

# Setting Up a Private Database from a Snapshot and Connecting via EC2 and PGAdmin

---

## Contents

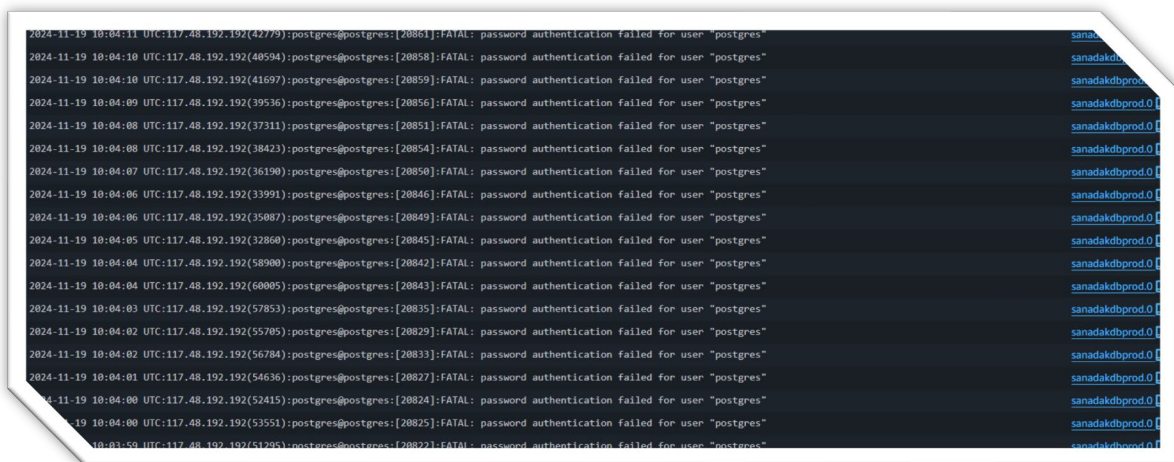
1. Introduction .....	3
2. Creating a Snapshot .....	4
3. Restoring a Private Database from the Snapshot.....	5
4. Connecting to the Database via EC2.....	6
5. Assisting with PGAdmin Connection .....	6

## 1. Introduction

### Why Create a Private Database in RDS Without Public IP Access?

This setup is crucial to enhance database security and mitigate risks associated with public access. Previously, the database was exposed to the public internet, leaving it vulnerable to frequent brute-force attacks, such as repeated password authentication failures (e.g., ``postgres@identity:[9181]:FATAL: password authentication failed for user "postgres"`)`).

These attacks attempt to gain unauthorized access by guessing credentials.

