

Cycle 08 AWS Homework

Topic: Amazon Redshift Cluster Deployment and Secure Connectivity

1. Deploy and Configure a Redshift Cluster

Objective: Provision and set up a Redshift data warehouse from scratch.

Steps Performed:

1. Logged into AWS Console → Amazon Redshift.
2. Created a **single-node RA3 cluster** in the assigned region.
3. Cluster named: `studentname-hw-cluster`.
4. Set master username and secure password (different from demo credentials).
5. Initially kept **Public Access = No**.
6. Waited for the cluster status to show **Available**.

Cluster Details Captured:

- Cluster Identifier: `studentname-hw-cluster`
- Node Type: `ra3.xlplus` (single node)
- Endpoint: `<copy from AWS Console>`
- Port: `5439`

(Insert Screenshot: Redshift Cluster available with endpoint highlighted)

2. Establish a Secure Connection using an EC2 Client

Objective: Access Redshift securely via EC2.

Steps Performed:

1. Launched an **Ubuntu EC2 instance** in the same VPC and region as Redshift.

2. Connected via **SSH**:

```
ssh -i "key.pem" ubuntu@<EC2-Public-IP>
```

3. Updated and installed PostgreSQL client:

```
sudo apt update  
sudo apt install postgresql-client -y
```

4. Verified installation:

```
psql --version
```

(Insert Screenshot: psql version confirmation on EC2)

3. Configure Network Security for Access

Objective: Enable secure database connectivity.

Steps Performed:

1. Modified Redshift → **Publicly Accessible = Yes**.

2. Located **EC2 security group** → Added inbound rule:

- Type: PostgreSQL
- Port: 5439
- Source: Redshift SG (best practice) / My IP (for exercise).

3. Located **Redshift security group** → Added inbound rule:

- Type: PostgreSQL
- Port: 5439
- Source: EC2 security group.

(Insert Screenshot: Security group inbound rules for both EC2 and Redshift)

4. Connect and Perform Basic Database Operations

Objective: Practice SQL commands in Redshift via `psql`.

Steps Performed:

1. Connected to Redshift:

```
psql -h <redshift-endpoint> -U <master-username> -d dev -p 5439
```

2. Listed databases:

```
\l
```

Step 3: Create a new database

```
CREATE DATABASE training_db;
```

Step 4: To check databases

```
SELECT datname FROM pg_database;
```

Step 5: Connect to new database

```
\c training_db;
```

Step 6: Create a table

```
CREATE TABLE courses (
    course_id INT,
```

```
course_name VARCHAR(100),  
duration_months INT  
);
```

Step 7: List tables

```
\d  
-- or  
SELECT tablename FROM pg_table_def WHERE schemaname = 'public';
```

Step 8: View schema of the table (fixing earlier mistake)

```
SELECT col, type, encoding, distkey, sortkey  
FROM pg_table_def  
WHERE tablename = 'courses';
```

Step 9: Insert sample rows

```
INSERT INTO courses VALUES  
(101, 'AWS Fundamentals', 2),  
(102, 'Data Engineering with Redshift', 3),  
(103, 'Machine Learning Basics', 4);
```

