cluster. It covers the generation of sample data, formatting it, and loading it into the Redshift database. Additionally, for those interested in replicating or examining the infrastructure setup, I offer an Infrastructure as Code (**IaC**) option click **here**. This project demonstrates the implementation of a robust data pipeline leveraging AWS services to efficiently manage and analyze datasets.

This demonstration walks through getting started with Amazon Redshift cluster and shows how to both load and query your data.

NB: For those interested in deploying this project kindly create an **AWS account** if you don't have one before proceeding.

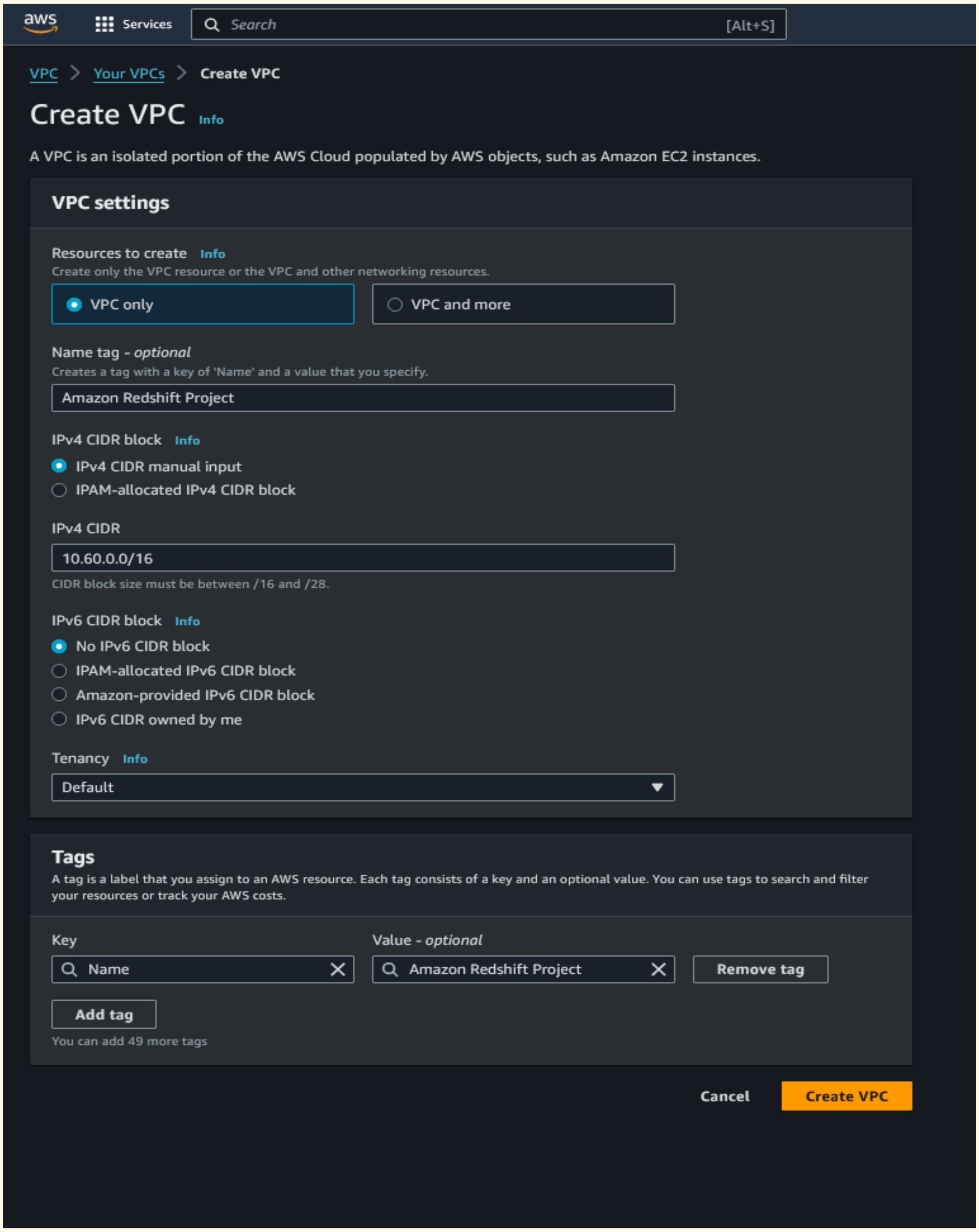
PREPARING THE ENVIRONMENT

A). Create a VPC.

The first step in configuring your environment is to create a virtual private cloud (VPC) to hold the resources for both your Amazon Elastic Compute Cloud (Amazon EC2) instance and Amazon Redshift database. (*The N.Virginia us-east-1 Region was used in this demonstration*).

In this project, the following settings were used:

- *Name:* Amazon Redshift Project
- *Location:* us-east-1
- *IPv4 CIDR block:* 10.60.0.0/16
- *Internet gateway:* Redshift IGW
- *Public route table:* Redshift Project Public
- *Private route table:* Redshift Project Private



Create VPC

A VPC is an isolated portion of the AWS Cloud populated by AWS objects, such as Amazon EC2 instances.

VPC settings

Resources to create Info
Create only the VPC resource or the VPC and other networking resources.
● VPC only ○ VPC and more

Name tag - *optional*
Creates a tag with a key of 'Name' and a value that you specify.
Amazon Redshift Project

IPv4 CIDR block Info
● IPv4 CIDR manual input
○ IPAM-allocated IPv4 CIDR block

IPv4 CIDR
10.60.0.0/16
CIDR block size must be between /16 and /28.

IPv6 CIDR block Info
● No IPv6 CIDR block
○ IPAM-allocated IPv6 CIDR block
○ Amazon-provided IPv6 CIDR block
○ IPv6 CIDR owned by me

Tenancy Info
Default ▼

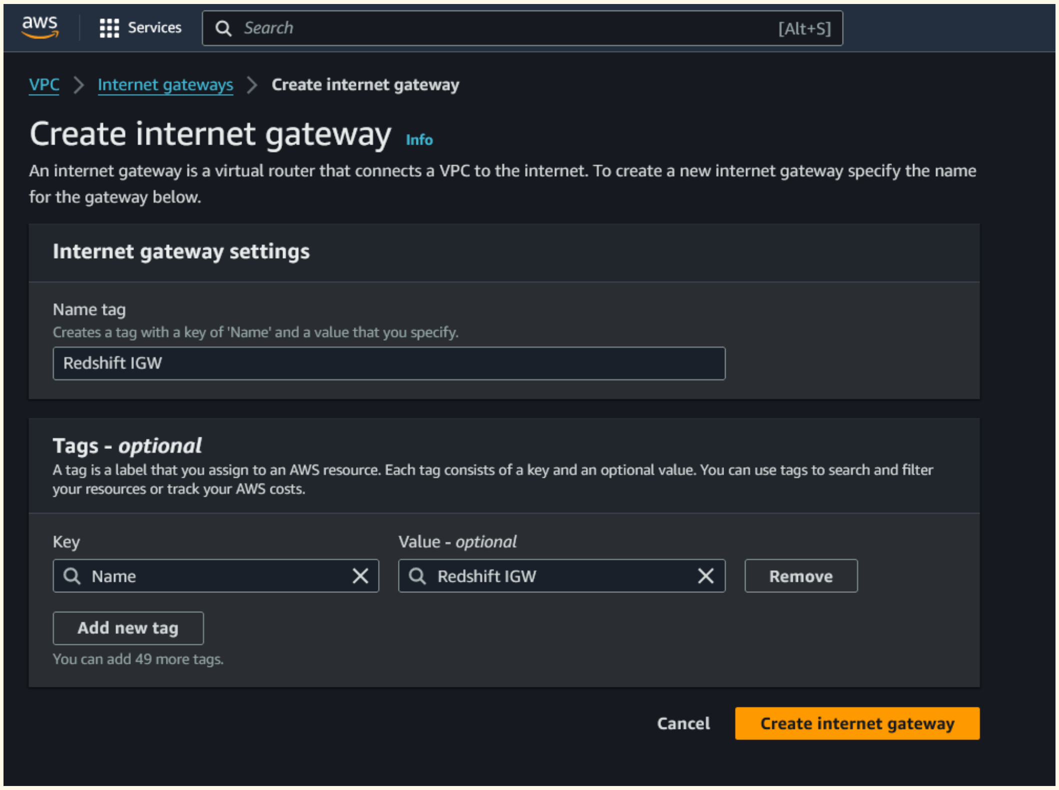
Tags

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - *optional*
Name	Amazon Redshift Project

You can add 49 more tags

Figure 1. VPC Creation - After review click the 'create VPC' button.



Create internet gateway

An internet gateway is a virtual router that connects a VPC to the internet. To create a new internet gateway specify the name for the gateway below.

Internet gateway settings

Name tag
Creates a tag with a key of 'Name' and a value that you specify.
Redshift IGW

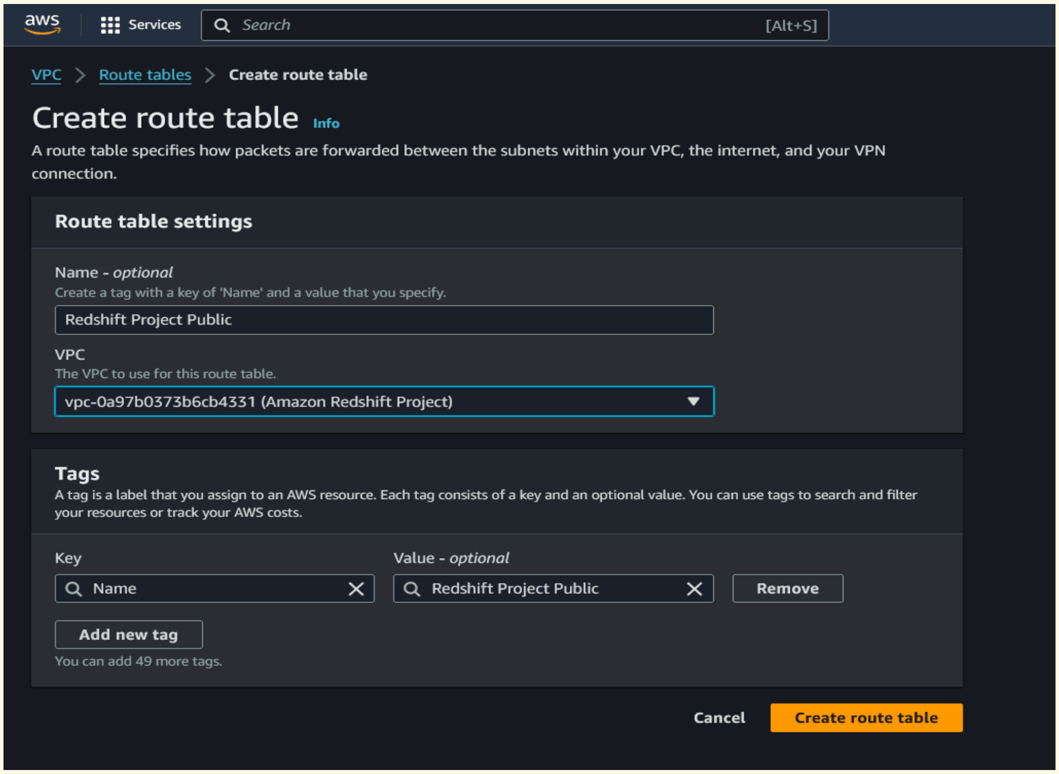
Tags - *optional*

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - *optional*
Name	Redshift IGW

You can add 49 more tags.

Figure 2. Internet Gateway creation.



Create route table

A route table specifies how packets are forwarded between the subnets within your VPC, the internet, and your VPN connection.

Route table settings

Name - *optional*
Create a tag with a key of 'Name' and a value that you specify.
Redshift Project Public

VPC
The VPC to use for this route table.
vpc-0a97b0373b6cb4331 (Amazon Redshift Project)

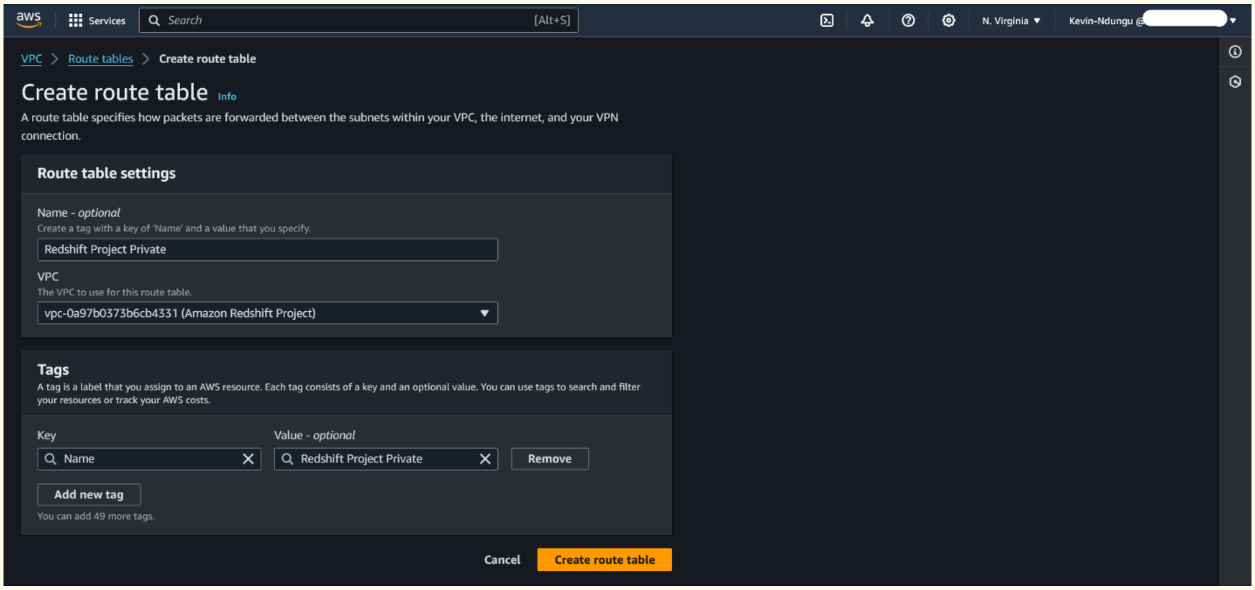
Tags

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - *optional*
Name	Redshift Project Public

You can add 49 more tags.