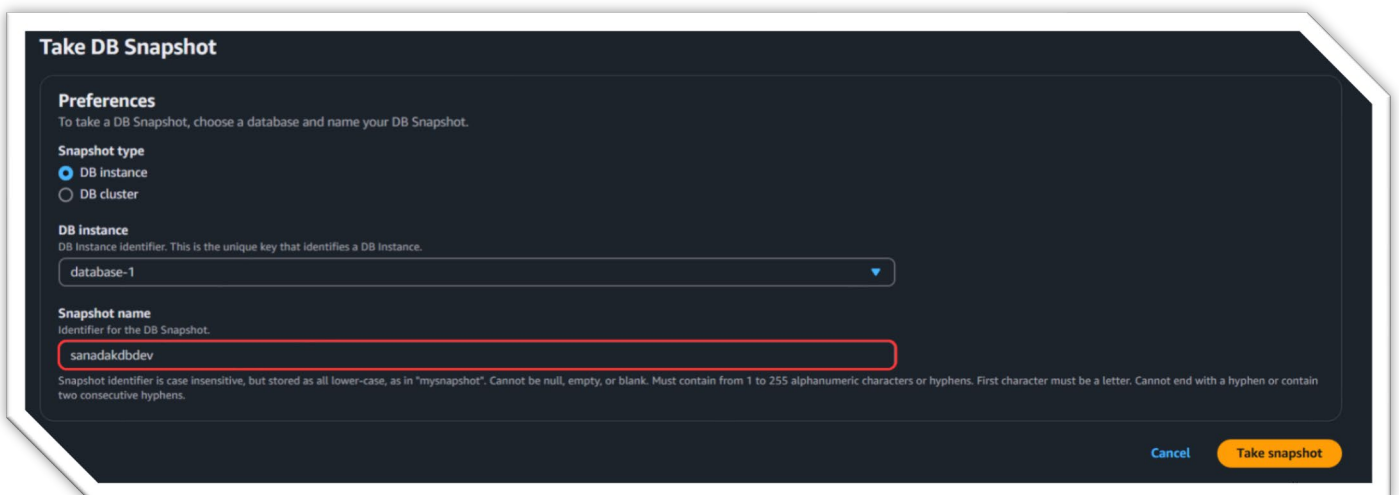
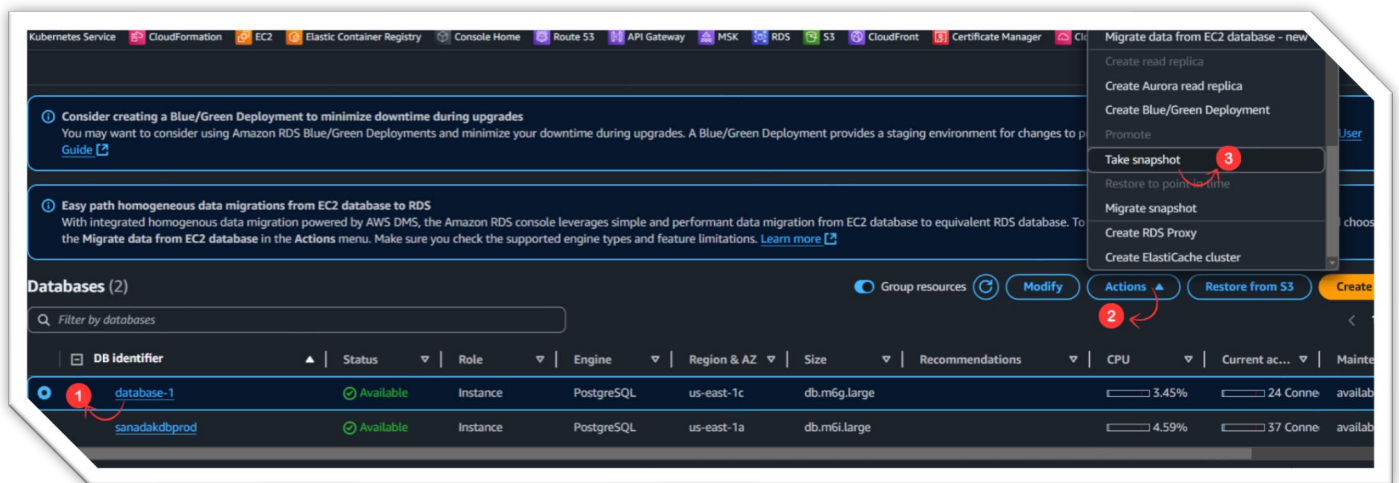


```
2024-11-19 10:04:11 UTC:117.48.192.192(42779):postgres@postgres:[20861]:FATAL: password authentication failed for user "postgres"
2024-11-19 10:04:10 UTC:117.48.192.192(40594):postgres@postgres:[20858]:FATAL: password authentication failed for user "postgres"
2024-11-19 10:04:10 UTC:117.48.192.192(41697):postgres@postgres:[20859]:FATAL: password authentication failed for user "postgres"
2024-11-19 10:04:09 UTC:117.48.192.192(39536):postgres@postgres:[20856]:FATAL: password authentication failed for user "postgres"
2024-11-19 10:04:08 UTC:117.48.192.192(37311):postgres@postgres:[20851]:FATAL: password authentication failed for user "postgres"
2024-11-19 10:04:08 UTC:117.48.192.192(38423):postgres@postgres:[20854]:FATAL: password authentication failed for user "postgres"
2024-11-19 10:04:07 UTC:117.48.192.192(36190):postgres@postgres:[20850]:FATAL: password authentication failed for user "postgres"
2024-11-19 10:04:06 UTC:117.48.192.192(33991):postgres@postgres:[20846]:FATAL: password authentication failed for user "postgres"
2024-11-19 10:04:06 UTC:117.48.192.192(35087):postgres@postgres:[20849]:FATAL: password authentication failed for user "postgres"
2024-11-19 10:04:05 UTC:117.48.192.192(32880):postgres@postgres:[20845]:FATAL: password authentication failed for user "postgres"
2024-11-19 10:04:04 UTC:117.48.192.192(58900):postgres@postgres:[20842]:FATAL: password authentication failed for user "postgres"
2024-11-19 10:04:04 UTC:117.48.192.192(60005):postgres@postgres:[20843]:FATAL: password authentication failed for user "postgres"
2024-11-19 10:04:03 UTC:117.48.192.192(57853):postgres@postgres:[20835]:FATAL: password authentication failed for user "postgres"
2024-11-19 10:04:02 UTC:117.48.192.192(55785):postgres@postgres:[20829]:FATAL: password authentication failed for user "postgres"
2024-11-19 10:04:02 UTC:117.48.192.192(56784):postgres@postgres:[20833]:FATAL: password authentication failed for user "postgres"
2024-11-19 10:04:01 UTC:117.48.192.192(54636):postgres@postgres:[20827]:FATAL: password authentication failed for user "postgres"
2024-11-19 10:04:00 UTC:117.48.192.192(52415):postgres@postgres:[20824]:FATAL: password authentication failed for user "postgres"
2024-11-19 10:04:00 UTC:117.48.192.192(53551):postgres@postgres:[20825]:FATAL: password authentication failed for user "postgres"
2024-11-19 10:03:59 UTC:117.48.192.192(51295):postgres@postgres:[20822]:FATAL: password authentication failed for user "postgres"
```