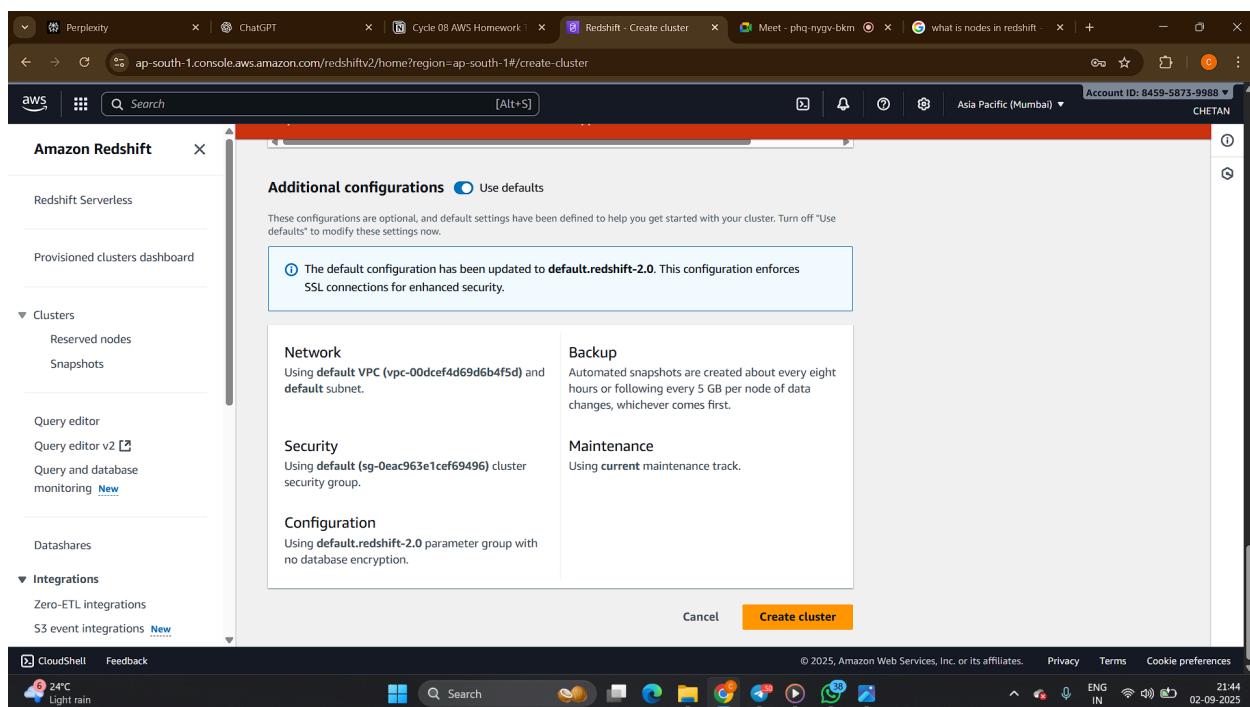
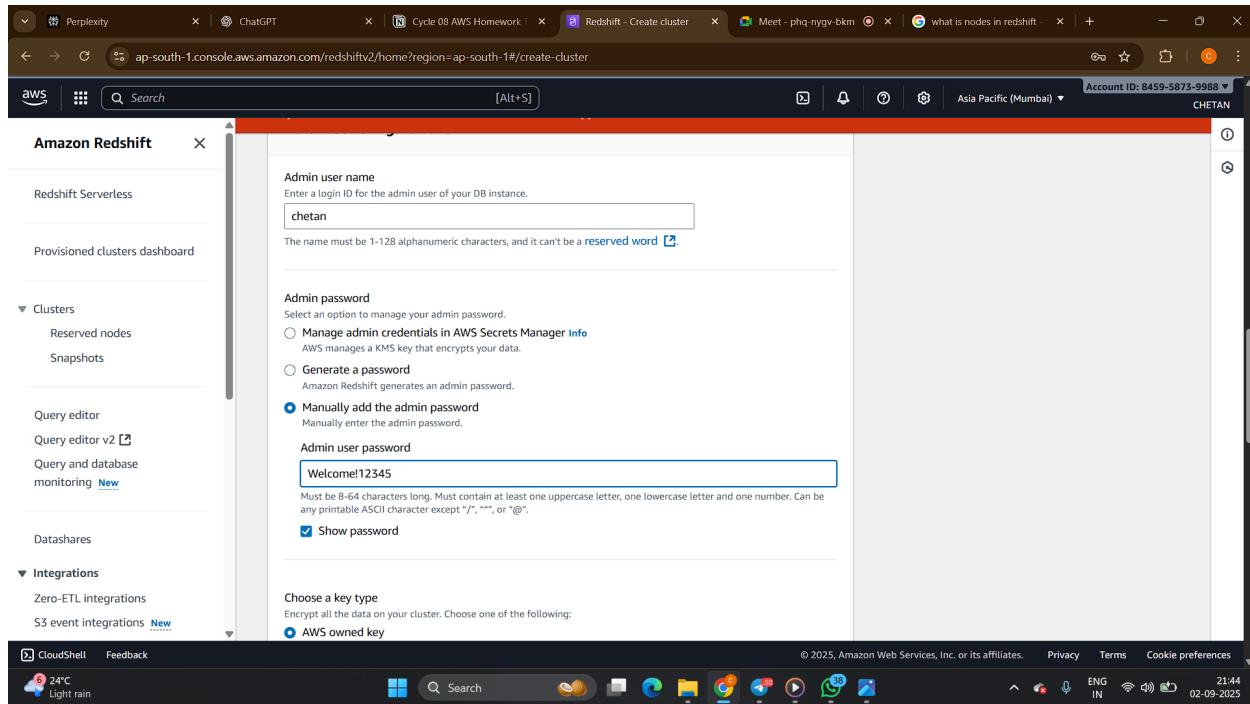
Step 10: Query the table

```
SELECT * FROM courses;
```