Figure 3. Public Route Table, click on the 'create route' table button.



VPC > Route tables > Create route table

Create route table Info

A route table specifies how packets are forwarded between the subnets within your VPC, the internet, and your VPN connection.

Route table settings

Name - *optional*
Create a tag with a key of 'Name' and a value that you specify.
Redshift Project Private

VPC
The VPC to use for this route table.
vpc-0a97b0373b6cb4331 (Amazon Redshift Project)

Tags
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - *optional*	
Name	Redshift Project Private	Remove

Add new tag
You can add 49 more tags.

Cancel　Create route table

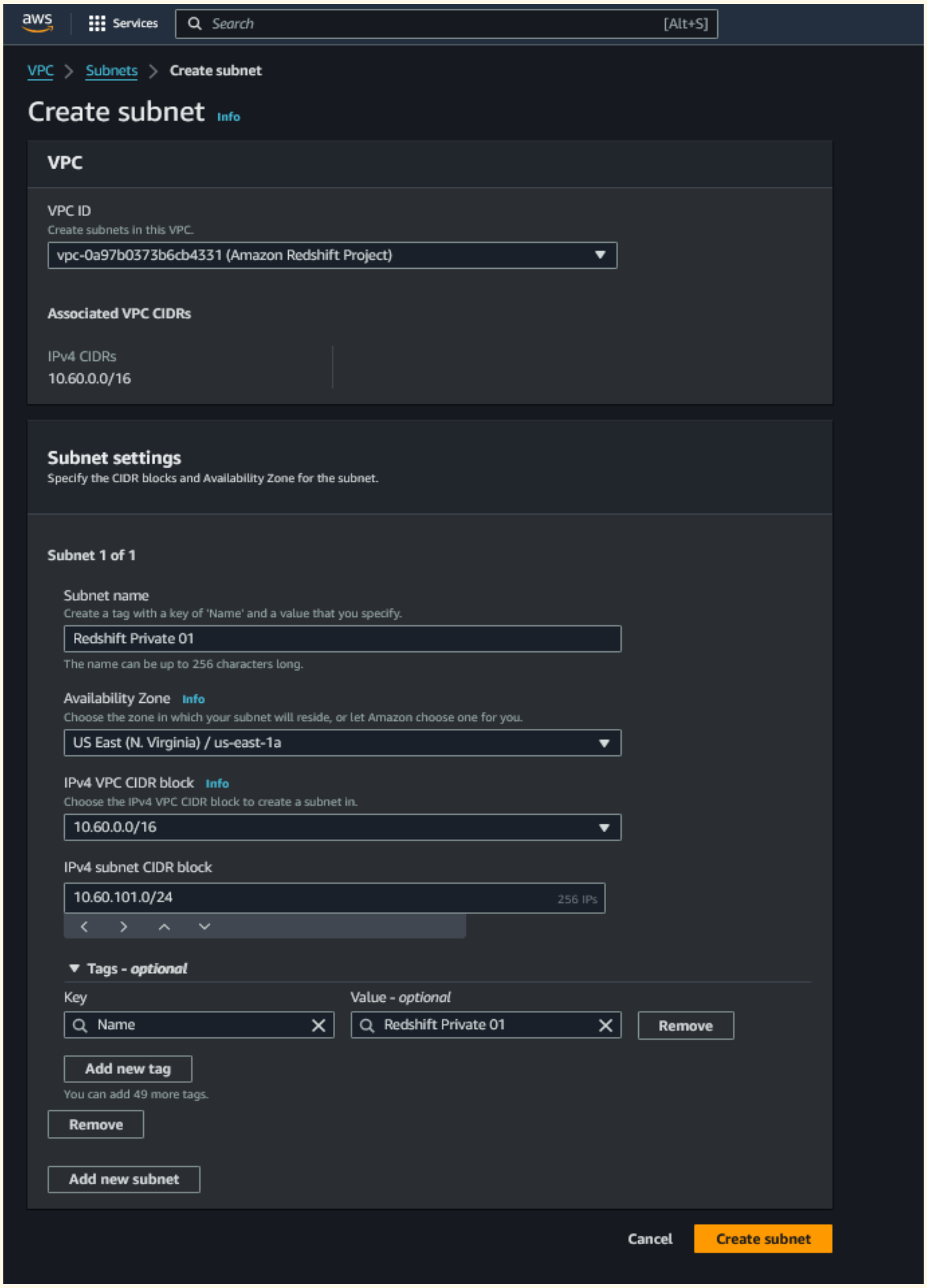
Figure 4. Private Route Table, click on the 'create route table' button.

B). Create the subnets.

Amazon Redshift is a managed service that connects to subnets in your VPC.

In this project, the following settings were used:

- **Private subnet 1**
    - *Name:* Redshift Private 01
    - *CIDR:* 10.60.101.0/24



Create subnet Info

VPC

VPC ID
Create subnets in this VPC.
vpc-0a97b0373b6cb4331 (Amazon Redshift Project)

Associated VPC CIDRs

IPv4 CIDRs
10.60.0.0/16

Subnet settings
Specify the CIDR blocks and Availability Zone for the subnet.

Subnet 1 of 1

Subnet name
Create a tag with a key of 'Name' and a value that you specify.
Redshift Private 01
The name can be up to 256 characters long.

Availability Zone Info
Choose the zone in which your subnet will reside, or let Amazon choose one for you.
US East (N. Virginia) / us-east-1a

IPv4 VPC CIDR block Info
Choose the IPv4 VPC CIDR block to create a subnet in.
10.60.0.0/16

IPv4 subnet CIDR block
10.60.101.0/24 256 IPs

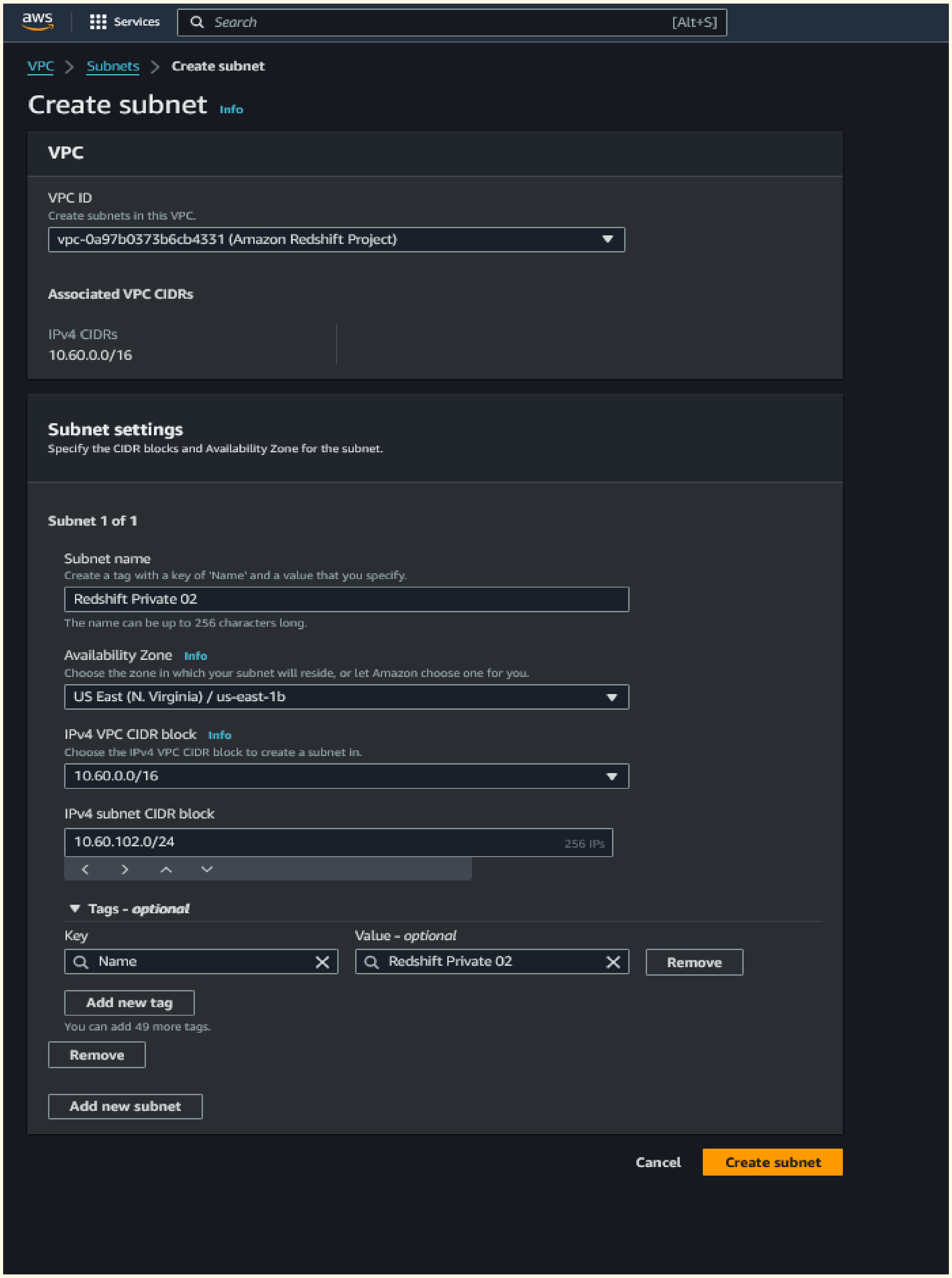
Tags - *optional*

Key	Value - *optional*	
Name	Redshift Private 01	Remove

Add new tag
You can add 49 more tags.

Figure 5. Click on the 'create subnet' button.

- **Private subnet 2**
    - *Name:* Redshift Private 02
    - *CIDR:* 10.60.102.0/24



Create subnet Info

VPC

VPC ID
Create subnets in this VPC.
vpc-0a97b0373b6cb4331 (Amazon Redshift Project)

Associated VPC CIDRs

IPv4 CIDRs
10.60.0.0/16

Subnet settings
Specify the CIDR blocks and Availability Zone for the subnet.

Subnet 1 of 1

Subnet name
Create a tag with a key of 'Name' and a value that you specify.
Redshift Private 02
The name can be up to 256 characters long.

Availability Zone Info
Choose the zone in which your subnet will reside, or let Amazon choose one for you.
US East (N. Virginia) / us-east-1b

IPv4 VPC CIDR block Info
Choose the IPv4 VPC CIDR block to create a subnet in.
10.60.0.0/16

IPv4 subnet CIDR block
10.60.102.0/24 256 IPs

▼ Tags - *optional*

Key	Value - *optional*
Name	Redshift Private 02

You can add 49 more tags.

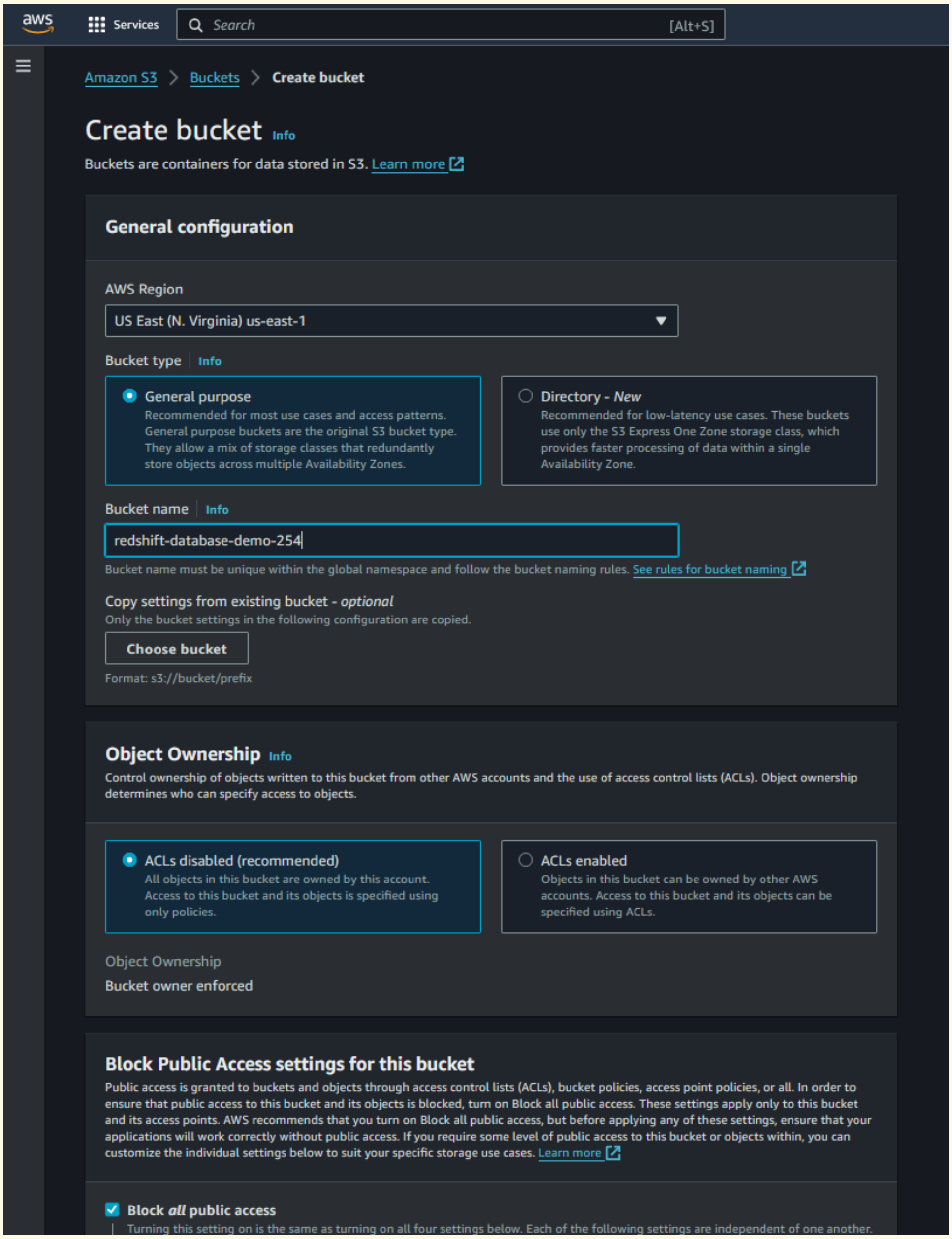
Figure 6. Click on the 'create subnet' button.