This document provides a step-by-step guide for creating a private database from old one, connecting it through an EC2 instance, and PGAdmin.

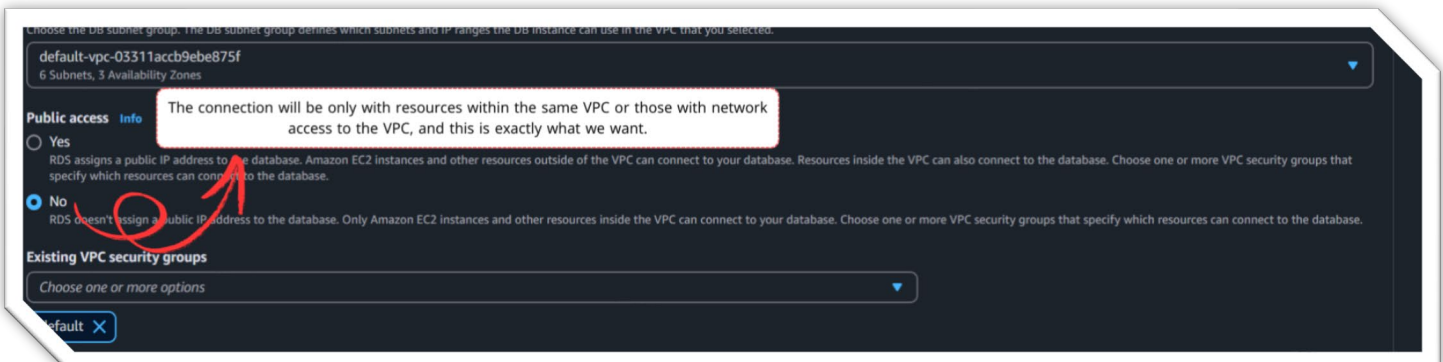
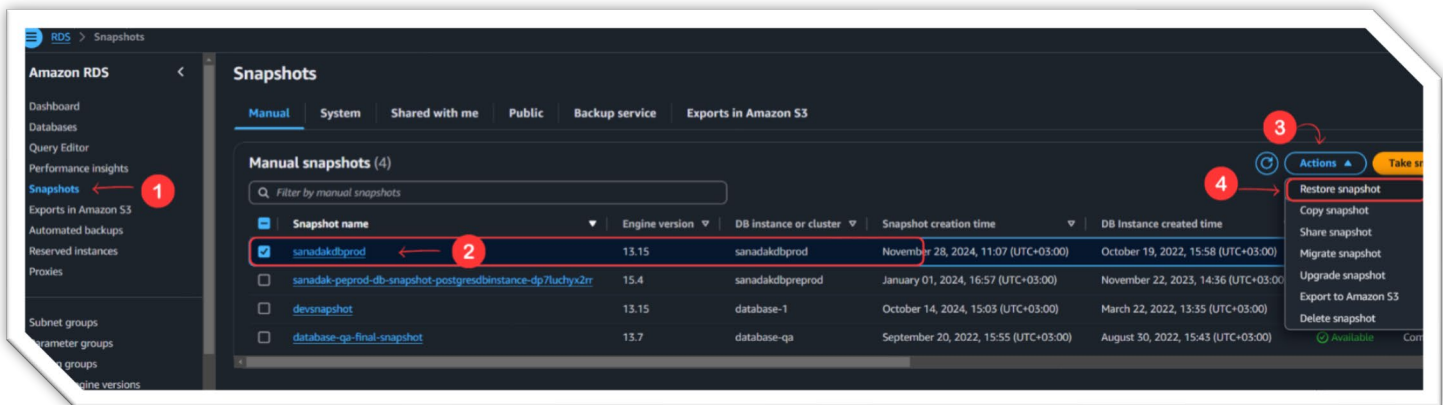
## 2. Creating a Snapshot