(Insert Screenshot: psql session showing database, table, and SELECT results)

The screenshot shows the AWS Redshift Clusters page. On the left, a sidebar menu includes 'Redshift Serverless', 'Provisioned clusters dashboard', 'Clusters' (with 'Reserved nodes' and 'Schemas' sub-options), 'Query editor', 'Data monitoring', 'Dataload', 'Integrations', 'Zero-ETL integrations', and 'S3 event integrations'. The main content area has a heading 'then select "Register with AWS Glue Data Catalog" from the Actions menu.' Below this, there's a section for connecting via the query editor v2, another for JDBC or ODBC drivers, and a 'Clusters (0)' table with a 'Create cluster' button. The bottom status bar shows 'CloudShell' and 'Feedback'.

The screenshot shows the 'Create cluster' wizard. The 'Cluster configuration' step is displayed. It includes fields for 'Cluster identifier' (set to 'redshift-cluster-1'), 'Choose the size of the cluster' (radio button selected for 'I'll choose' and 'ra3.large' node type), and 'AZ configuration' (radio button selected for 'Help me choose'). A note about Redshift Serverless is visible at the top. The bottom status bar shows 'CloudShell' and 'Feedback'.



The screenshot shows the AWS Redshift Clusters console. On the left sidebar, under 'Clusters', there is a section for 'Reserved nodes' and 'Schemas'. Below that, under 'Integrations', there are 'Zero-ETL integrations' and 'S3 event integrations'. The main area displays a 'Clusters (1) Info' table with one row for 'redshift-cluster-1', which is currently 'Creating'. To the right of the table, there are sections for 'Query data using Redshift query editor', 'Work with your client tools', and 'Choose your JDBC or ODBC driver'. The 'Driver' dropdown is set to 'JDBC 4.2 without AWS SDK (.jar)'. Buttons for 'Copy JDBC URL' and 'Copy ODBC URL' are available, along with a 'Download driver' link.

The screenshot shows the 'Create the default IAM role' dialog box. The 'Associated IAM roles (0)' section includes a 'Set default' dropdown, an 'Actions' button, and an 'Associate' button. Below this, there is a note about associating a serverless endpoint with an IAM role. The 'Select Access Type' section has a 'Full Access' radio button selected. The 'Specify an S3 bucket for the IAM role to access' section includes a note about creating a new bucket or using an existing one, and two options: 'No additional S3 bucket' (selected) and 'Specific S3 buckets'. At the bottom of the dialog are 'Cancel' and 'Create IAM role as default' buttons.

The screenshot shows the AWS Redshift Serverless setup page. A message box at the top right says: "create an IAM role as the default for this configuration that has the [AmazonRedshiftAllCommandsFullAccess](#) policy attached. This policy includes permissions to run SQL commands to COPY, UNLOAD, and query data with Amazon Redshift Serverless. This policy also grants permissions to run SELECT statements for related services, such as Amazon S3, Amazon CloudWatch logs, Amazon SageMaker, and AWS Glue. You won't be able to run these SQL commands without an IAM role attached to your namespace." Below this, a green success message box says: "The IAM role [AmazonRedshift-CommandsAccessRole-20250902T215338](#) was successfully created and set as the default." A table titled "Associated IAM roles (1)" shows one item: "AmazonRedshift-CommandsAccessRole-20250902T215338" with status "Not applied". The status bar at the bottom indicates it's 21:53 on 02-09-2025.