C). Create an Amazon S3 bucket.

The data used for this demo is randomly generated and uploaded to an Amazon Simple Storage Service (Amazon S3) bucket.

In this project, the following settings were used:

- *Region:* us-east-1
- *Bucket:* redshift-database-demo *(Make sure to use a unique bucket name)*



Create bucket Info

Buckets are containers for data stored in S3. Learn more

General configuration

AWS Region

US East (N. Virginia) us-east-1

Bucket type Info

● General purpose
Recommended for most use cases and access patterns. General purpose buckets are the original S3 bucket type. They allow a mix of storage classes that redundantly store objects across multiple Availability Zones.

○ Directory - *New*
Recommended for low-latency use cases. These buckets use only the S3 Express One Zone storage class, which provides faster processing of data within a single Availability Zone.

Bucket name Info

redshift-database-demo-254

Bucket name must be unique within the global namespace and follow the bucket naming rules. See rules for bucket naming

Copy settings from existing bucket - *optional*
Only the bucket settings in the following configuration are copied.

Choose bucket

Format: s3://bucket/prefix

Object Ownership Info

Control ownership of objects written to this bucket from other AWS accounts and the use of access control lists (ACLs). Object ownership determines who can specify access to objects.

● ACLs disabled (recommended)
All objects in this bucket are owned by this account. Access to this bucket and its objects is specified using only policies.

○ ACLs enabled
Objects in this bucket can be owned by other AWS accounts. Access to this bucket and its objects can be specified using ACLs.

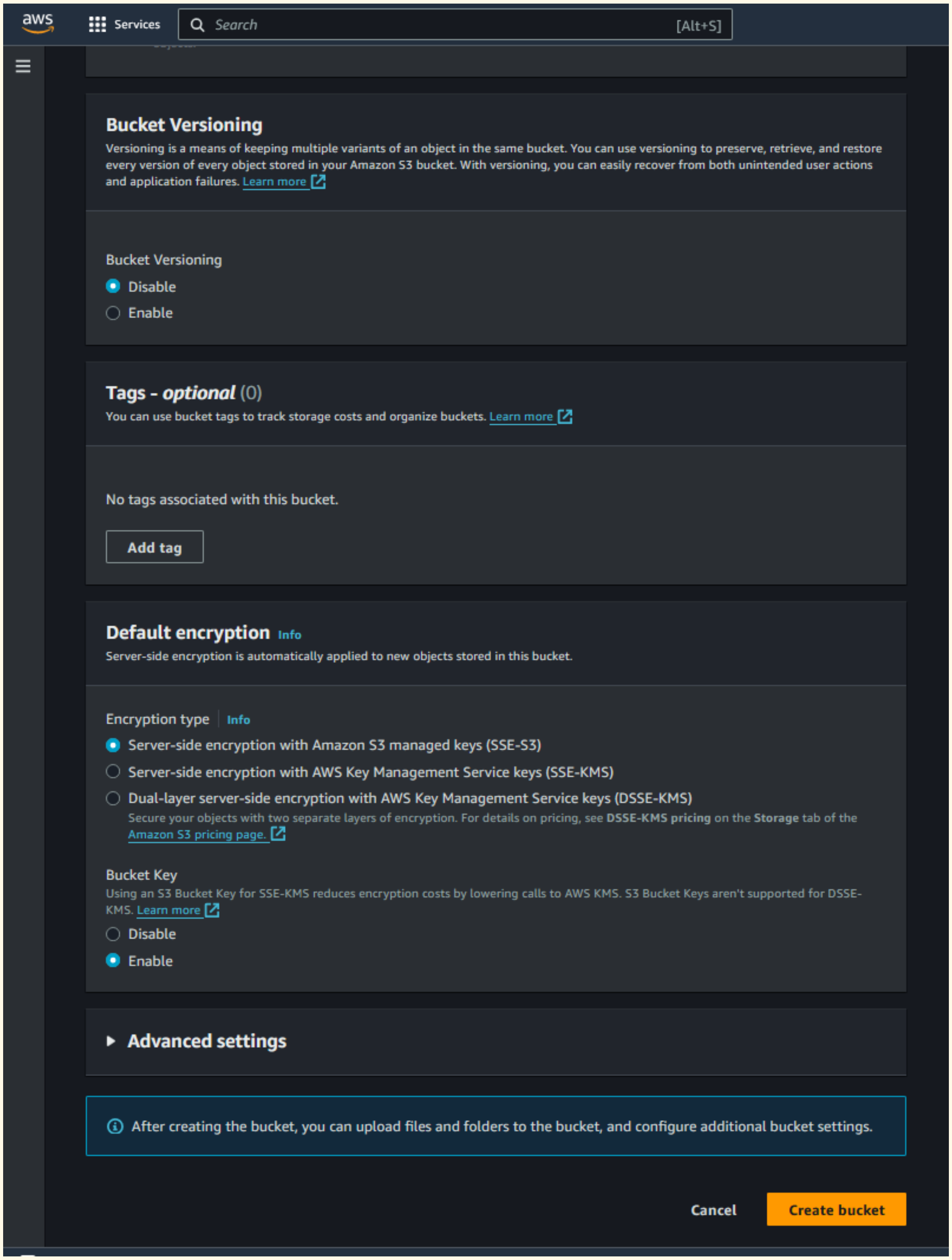
Object Ownership
Bucket owner enforced

Block Public Access settings for this bucket

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to this bucket and its objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to this bucket or objects within, you can customize the individual settings below to suit your specific storage use cases. Learn more

☑ Block *all* public access
Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

Figure 7.



Bucket Versioning

Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your Amazon S3 bucket. With versioning, you can easily recover from both unintended user actions and application failures. Learn more

Bucket Versioning

● Disable

○ Enable

Tags - *optional* (0)

You can use bucket tags to track storage costs and organize buckets. Learn more

No tags associated with this bucket.

Add tag

Default encryption Info

Server-side encryption is automatically applied to new objects stored in this bucket.

Encryption type | Info

● Server-side encryption with Amazon S3 managed keys (SSE-S3)

○ Server-side encryption with AWS Key Management Service keys (SSE-KMS)

○ Dual-layer server-side encryption with AWS Key Management Service keys (DSSE-KMS)
Secure your objects with two separate layers of encryption. For details on pricing, see **DSSE-KMS pricing** on the **Storage** tab of the Amazon S3 pricing page.

Bucket Key

Using an S3 Bucket Key for SSE-KMS reduces encryption costs by lowering calls to AWS KMS. S3 Bucket Keys aren't supported for DSSE-KMS. Learn more

○ Disable

● Enable

▸ **Advanced settings**

ⓘ After creating the bucket, you can upload files and folders to the bucket, and configure additional bucket settings.

Cancel | Create bucket

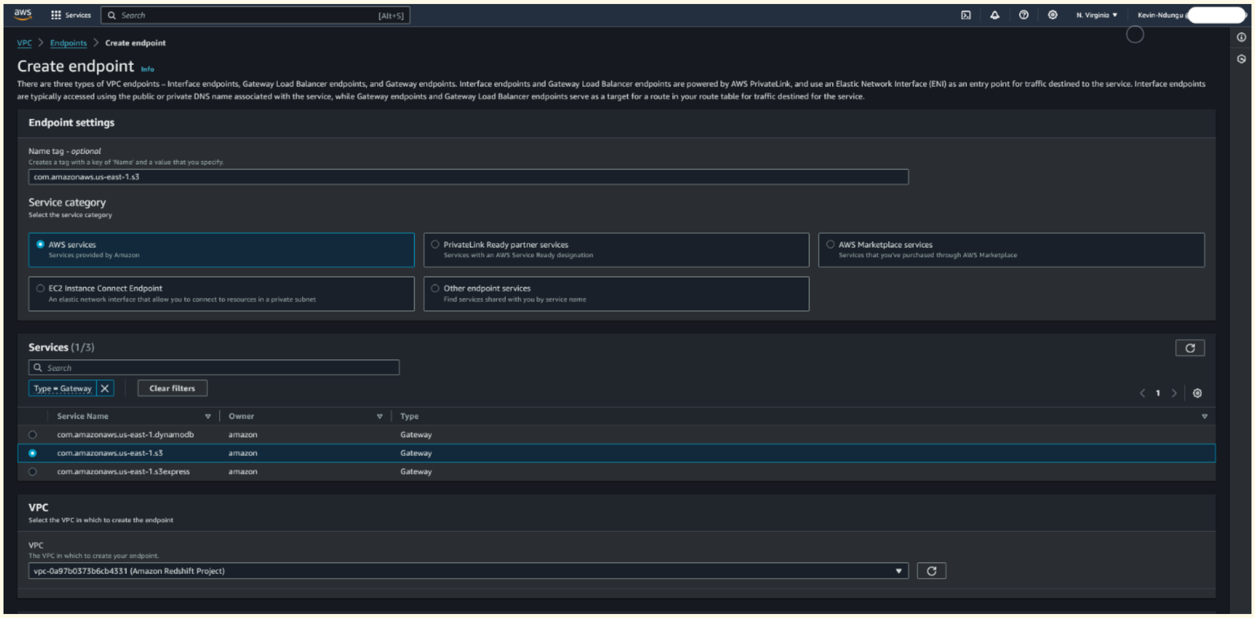
Figure 8.

D). Create an Amazon S3 endpoint.

By default, traffic between an Amazon Redshift cluster and Amazon S3 traverses the public AWS network. A VPC endpoint forces all COPY and UNLOAD traffic between your cluster and your data on Amazon S3 to stay in your VPC.

In this project, the following settings were used:

- *Service name:* com.amazonaws.us-east-1.s3
- *Endpoint type:* Gateway
- *Route table:* Redshift Project Private



Create endpoint

There are three types of VPC endpoints – Interface endpoints, Gateway Load Balancer endpoints, and Gateway endpoints. Interface endpoints and Gateway Load Balancer endpoints are powered by AWS PrivateLink, and use an Elastic Network Interface (ENI) as an entry point for traffic destined to the service. Interface endpoints are typically accessed using the public or private DNS name associated with the service, while Gateway endpoints and Gateway Load Balancer endpoints serve as a target for a route in your route table for traffic destined for the service.

Endpoint settings

Name tag - *optional*
Creates a tag with a key of 'Name' and a value that you specify.
com.amazonaws.us-east-1.s3

Service category
Select the service category

- ● AWS services — Services provided by Amazon
- ○ PrivateLink Ready partner services — Services with an AWS Service Ready designation
- ○ AWS Marketplace services — Services that you've purchased through AWS Marketplace
- ○ EC2 Instance Connect Endpoint — An elastic network interface that allow you to connect to resources in a private subnet
- ○ Other endpoint services — Find services shared with you by service name

Services (1/3)

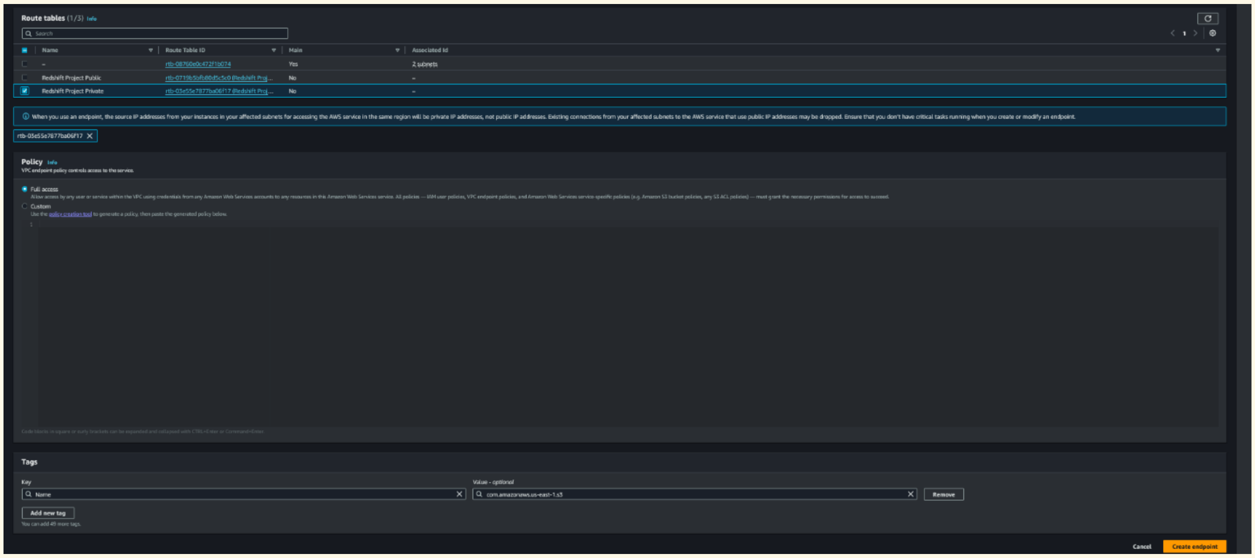
Type = Gateway

	Service Name	Owner	Type
○	com.amazonaws.us-east-1.dynamodb	amazon	Gateway
●	com.amazonaws.us-east-1.s3	amazon	Gateway
○	com.amazonaws.us-east-1.s3express	amazon	Gateway

VPC
Select the VPC in which to create the endpoint

VPC
The VPC in which to create your endpoint.
vpc-0a97b0373b6cb4331 (Amazon Redshift Project)

Figure 9. Part A.