The screenshot shows the AWS EC2 Launch an instance page. It starts with a "Name and tags" section where "RedshiftClient" is entered. Below it is a "Application and OS Images (Amazon Machine Image)" section with a search bar and a list of AMI icons: Amazon Linux, macOS, Ubuntu, Windows, Red Hat, SUSE Linux, and Debian. To the right, there's a "Summary" section with "Number of instances": 1, "Software Image (AMI)": Canonical, Ubuntu, 24.04, amd64... (with a "read more" link), "Virtual server type (instance type)": t2.micro, "Firewall (security group)": New security group, "Storage (volumes)": 1 volume(s) - 8 GiB, and a note about "Free tier" usage. At the bottom are "Cancel", "Launch instance", and "Preview code" buttons. The status bar at the bottom indicates it's 21:55 on 02-09-2025.

The screenshot shows the AWS EC2 Instances page. On the left, there's a navigation sidebar with sections like EC2, Dashboard, EC2 Global View, Events, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images (AMIs, AMI Catalog), and Elastic Block Store (Volumes, Snapshots). The main content area is titled 'Instances (1) Info' and shows a table with one row. The table columns are Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, and Public IPv4. The single instance listed is 'RedshiftClient' with ID 'i-0c05702a523181e1b', status 'Pending', type 't2.micro', and other details matching the table headers.

```

root@ip-172-31-35-220:~ 2 16:28:41 UTC 2025

System information as of Tue Sep 2 16:28:41 UTC 2025
System load: 0.12      Processes:          109
Usage of /: 25.6% of 6.71GB  Users logged in: 0
Memory usage: 22%        IPv4 address for enX0: 172.31.35.220
Swap usage: 0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-35-220:~$ sudo -i
root@ip-172-31-35-220:~# apt get update -i
E: Command line option 'i' [from -i] is not understood in combination with the other options.
root@ip-172-31-35-220:~# apt get update -y
E: Invalid operation get
root@ip-172-31-35-220:~# apt-get update -y
Hit:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:3 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Get:4 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Get:5 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Packages [15.0 MB]

```

```
| root@ip-172-31-35-220: ~ + - x | 22:04 02-09-2025
Get: 31 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-backports/main amd64 Components [7056 B]
Get: 32 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-backports/main amd64 c-n-f Metadata [272 B]
Get: 33 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-backports/universe amd64 Packages [30.2 kB]
Get: 34 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-backports/universe Translation-en [17.4 kB]
Get: 35 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-backports/universe amd64 Components [19.2 kB]
Get: 36 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-backports/universe amd64 c-n-f Metadata [1394 B]
Get: 37 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-backports/restricted amd64 Components [212 B]
Get: 38 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-backports/restricted amd64 c-n-f Metadata [216 B]
Get: 39 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-backports/multiverse amd64 Components [212 B]
Get: 40 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-backports/multiverse amd64 c-n-f Metadata [116 B]
Get: 41 http://security.ubuntu.com/ubuntu noble-security/main Translation-en [191 kB]
Get: 42 http://security.ubuntu.com/ubuntu noble-security/main amd64 Components [21.6 kB]
Get: 43 http://security.ubuntu.com/ubuntu noble-security/main amd64 c-n-f Metadata [8708 B]
Get: 44 http://security.ubuntu.com/ubuntu noble-security/universe amd64 Packages [879 kB]
Get: 45 http://security.ubuntu.com/ubuntu noble-security/universe Translation-en [194 kB]
Get: 46 http://security.ubuntu.com/ubuntu noble-security/universe amd64 Components [52.3 kB]
Get: 47 http://security.ubuntu.com/ubuntu noble-security/universe amd64 c-n-f Metadata [18.0 kB]
Get: 48 http://security.ubuntu.com/ubuntu noble-security/restricted amd64 Packages [1697 kB]
Get: 49 http://security.ubuntu.com/ubuntu noble-security/restricted Translation-en [381 kB]
Get: 50 http://security.ubuntu.com/ubuntu noble-security/restricted amd64 Components [208 B]
Get: 51 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 Packages [18.5 kB]
Get: 52 http://security.ubuntu.com/ubuntu noble-security/multiverse Translation-en [4288 B]
Get: 53 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 Components [208 B]
Get: 54 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 c-n-f Metadata [380 B]
Fetched 37.0 MB in 7s (5182 kB/s)
Reading package lists... Done
root@ip-172-31-35-220:~# sudo apt-get install -y postgresql-client
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libpq5 postgresql-client-16 postgresql-client-common
Suggested packages:
  postgresql-16 postgresql-doc-16
The following NEW packages will be installed:
  libpq5 postgresql-client postgresql-client-16 postgresql-client-common
0 upgraded, 4 newly installed, 0 to remove and 13 not upgraded.
Need to get 1485 kB of archives.
After this operation, 4948 kB of additional disk space will be used.
Get:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 libpq5 amd64 16.9-0ubuntu0.24.04.1 [143 kB]
Get:2 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 postgresql-client-common all 257build1.1 [36.4 kB]

24°C Light rain 22:04 02-09-2025
```

```
| root@ip-172-31-35-220: ~ + - x | 22:05 02-09-2025
root@ip-172-31-35-220:~# psql --version
psql (PostgreSQL) 16.9 (Ubuntu 16.9-0ubuntu0.24.04.1)
root@ip-172-31-35-220:~#
```

Screenshot of the AWS Redshift Cluster Details page for 'redshift-cluster-1'.

General information

Cluster identifier redshift-cluster-1	Status Available	Node type ra3.large	Endpoint redshift-cluster-1.c4zejdlbqhc.ap-south-1.redshift.amazonaws.com:5439/dev
Custom domain name -	Date created September 02, 2025, 21:46 IST	Number of nodes 1	JDBC URL jdbc:redshift://redshift-cluster-1.c4zejdlbqhc.ap-south-1.redshift.amazonaws.com:5439/dev
Cluster namespace ARN arn:aws:redshift:ap-south-1:845958739988:namespace:df5f9da3-221c-4668-9972-bccbfef57ff9	Multi-AZ No	Patch version Patch 192	ODBC URL Driver={Amazon Redshift (x64)};Server=redshift-cluster-1.c4zejdlbqhc.ap-south-1.redshift.amazonaws.com;Database=dev
Namespace register status Deregistered	Cluster configuration Production	Storage used 0.04 of 976.6 GB used (0.00%)	

Actions | Edit | Add partner integration | **Query data**

Cluster details | Query monitoring | **Databases** | Datashares | Integrations | Resource Policy | Schedules | Maintenance | Properties

CloudShell | Feedback | © 2025, Amazon Web Services, Inc. or its affiliates. | Privacy | Terms | Cookie preferences | ENG IN | 22:09 | 02-09-2025

Screenshot of the AWS Redshift Cluster Details page for 'redshift-cluster-1'.

Properties

Cluster namespace ARN arn:aws:redshift:ap-south-1:845958739988:namespace:df5f9da3-221c-4668-9972-bccbfef57ff9	Multi-AZ No	Patch version Patch 192	ODBC URL Driver={Amazon Redshift (x64)};Server=redshift-cluster-1.c4zejdlbqhc.ap-south-1.redshift.amazonaws.com;Database=dev
Namespace register status Deregistered	Cluster configuration Production	Storage used 0.04 of 976.6 GB used (0.00%)	

Cluster details | Query monitoring | Databases | Datashares | Integrations | Resource Policy | Schedules | Maintenance | **Properties**