Route tables (1/3)

	Name	Route Table ID	Main	Associated Id
☐	-	rtb-08760e0c472f1b074	Yes	2 subnets
☐	Redshift Project Public	rtb-0719b5bfb80d5c5c0 (Redshift Proj...	No	-
☑	Redshift Project Private	rtb-03e55e7877ba06f17 (Redshift Proj...	No	-

ⓘ When you use an endpoint, the source IP addresses from your instances in your affected subnets for accessing the AWS service in the same region will be private IP addresses, not public IP addresses. Existing connections from your affected subnets to the AWS service that use public IP addresses may be dropped. Ensure that you don't have critical tasks running when you create or modify an endpoint.

rtb-03e55e7877ba06f17

Policy
VPC endpoint policy controls access to the service.

- ● Full access — Allow access by any user or service within the VPC using credentials from any Amazon Web Services accounts to any resources in this Amazon Web Services service. All policies — IAM user policies, VPC endpoint policies, and Amazon Web Services service-specific policies (e.g. Amazon S3 bucket policies, any S3 ACL policies) — must grant the necessary permissions for access to succeed.
- ○ Custom — Use the policy creation tool to generate a policy, then paste the generated policy below.

Code blocks in square or curly brackets can be expanded and collapsed with CTRL+Enter or Command+Enter.

Tags

Key	Value - optional
Name	com.amazonaws.us-east-1.s3

You can add 49 more tags.

Figure 10. Click the Create endpoint button.

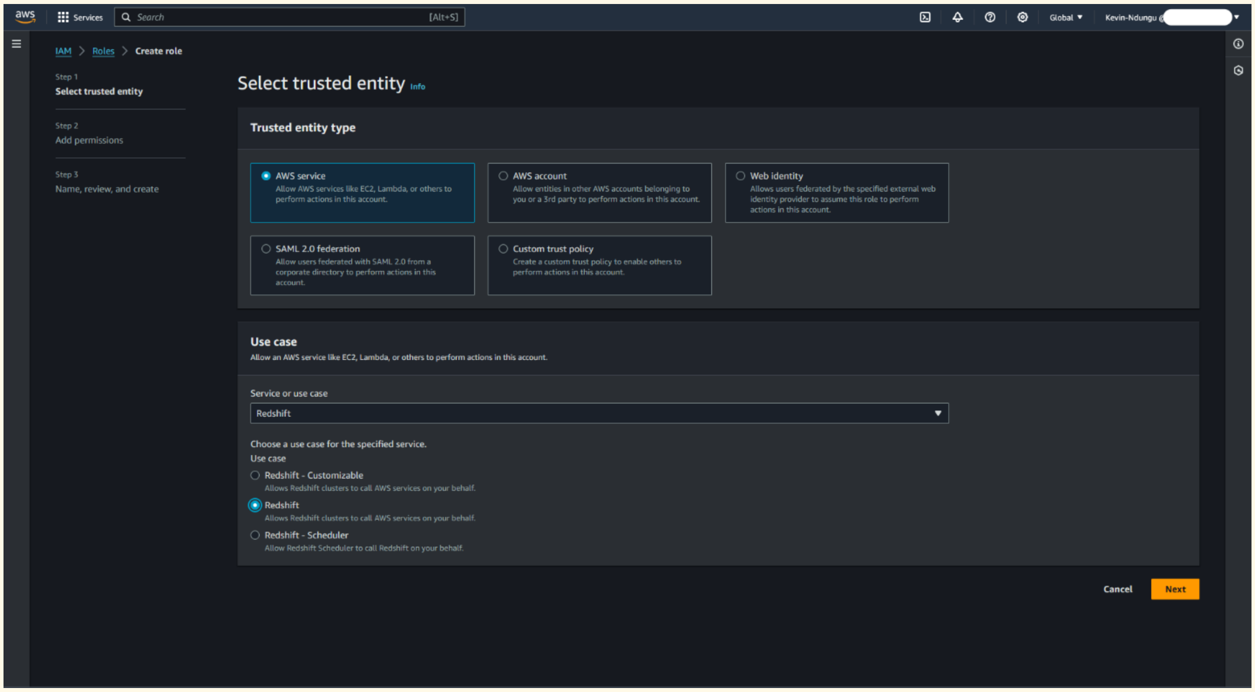
NB: The Amazon Redshift cluster must be in the same Region as the S3 bucket.

E). Create an IAM role.

Amazon Redshift needs permission to copy data to and from S3 buckets. This access is granted using AWS Identity and Access Management (IAM) roles. A role defines a set of permissions for making AWS service requests.

In this project, the following settings were used:

- *Role name:* RedshiftDemoProject
- *Policy name:* RedshiftDemoProject



Step 1
Select trusted entity

Step 2
Add permissions

Step 3
Name, review, and create

Select trusted entity

Trusted entity type

● AWS service
Allow AWS services like EC2, Lambda, or others to perform actions in this account.

○ AWS account
Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.

○ Web identity
Allows users federated by the specified external web identity provider to assume this role to perform actions in this account.

○ SAML 2.0 federation
Allow users federated with SAML 2.0 from a corporate directory to perform actions in this account.

○ Custom trust policy
Create a custom trust policy to enable others to perform actions in this account.

Use case
Allow an AWS service like EC2, Lambda, or others to perform actions in this account.

Service or use case
Redshift ▾

Choose a use case for the specified service.
Use case

○ Redshift - Customizable
Allows Redshift clusters to call AWS services on your behalf.

● Redshift
Allows Redshift clusters to call AWS services on your behalf.

○ Redshift - Scheduler
Allow Redshift Scheduler to call Redshift on your behalf.

Figure 11.

In this example, the role and policy names are the same. This was for ease of use. They can be different.

The policy gives Amazon Redshift full access to a specific S3 bucket, redshift-database-demo, and the objects inside it.

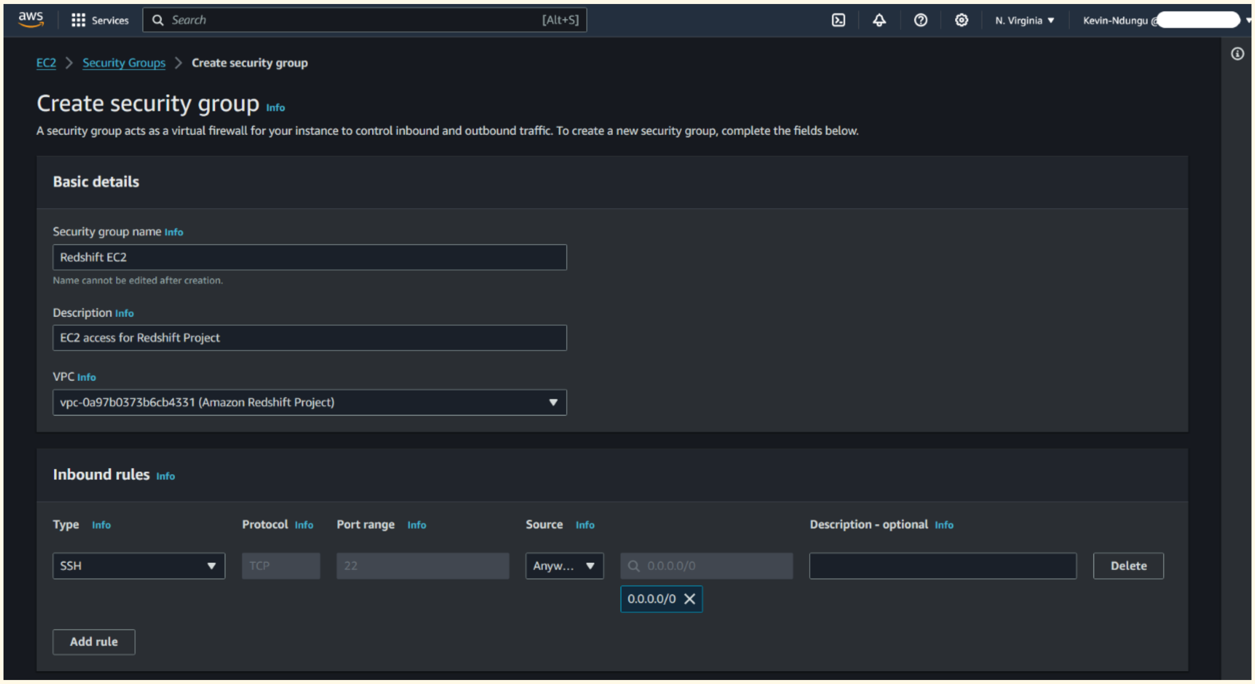
It also has a trust relationship with redshift.amazonaws.com. The trust relationship defines who/what can assume a role. Because of this trust relationship, only Amazon Redshift can assume this role, and it has to be explicitly assigned to the cluster.

F). Create the security groups.

Security groups control access to EC2 instances and Amazon Redshift clusters. In this project, two security groups were created: one is for an EC2 instance and the other for the Amazon Redshift cluster.

In this project, the following settings were used:

- **Group 1**
  - *Name:* Redshift EC2
  - *Description:* EC2 access for Redshift Project
  - *Rule:* Allow SSH (port 22) from 0.0.0.0/0 (the internet)


aws
Services
Search
[Alt+S]
N. Virginia
Kevin-Ndungu @ [redacted]
EC2 > Security Groups > Create security group
Info
Add rule
Delete

Create security group

A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. To create a new security group, complete the fields below.

Basic details

Security group name: Redshift EC2

Name cannot be edited after creation.

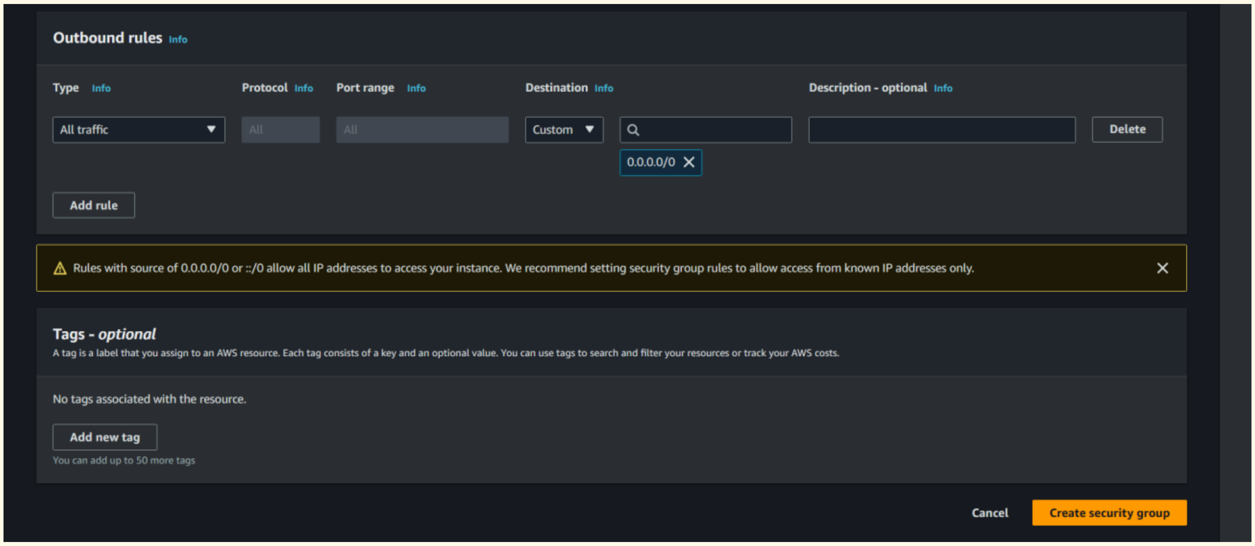
Description: EC2 access for Redshift Project

VPC: vpc-0a97b0373b6cb4331 (Amazon Redshift Project)

Inbound rules

Type	Protocol	Port range	Source		Description - optional
SSH	TCP	22	Anyw...	0.0.0.0/0	

Figure 12.



Outbound rules Info

Type Info	Protocol Info	Port range Info	Destination Info		Description - optional Info	
All traffic	All	All	Custom	0.0.0.0/0		Delete

Add rule

⚠ Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

Tags - *optional*

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

No tags associated with the resource.