Database configurations

Database configurations		Edit admin credentials	Rotate encryption keys	Edit
Database name dev	Parameter group Defines database parameter and query queues for all the databases. default.redshift-2.0	Encryption Enabled	Audit logging Disabled	
Port 5439		AWS KMS key ID AWS_OWNED_KMS_KEY		
Admin user name chetan	SSH ingestion setting (cluster public key) ssh-rsa AAAAAB3NzaC1y2EAAAQABAAQDa95qxlo9KmiyFn/AVBOV2gFSUUhWzK5BDw1Lch1hn/Ogs10TAL/5eEWwy60EmiisBT0mjqiXzbx6NbX3+EsTQ9kyU9GJwy0UuaWD7WTg2+OfJXz/3gFO4OEotAyE1k7B818BTdFkFbVp0Q			

CloudShell | Feedback | © 2025, Amazon Web Services, Inc. or its affiliates. | Privacy | Terms | Cookie preferences | ENG IN | 22:10 | 02-09-2025

Screenshot of the AWS Redshift Cluster Properties page.

Network and security settings

Virtual private cloud (VPC) vpc-00dcf4d69d6b4f5d	Availability Zone ap-south-1b	VPC security group sg-0eac963e1cef69496	Publicly accessible Turned off
Subnet group default	Enhanced VPC routing Disabled	IP address type -	
Endpoint URL vpce-01e70f42b9de171a3			

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

Screenshot of the AWS EC2 Security Groups page.

Edit inbound rules

Security Group Rule	Protocol & Port	Source	Action
sgr-0daaaae4f0126c177	PostgreSQL / 5432	Custom / ::/0	Delete
sgr-080050ee61f907a44	PostgreSQL / 5432	Custom / 0.0.0.0/0	Delete
sgr-0d2e05dd47beab888	MySQL/Aurora / 3306	Custom / ::/0	Delete
sgr-007eb44ef28d99381	MySQL/Aurora / 3306	Custom / 0.0.0.0/0	Delete
-	Custom TCP / 5439	Anywhere / 0.0.0.0/0	Delete
-	Custom TCP / 5439	Anywhere / ::/0	Delete

Add rule

⚠ Rules with source of 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.

Buttons: Cancel, Preview changes, Save rules

The screenshot shows the AWS Redshift cluster properties page for a cluster named 'redshift-cluster-1'. The 'Network and security settings' section is displayed, showing the following details:

- Virtual private cloud (VPC)**: `vpc-00dcef4d69d6b4f5d`
- Availability Zone**: `ap-south-1b`
- VPC security group**: `sg-0eac963e1cef69496`
- Publicly accessible**: Turned off
- Subnet group**: `default`
- Enhanced VPC routing**: Disabled
- Endpoint URL**: `vpce-01e70f42b9de171a3`
- IP address type**: -

A note at the bottom of the section states: "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 CloudWatch Metrics, and Amazon Kinesis Data Analytics." Below this note is a link to "Edit".

The screenshot shows the 'Edit' page for the same Redshift cluster. The 'Virtual private cloud (VPC)' section is expanded, showing the following configuration:

- VPC security groups**: A dropdown menu titled 'Choose one or more security groups' contains the entry `default` (`sg-0eac963e1cef69496`).
- Cluster subnet group**: Set to `default`.
- Availability Zone**: Set to `No preference`.
- Enhanced VPC routing**: Set to `Turn off`.
- Publicly accessible**: Set to `Turn on Publicly accessible`.

A warning message in a callout box states: "Turning on the Publicly accessible feature grants outside sources access to your Redshift Provisioned cluster. This instance becomes public and outside sources can connect to it." Below this message is a link to "Learn more".

The screenshot shows the AWS Redshift cluster properties page for a cluster named 'redshift-cluster-1'. The 'Network and security settings' section is displayed, showing the following details:

Virtual private cloud (VPC)	Availability Zone	VPC security group	Publicly accessible
vpc-00dcef4d69d6b4f5d	ap-south-1b	sg-0eac963e1cef69496	Turned on
Subnet group	Enhanced VPC routing		
default	Disabled		
Endpoint URL			
vpce-01e70f42b9de171a3			

A tooltip for the endpoint URL indicates it has been copied.

The screenshot shows the same AWS Redshift cluster properties page as the previous one, but with a tooltip indicating that the endpoint URL 'vpce-01e70f42b9de171a3' has been copied. The rest of the network and security settings remain the same.

```
root@ip-172-31-35-220:~# psql --version
psql (PostgreSQL) 16.9 (Ubuntu 16.9-0ubuntu0.24.04.1)
root@ip-172-31-35-220:~# psql -h redshift-cluster-1.c4zejdlbqhc.ap-south-1.redshift.amazonaws.com:5439/dev -p 5439 -d dev --username chetan -W
psql: error: could not translate host name "redshift-cluster-1.c4zejdlbqhc.ap-south-1.redshift.amazonaws.com:5439/dev" to address: Name or service not known
n
root@ip-172-31-35-220:~# psql -h redshift-cluster-1.c4zejdlbqhc.ap-south-1.redshift.amazonaws.com \
-p 5439 \
-d dev \
-U chetan
Password for user chetan:
psql (16.9 (Ubuntu 16.9-0ubuntu0.24.04.1), server 8.0.2)
WARNING: psql major version 16, server major version 8.0.
        Some psql features might not work.
SSL connection (protocol: TLSv1.3, cipher: TLS_AES_128_GCM_SHA256, compression: off)
Type "help" for help.

dev=#

```

```
hw_db=# \q
root@ip-172-31-35-220:~# psql -h redshift-cluster-1.c4zejdlbqhc.ap-south-1.redshift.amazonaws.com -p 5439 -d dev -U chetan
Password for user chetan:
psql (16.9 (Ubuntu 16.9-0ubuntu0.24.04.1), server 8.0.2)
WARNING: psql major version 16, server major version 8.0.
        Some psql features might not work.
SSL connection (protocol: TLSv1.3, cipher: TLS_AES_128_GCM_SHA256, compression: off)
Type "help" for help.

dev=#
ERROR: column d.datcollate does not exist
dev=# CREATE DATABASE training_db;
CREATE DATABASE
dev=# SELECT datname FROM pg_database;
datname
-----
awsdatacatalog
hw_db
training_db
dev
padb_harvest
sys/internal
template
template0
(8 rows)

dev=# \c training_db;
psql (16.9 (Ubuntu 16.9-0ubuntu0.24.04.1), server 8.0.2)
WARNING: psql major version 16, server major version 8.0.
        Some psql features might not work.
SSL connection (protocol: TLSv1.3, cipher: TLS_AES_128_GCM_SHA256, compression: off)
You are now connected to database "training_db" as user "chetan".
training_db=# CREATE TABLE courses (
    course_id INT,
    course_name VARCHAR(100),
    duration_months INT
);
CREATE TABLE
training_db=# \d
   Name      | Type
-----+-----
SELECT tablename FROM pg_table_def WHERE schemaname = 'public';

23C
Mostly cloudy

```

```

root@ip-172-31-35-220: ~ + ~
(8 rows)

dev=# \c training_db;
psql (16.9 (Ubuntu 16.9-0ubuntu0.24.04.1), server 8.0.2)
WARNING: pgsql major version 16, server major version 8.0.
      Some pgsql features might not work.
SSL connection (protocol: TLSv1.3, cipher: TLS_AES_128_GCM_SHA256, compression: off)
You are now connected to database "training_db" as user "chetan".
training_db=# CREATE TABLE courses (
    course_id INT,
    course_name VARCHAR(100),
    duration_months INT
);
CREATE TABLE
training_db=# \d
-- or
SELECT tablename FROM pg_table_def WHERE schemaname = 'public';
Did not find any relation named "--".
training_db=# SELECT tablename FROM pg_table_def WHERE schemaname = 'public';
tablename
-----
courses
courses
courses
(3 rows)

training_db=# SELECT col, type, encoding, distkey, sortkey
FROM pg_table_def
WHERE tablename = 'courses';
ERROR: column "col" does not exist in pg_table_def
training_db=# SELECT "column", type, encoding, distkey, sortkey
FROM pg_table_def
WHERE tablename = 'courses';
   column   |      type      | encoding | distkey | sortkey
-----+-----+-----+-----+-----+
course_id | integer      | az64     | f       | 0
course_name | character varying(100) | lzo      | f       | 0
duration_months | integer      | az64     | f       | 0
(3 rows)

training_db=#