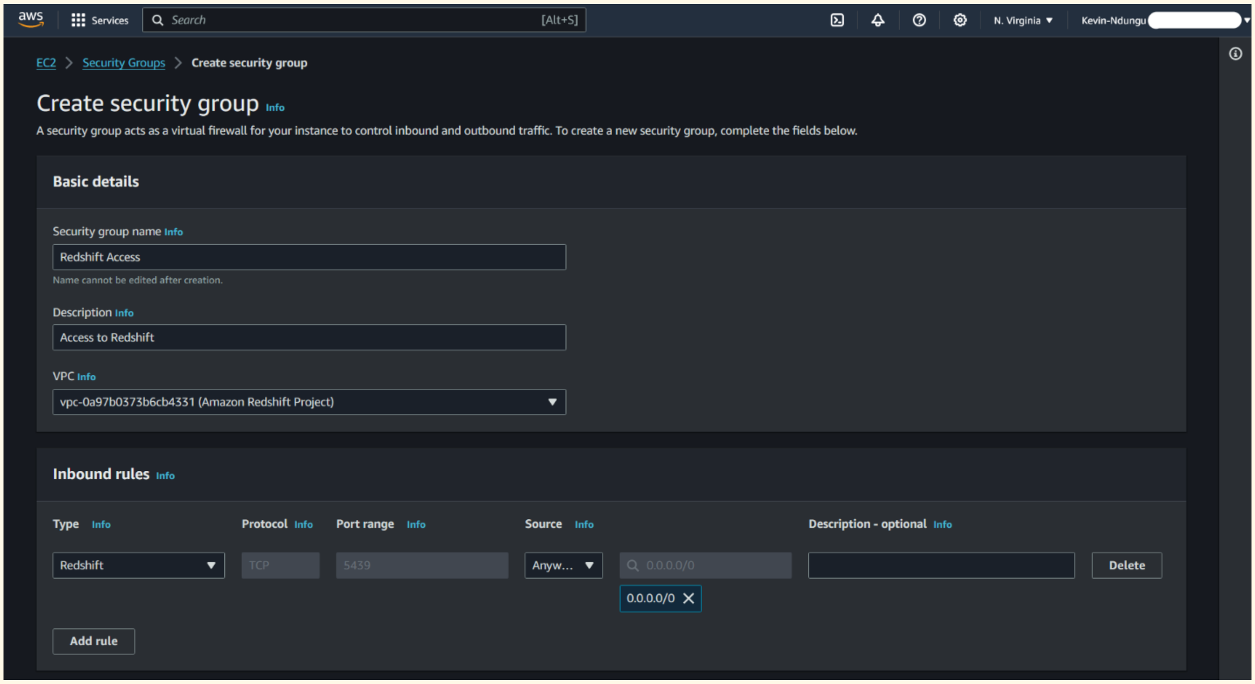
Add new tag

You can add up to 50 more tags

Cancel | Create security group

Figure 13. Click the Create security group button.

- **Group 2**
  - *Name:* Redshift Access
  - *Description:* Access to Redshift
  - *Rule:* Allow TCP port 5439 from the group "Redshift EC2"



EC2 > Security Groups > Create security group

Create security group Info

A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. To create a new security group, complete the fields below.

Basic details

Security group name Info: Redshift Access

Name cannot be edited after creation.

Description Info: Access to Redshift

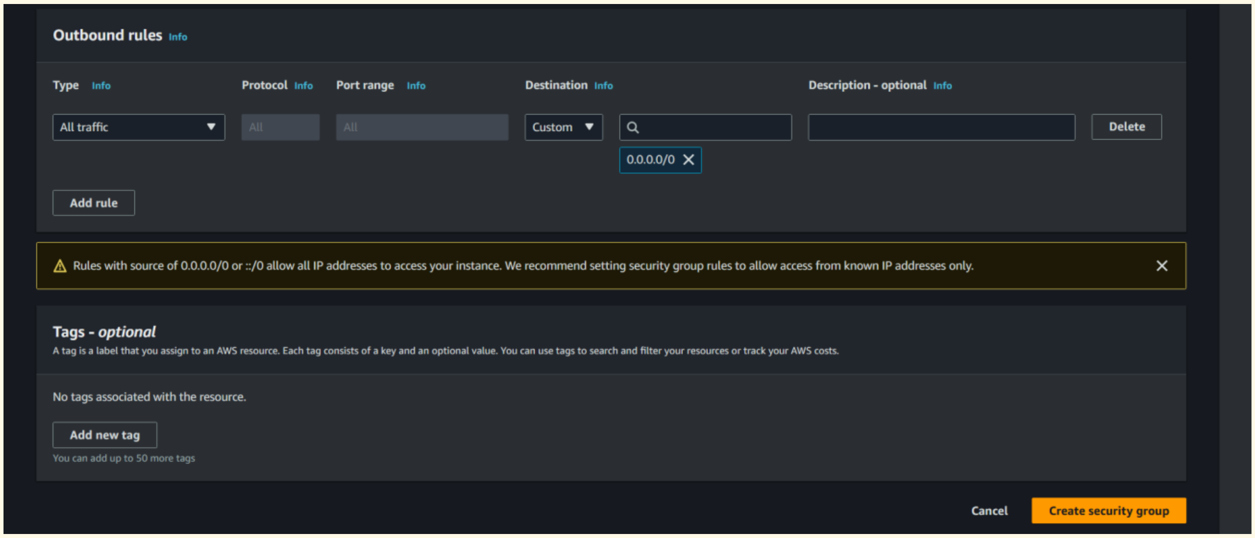
VPC Info: vpc-0a97b0373b6cb4331 (Amazon Redshift Project)

Inbound rules Info

Type Info	Protocol Info	Port range Info	Source Info		Description - optional Info	
Redshift	TCP	5439	Anyw...	0.0.0.0/0		Delete

Add rule

Figure 14.



Type Info	Protocol Info	Port range Info	Destination Info		Description - optional Info
All traffic	All	All	Custom	0.0.0.0/0	

⚠ Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

Tags - *optional*

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

No tags associated with the resource.

You can add up to 50 more tags

Figure 15. Click the Create security group button.

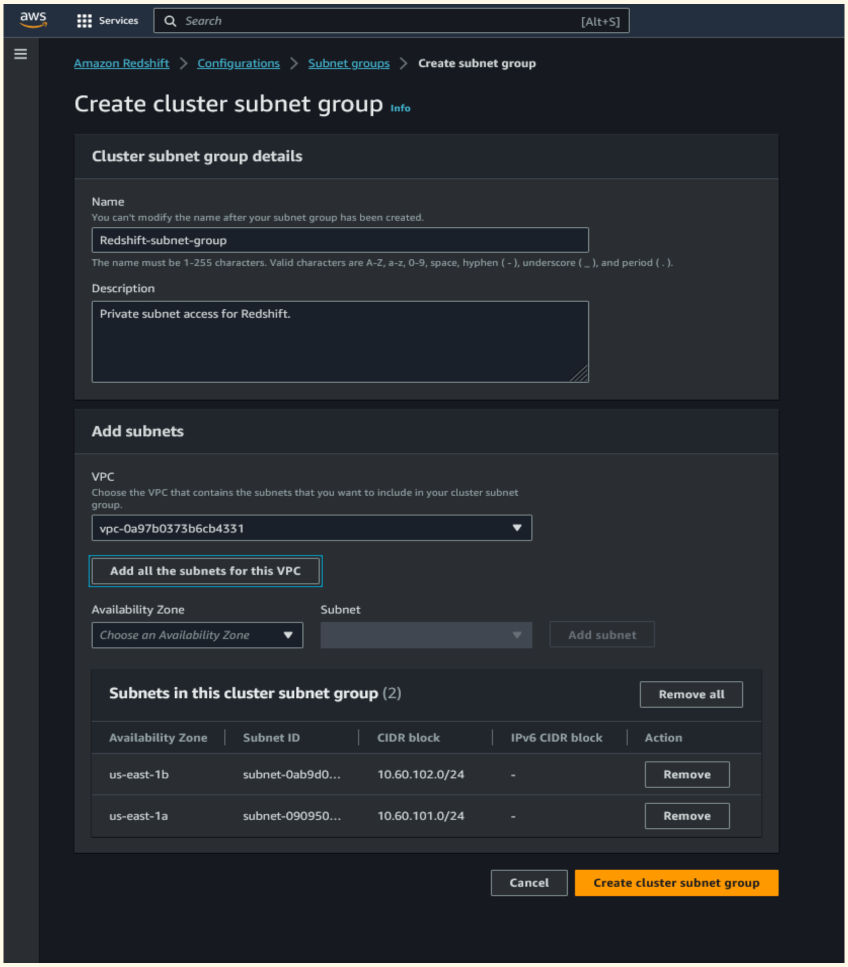
You could use a single group for both EC2 and Amazon Redshift access.

G). Create the subnet group.

It is recommended that you place your Amazon Redshift cluster in a private subnet.

In this project, the following settings were used:

- *Name:* Redshift-subnet-group
- *Description:* Private subnet access for Redshift
- *VPC:* Amazon Redshift Project
- Add all private subnets.



Amazon Redshift > Configurations > Subnet groups > Create subnet group

Create cluster subnet group

Cluster subnet group details

Name
You can't modify the name after your subnet group has been created.
Redshift-subnet-group
The name must be 1-255 characters. Valid characters are A-Z, a-z, 0-9, space, hyphen (-), underscore (_), and period (.).

Description
Private subnet access for Redshift.

Add subnets

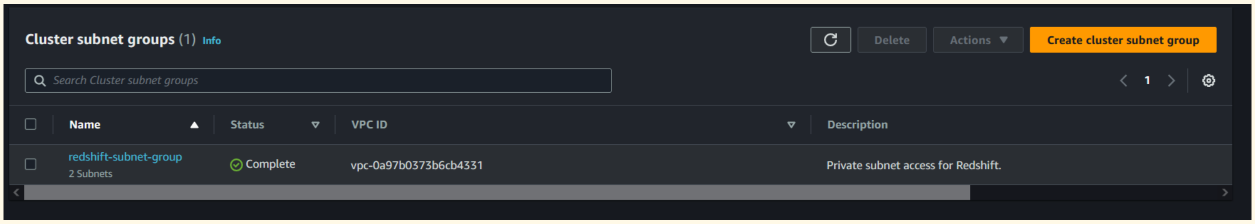
VPC
Choose the VPC that contains the subnets that you want to include in your cluster subnet group.
vpc-0a97b0373b6cb4331

Availability Zone: Choose an Availability Zone
Subnet:

Subnets in this cluster subnet group (2)

Availability Zone	Subnet ID	CIDR block	IPv6 CIDR block	Action
us-east-1b	subnet-0ab9d0...	10.60.102.0/24	-	Remove
us-east-1a	subnet-090950...	10.60.101.0/24	-	Remove

Figure 16. Click the Create cluster subnet group button.

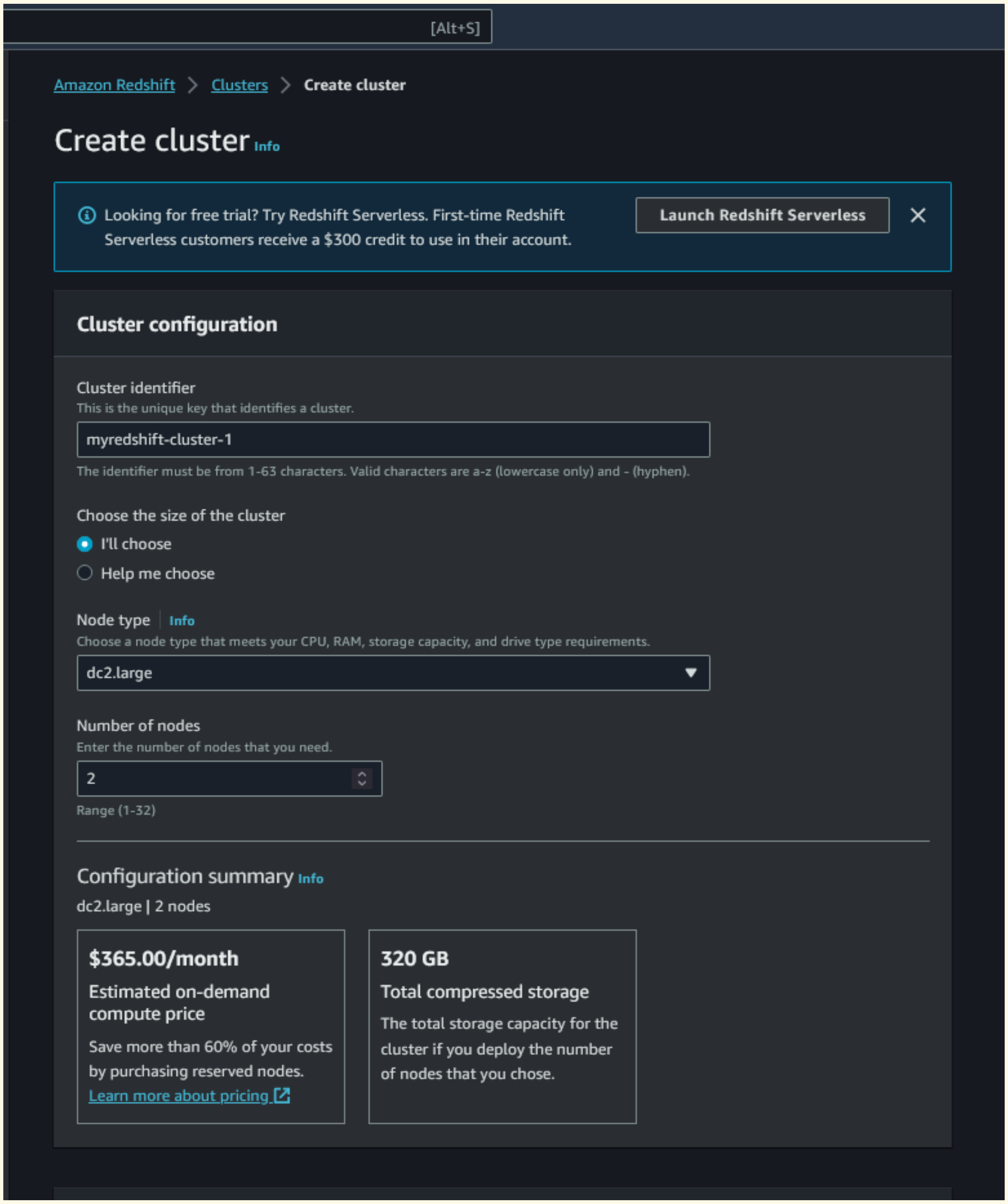


Cluster subnet groups (1)

Name	Status	VPC ID	Description
redshift-subnet-group 2 Subnets	Complete	vpc-0a97b0373b6cb4331	Private subnet access for Redshift.

Figure 17. Successful creation of the cluster subnet group.

Now at this stage we begin creation of the Amazon Redshift cluster.



[Alt+S]
Amazon Redshift > Clusters > Create cluster

Create cluster Info

Looking for free trial? Try Redshift Serverless. First-time Redshift Serverless customers receive a $300 credit to use in their account.

Launch Redshift Serverless

Cluster configuration

Cluster identifier
This is the unique key that identifies a cluster.
myredshift-cluster-1
The identifier must be from 1-63 characters. Valid characters are a-z (lowercase only) and - (hyphen).

Choose the size of the cluster
● I'll choose
○ Help me choose

Node type | Info
Choose a node type that meets your CPU, RAM, storage capacity, and drive type requirements.
dc2.large

Number of nodes
Enter the number of nodes that you need.
2
Range (1-32)

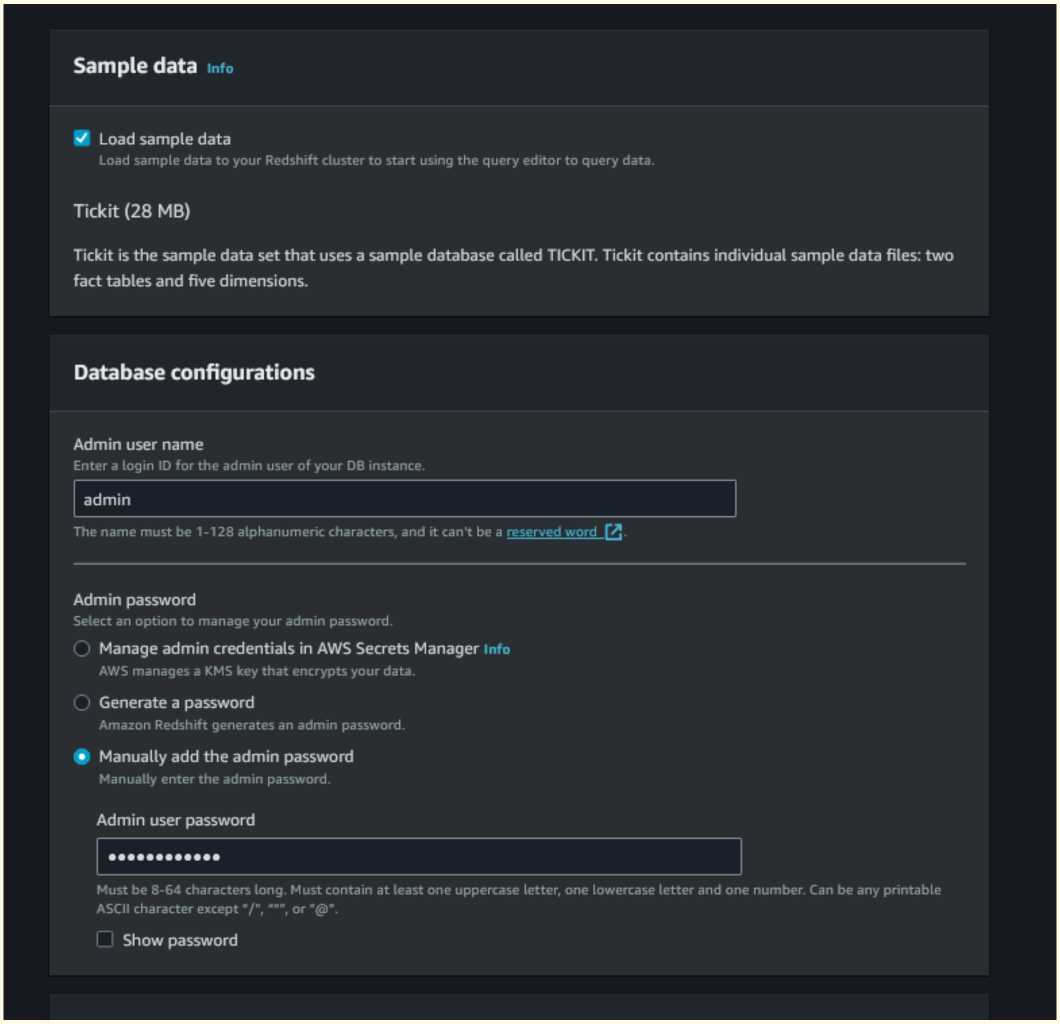
Configuration summary Info
dc2.large | 2 nodes

$365.00/month
Estimated on-demand compute price
Save more than 60% of your costs by purchasing reserved nodes.
Learn more about pricing

320 GB
Total compressed storage
The total storage capacity for the cluster if you deploy the number of nodes that you chose.

Figure 18.

Choose the dc2.large node type and the number of nodes two.



Sample data Info

☑ Load sample data
Load sample data to your Redshift cluster to start using the query editor to query data.

Tickit (28 MB)

Tickit is the sample data set that uses a sample database called TICKIT. Tickit contains individual sample data files: two fact tables and five dimensions.

Database configurations

Admin user name
Enter a login ID for the admin user of your DB instance.
admin
The name must be 1-128 alphanumeric characters, and it can't be a reserved word.

Admin password
Select an option to manage your admin password.

○ Manage admin credentials in AWS Secrets Manager Info
AWS manages a KMS key that encrypts your data.

○ Generate a password
Amazon Redshift generates an admin password.

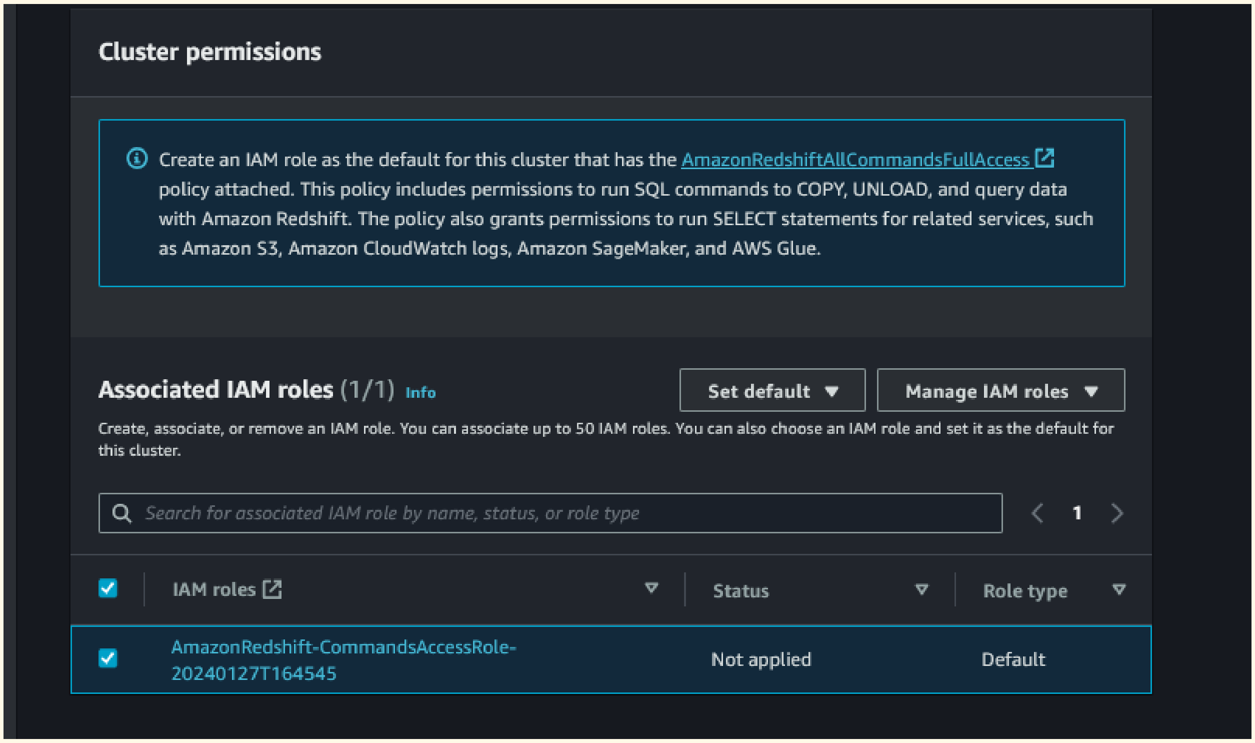
● Manually add the admin password
Manually enter the admin password.

Admin user password
••••••••••••
Must be 8-64 characters long. Must contain at least one uppercase letter, one lowercase letter and one number. Can be any printable ASCII character except "/", """, or "@".

□ Show password

Figure 19.

In the above diagram (Figure 19), you could opt to enable sample data generation. Configure the master's name and master password.



Cluster permissions

ⓘ Create an IAM role as the default for this cluster that has the AmazonRedshiftAllCommandsFullAccess policy attached. This policy includes permissions to run SQL commands to COPY, UNLOAD, and query data with Amazon Redshift. The policy also grants permissions to run SELECT statements for related services, such as Amazon S3, Amazon CloudWatch logs, Amazon SageMaker, and AWS Glue.

Associated IAM roles (1/1) Info

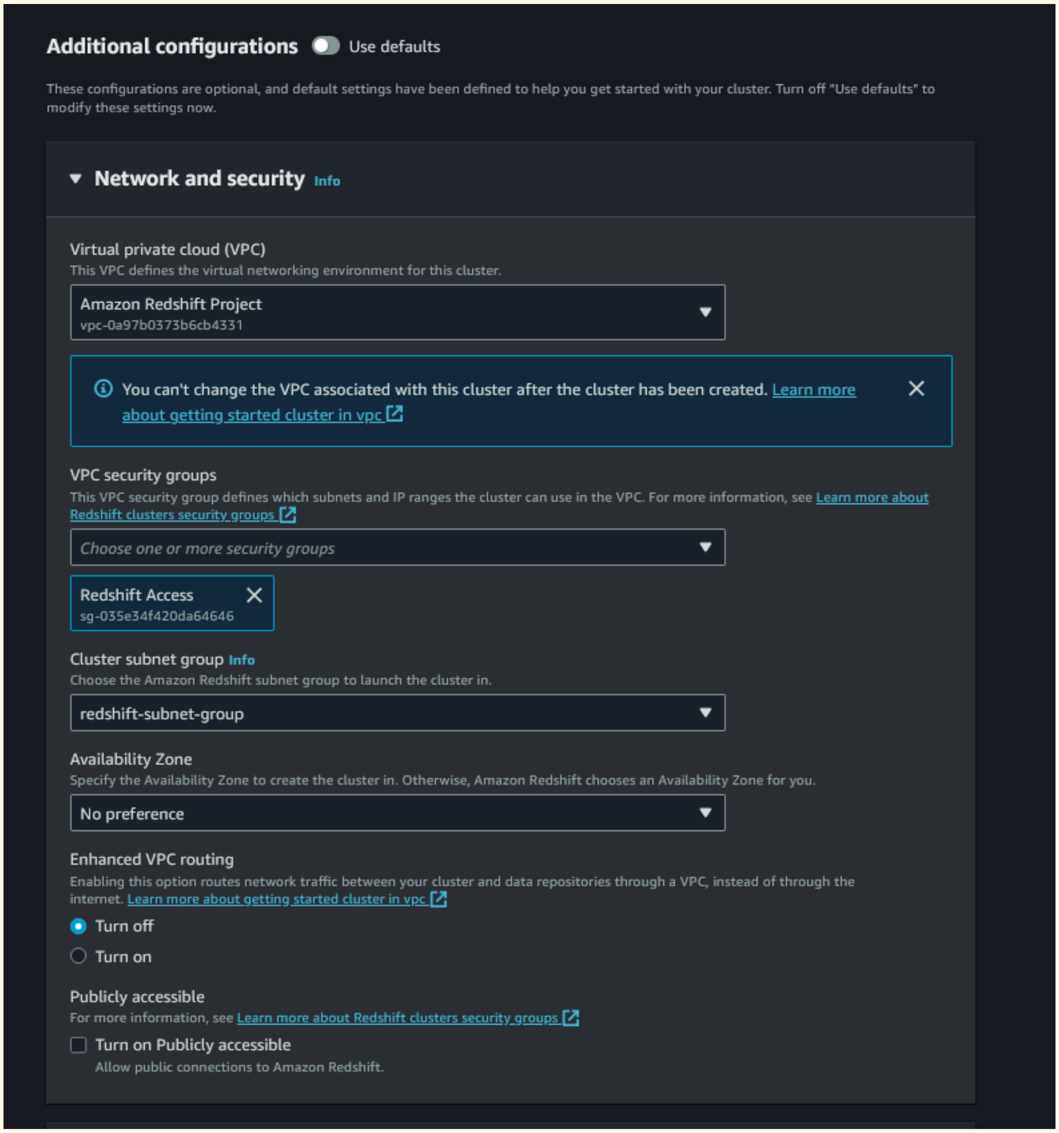
Create, associate, or remove an IAM role. You can associate up to 50 IAM roles. You can also choose an IAM role and set it as the default for this cluster.