```

23°C Mostly cloudy Search ENG IN 22:38 02-09-2025

```

root@ip-172-31-35-220: ~ + ~
CREATE TABLE
training_db=# \d
-- or
SELECT tablename FROM pg_table_def WHERE schemaname = 'public';
Did not find any relation named "--".
training_db=# SELECT tablename FROM pg_table_def WHERE schemaname = 'public';
tablename
-----
courses
courses
courses
(3 rows)

training_db=# SELECT col, type, encoding, distkey, sortkey
FROM pg_table_def
WHERE tablename = 'courses';
ERROR: column "col" does not exist in pg_table_def
training_db=# SELECT "column", type, encoding, distkey, sortkey
FROM pg_table_def
WHERE tablename = 'courses';
   column   |      type      | encoding | distkey | sortkey
-----+-----+-----+-----+-----+
course_id | integer      | az64     | f       | 0
course_name | character varying(100) | lzo      | f       | 0
duration_months | integer      | az64     | f       | 0
(3 rows)

training_db=# INSERT INTO courses VALUES
(101, 'AWS Fundamentals', 2),
(102, 'Data Engineering with Redshift', 3),
(103, 'Machine Learning Basics', 4);
INSERT 0 3
training_db=# SELECT * FROM courses;
course_id | course_name | duration_months
-----+-----+-----+
101 | AWS Fundamentals | 2
102 | Data Engineering with Redshift | 3
103 | Machine Learning Basics | 4
(3 rows)

training_db=#

```

23°C Mostly cloudy Search ENG IN 22:39 02-09-2025

```

root@ip-172-31-35-220:~# \d
training_db=# \d
-- or
SELECT tablename FROM pg_table_def WHERE schemaname = 'public';
Did not find any relation named "___".
training_db=# SELECT tablename FROM pg_table_def WHERE schemaname = 'public';
tablename
-----
courses
courses
courses
(3 rows)

training_db=# SELECT col, type, encoding, distkey, sortkey
FROM pg_table_def
WHERE tablename = 'courses';
ERROR: column "col" does not exist in pg_table_def
training_db=# SELECT "column", type, encoding, distkey, sortkey
FROM pg_table_def
WHERE tablename = 'courses';
   column   |      type      | encoding | distkey | sortkey
-----+-----+-----+-----+-----+
course_id | integer      | az64    | f       | 0
course_name | character varying(100) | lzo     | f       | 0
duration_months | integer      | az64    | f       | 0
(3 rows)

training_db=# INSERT INTO courses VALUES
(101, 'AWS Fundamentals', 2),
(102, 'Data Engineering with Redshift', 3),
(103, 'Machine Learning Basics', 4);
INSERT 0 3
training_db=# SELECT * FROM courses;
 course_id | course_name | duration_months
-----+-----+-----+
 101 | AWS Fundamentals | 2
 102 | Data Engineering with Redshift | 3
 103 | Machine Learning Basics | 4
(3 rows)

training_db=# \q
root@ip-172-31-35-220:~#

```

23°C Mostly cloudy

5. Cost Management and Cleanup

Steps Performed:

1. Took screenshots of:
 - Cluster status
 - Security group rules
 - psql successful connection and queries
2. Deleted Redshift cluster (confirmed deletion).
3. Terminated EC2 instance.

(Insert Screenshot: Redshift deletion in progress / EC2 terminated)

6. Learnings and Outcomes

- Understood the **Redshift cluster creation process** and endpoint usage.
- Learned how to **use EC2 as a jump host** for secure access.
- Practiced configuring **security groups for least-privilege access**.

- Gained hands-on experience with **psql client** for Redshift.
- Reinforced habit of **cleaning up AWS resources** to avoid costs.