Search for associated IAM role by name, status, or role type
1

☑	IAM roles	Status	Role type
☑	AmazonRedshift-CommandsAccessRole-20240127T164545	Not applied	Default

Figure 20.

Allocate the appropriate IAM role that was created earlier for Redshift’s access to the S3 bucket.



Additional configurations

Use defaults

These configurations are optional, and default settings have been defined to help you get started with your cluster. Turn off "Use defaults" to modify these settings now.

Network and security

Virtual private cloud (VPC)
This VPC defines the virtual networking environment for this cluster.
Amazon Redshift Project
vpc-0a97b0373b6cb4331

You can't change the VPC associated with this cluster after the cluster has been created. Learn more about getting started cluster in vpc

VPC security groups
This VPC security group defines which subnets and IP ranges the cluster can use in the VPC. For more information, see Learn more about Redshift clusters security groups
Choose one or more security groups
Redshift Access
sg-035e34f420da64646

Cluster subnet group
Choose the Amazon Redshift subnet group to launch the cluster in.
redshift-subnet-group

Availability Zone
Specify the Availability Zone to create the cluster in. Otherwise, Amazon Redshift chooses an Availability Zone for you.
No preference

Enhanced VPC routing
Enabling this option routes network traffic between your cluster and data repositories through a VPC, instead of through the internet. Learn more about getting started cluster in vpc
● Turn off
○ Turn on

Publicly accessible
For more information, see Learn more about Redshift clusters security groups
□ Turn on Publicly accessible
Allow public connections to Amazon Redshift.

Figure 21.

Choose the VPC we created earlier (Amazon Redshift Project) and select Redshift Access as the VPC security group, deselect the default security group. Choose the redshift subnet group for the cluster and leave the rest of the option as is.

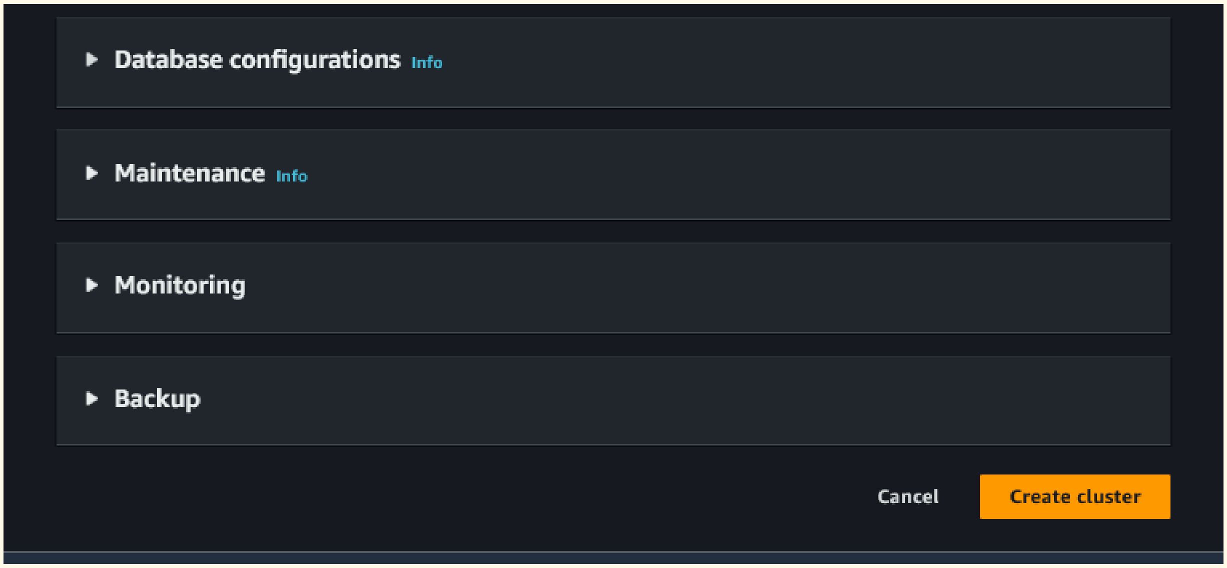
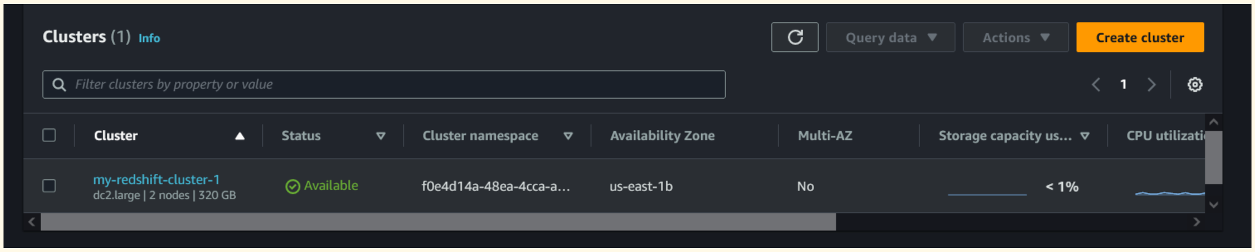


Figure 22. Leave this option in their default settings and click on the create cluster button.



□	Cluster	Status	Cluster namespace	Availability Zone	Multi-AZ	Storage capacity us...	CPU utilizati
□	my-redshift-cluster-1 dc2.large | 2 nodes | 320 GB	Available	f0e4d14a-48ea-4cca-a...	us-east-1b	No	< 1%	

Figure 23. Wait until the cluster creation is complete and available.

INTERACTING WITH OUR NEWLY CREATED AMAZON REDSHIFT CLUSTER

On the left side of the console on the navigation panel, get to the query console by clicking on the query editor. (You can choose between the Query editor or Query editor v2).

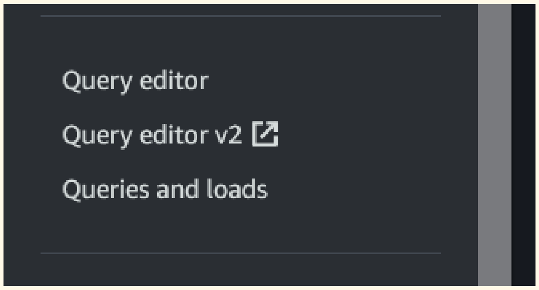


Figure 24.

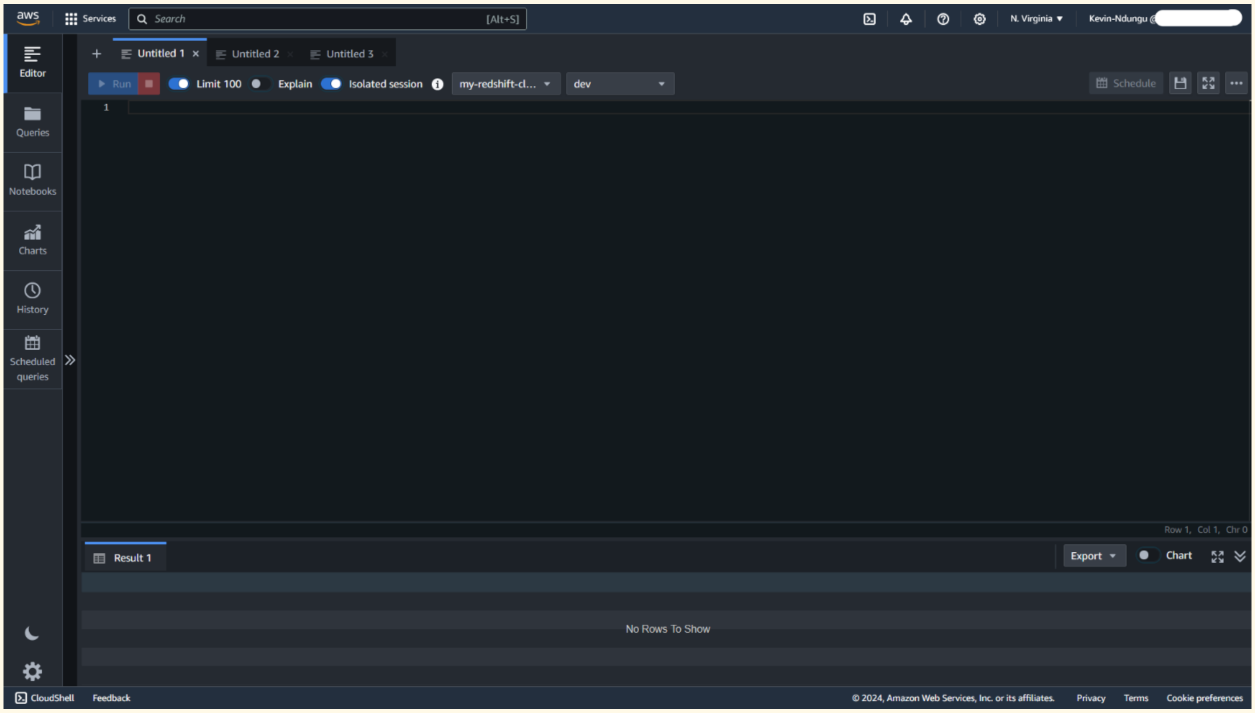
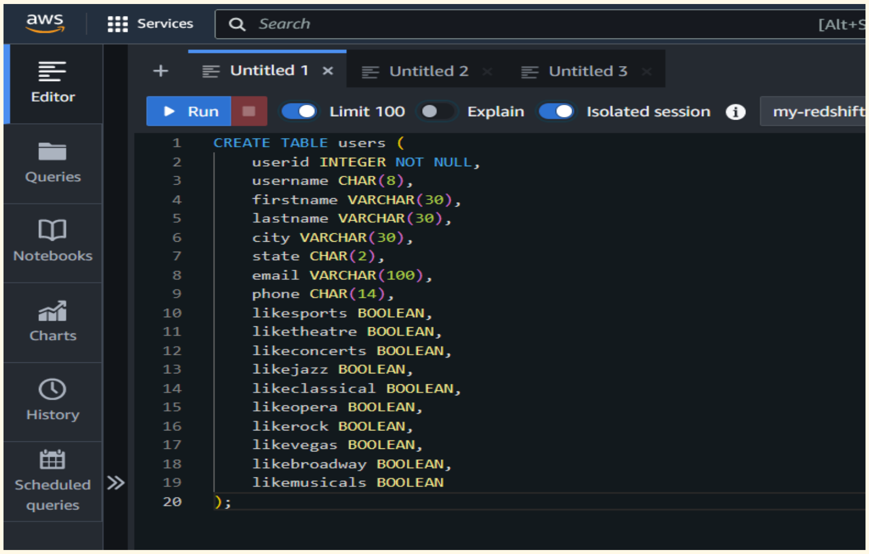


Figure 25.

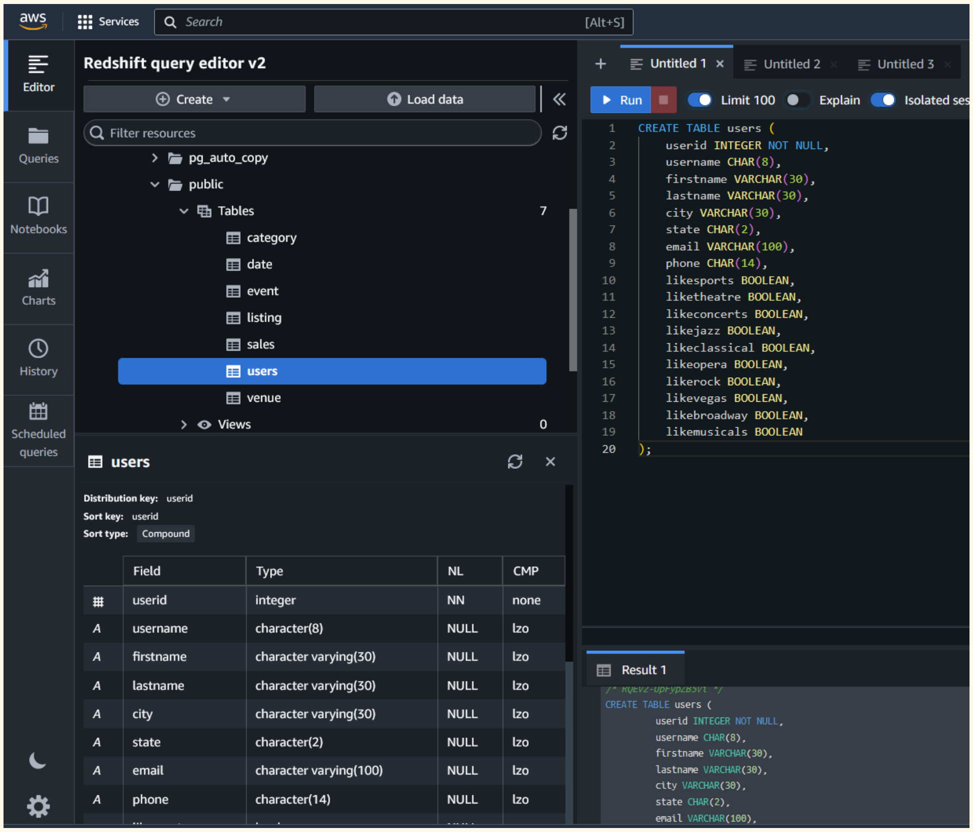
Once the editor opens, the next step is to create a table (*named users*), in this case the below create table query was used:



```sql
CREATE TABLE users (
    userid INTEGER NOT NULL,
    username CHAR(8),
    firstname VARCHAR(30),
    lastname VARCHAR(30),
    city VARCHAR(30),
    state CHAR(2),
    email VARCHAR(100),
    phone CHAR(14),
    likesports BOOLEAN,
    liketheatre BOOLEAN,
    likeconcerts BOOLEAN,
    likejazz BOOLEAN,
    likeclassical BOOLEAN,
    likeopera BOOLEAN,
    likerock BOOLEAN,
    likevegas BOOLEAN,
    likebroadway BOOLEAN,
    likemusicals BOOLEAN
);
```

Figure 26. Click on the Run button at the top left to create table.

Once the table (users) is created it will be listed among other tables in the query's navigation panel as shown below:



users

Distribution key: userid
Sort key: userid
Sort type: Compound

	Field	Type	NL	CMP
#	userid	integer	NN	none
A	username	character(8)	NULL	lzo
A	firstname	character varying(30)	NULL	lzo
A	lastname	character varying(30)	NULL	lzo
A	city	character varying(30)	NULL	lzo
A	state	character(2)	NULL	lzo
A	email	character varying(100)	NULL	lzo
A	phone	character(14)	NULL	lzo

```sql
CREATE TABLE users (
    userid INTEGER NOT NULL,
    username CHAR(8),
    firstname VARCHAR(30),
    lastname VARCHAR(30),
    city VARCHAR(30),
    state CHAR(2),
    email VARCHAR(100),
    phone CHAR(14),
    likesports BOOLEAN,
    liketheatre BOOLEAN,
    likeconcerts BOOLEAN,
    likejazz BOOLEAN,
    likeclassical BOOLEAN,
    likeopera BOOLEAN,
    likerock BOOLEAN,
    likevegas BOOLEAN,
    likebroadway BOOLEAN,
    likemusicals BOOLEAN
);
```

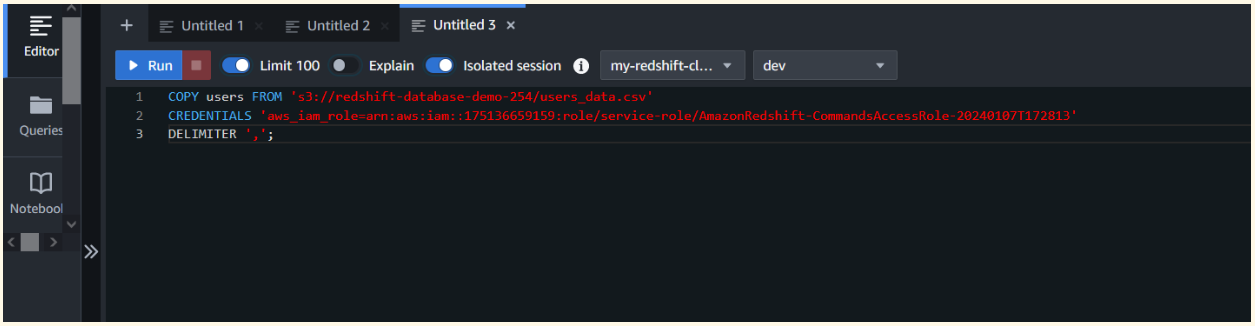
```sql
/* RQEV2-UpFypZB5vt */
CREATE TABLE users (
        userid INTEGER NOT NULL,
        username CHAR(8),
        firstname VARCHAR(30),
        lastname VARCHAR(30),
        city VARCHAR(30),
        state CHAR(2),
        email VARCHAR(100),
```

Figure 27.

Since the table is up, run the query in figure 28 was used to copy the data generated and stored in an S3 bucket to our newly created 'users' table.

So, I'm going to copy users' data from my S3 bucket. The credential that I passed is a role and it gives the ARN the Amazon resource name of the role. And in the file, itself is delimited by a comma and I ran the query.

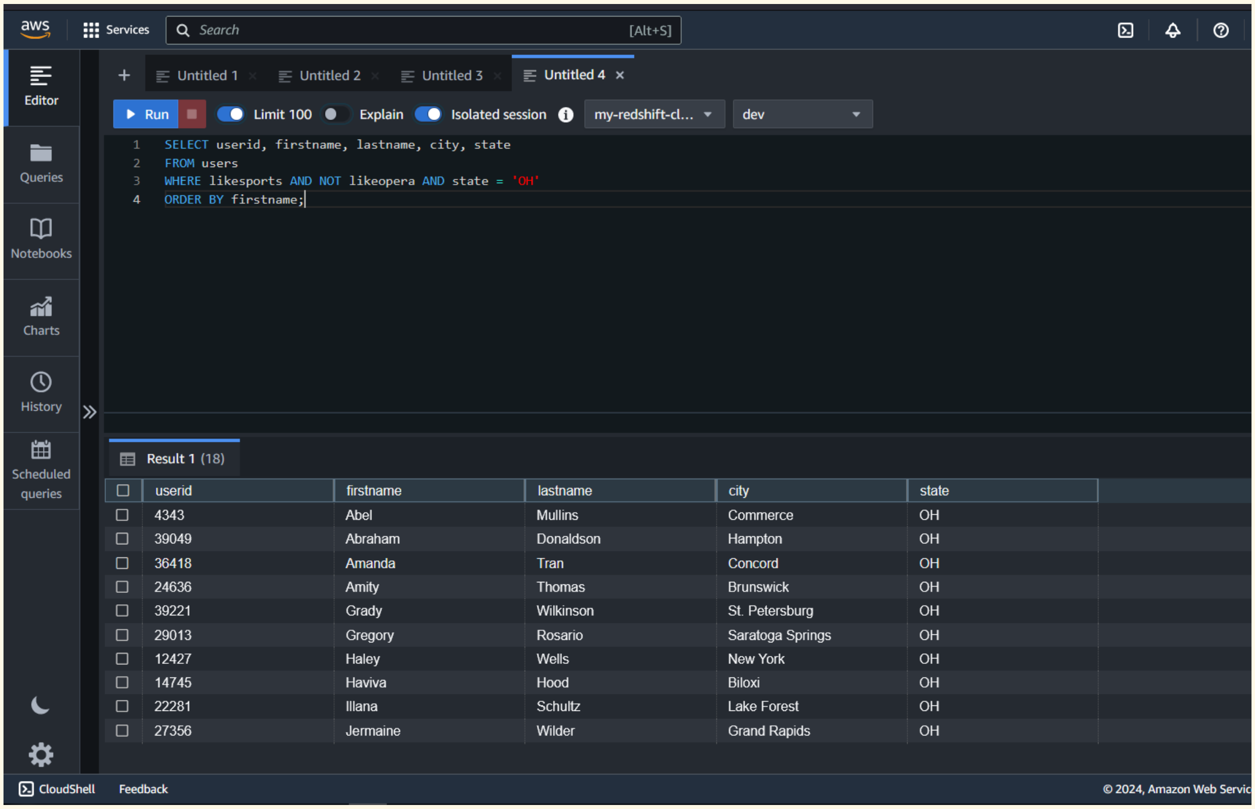
(Note: *Generate the data separately based on the information in the create table query and save it in a pipe delimited text or comma delimited format, store it in the S3 bucket created earlier*).



```
COPY users FROM 's3://redshift-database-demo-254/users_data.csv'
CREDENTIALS 'aws_iam_role=arn:aws:iam::175136659159:role/service-role/AmazonRedshift-CommandsAccessRole-20240107T172813'
DELIMITER ',';
```

Figure 28.

Once the users table is populated, we can begin running some queries and get to view the results in the bottom display of the query editor.

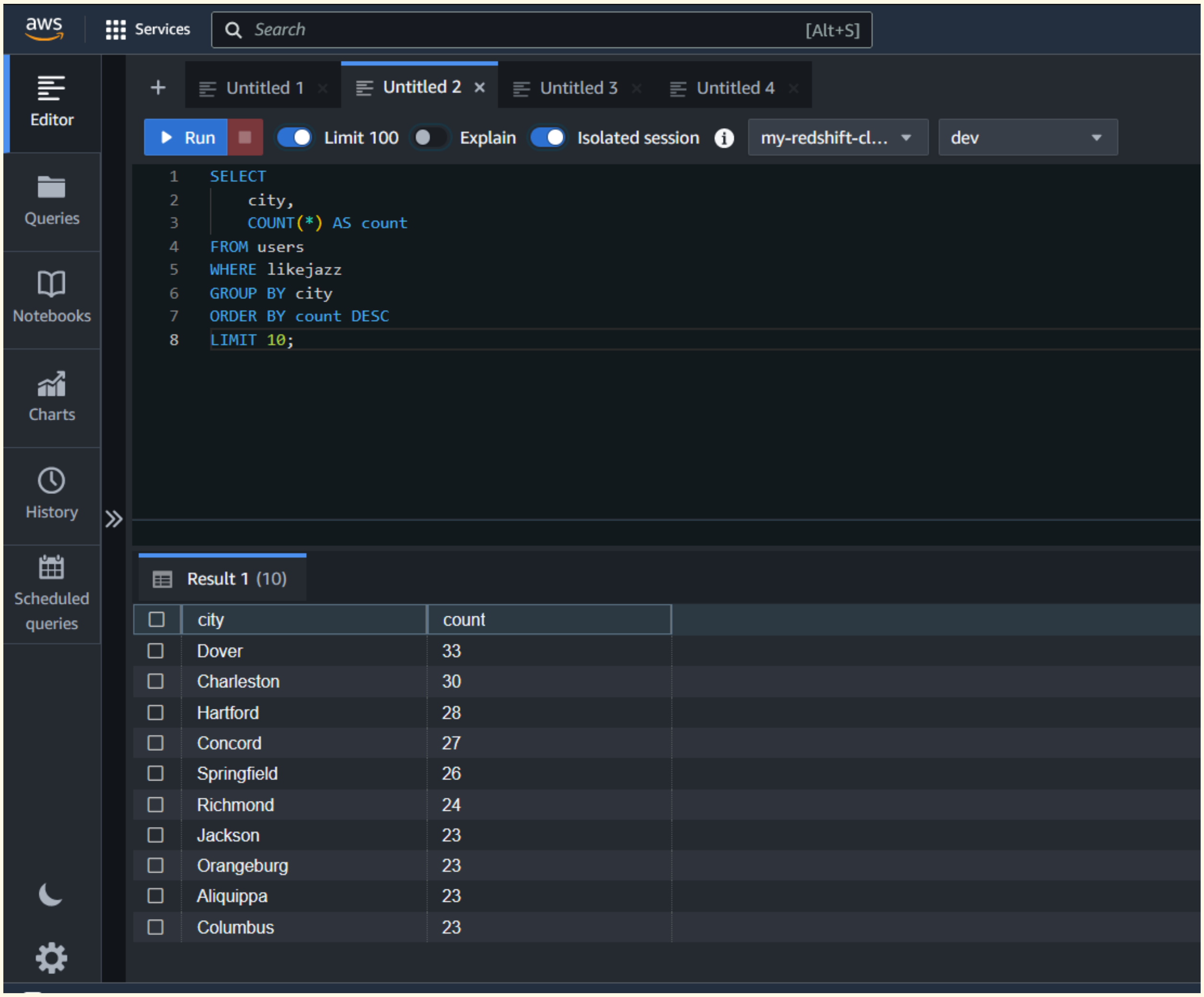


```
SELECT userid, firstname, lastname, city, state
FROM users
WHERE likesports AND NOT likeopera AND state = 'OH'
ORDER BY firstname;
```

userid	firstname	lastname	city	state
4343	Abel	Mullins	Commerce	OH
39049	Abraham	Donaldson	Hampton	OH
36418	Amanda	Tran	Concord	OH
24636	Amity	Thomas	Brunswick	OH
39221	Grady	Wilkinson	St. Petersburg	OH
29013	Gregory	Rosario	Saratoga Springs	OH
12427	Haley	Wells	New York	OH
14745	Haviva	Hood	Biloxi	OH
22281	Illana	Schultz	Lake Forest	OH
27356	Jermaine	Wilder	Grand Rapids	OH

Figure 29.

In running the query code in the above diagram, we want to get information user ID, first name, last name, city state from users where people like sports and do not like opera and the state is Ohio and then giving an order by first name. The output is displayed in the Result tab below the editor area.



```
SELECT
    city,
    COUNT(*) AS count
FROM users
WHERE likejazz
GROUP BY city
ORDER BY count DESC
LIMIT 10;
```

□	city	count
□	Dover	33
□	Charleston	30
□	Hartford	28
□	Concord	27
□	Springfield	26
□	Richmond	24
□	Jackson	23
□	Orangeburg	23
□	Aliquippa	23
□	Columbus	23

Figure 30.

In the above diagram I wanted to see the top 10, I ran the query and I as shown the top 10 are 33 in Dover down to 23 in Columbus.

CONCLUSION

In conclusion, the Redshift Data Pipeline project has reached a significant milestone with the successful configuration and creation of the Amazon Redshift cluster. Throughout this documentation, we've meticulously detailed the setup and configuration of our cloud environment, laying the foundation for a robust data infrastructure. From the inception of our Virtual Private Cloud (VPC) to the establishment of connectivity between our Redshift cluster and Amazon S3, each step has been carefully orchestrated to ensure seamless data management and analysis.

We're now poised to leverage the power of Amazon Redshift for advanced analytics and insights generation. As we continue our journey, this project serves as a testament to the efficacy of cloud-based data pipelines in driving business intelligence and decision-making processes.

ADDITIONAL RESOURCES

This project was inspired by the AWS Skill Builder Data warehouse materials and trainings. For more details in regards to AWS Big Data services, below are essentials links provided:

- Amazon Redshift FAQs
- Amazon Redshift Serverless
- AWS CloudFormation
- Amazon Simple Storage Service