1. Log in to the AWS Management Console.
2. Navigate to RDS > Snapshots.
3. Select the desired snapshot of the DEV-DB.
4. Click Actions > Create Snapshot.
5. Provide a name and confirm.
6. Wait until the snapshot creation status changes to "available."



### 3. Restoring a Private Database from the Snapshot

1. Go to the RDS > Snapshots section.
2. Select the desired snapshot.
3. Click Actions > Restore Snapshot.
4. Choose a new DB instance identifier and configure the settings (e.g., instance type, storage, and VPC settings).
5. Ensure that the database is restored in a private subnet for security.
6. Wait until the DB instance status changes to "available."

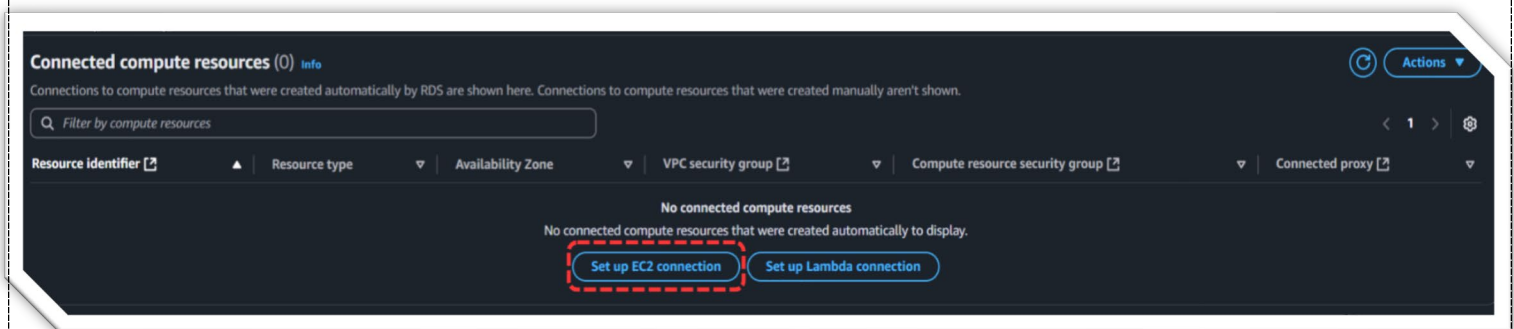


## 4. Connecting to the Database via EC2

To create EC2 instance check this [file](#).

1. Go to the RDS > Database section.
2. Select the desired database.
3. go to Connected compute resources section.
4. Click Set up EC2 connection> Select EC2 instance.

**Note:** Make sure that the EC2 instances and the database are in the same VPC."



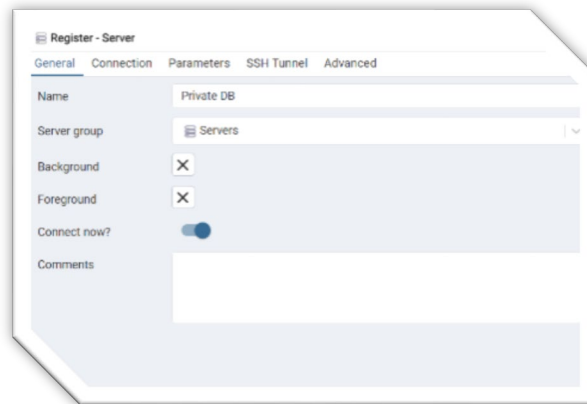
## 5. Assisting with PGAdmin Connection

1. Share the following database details with the user:

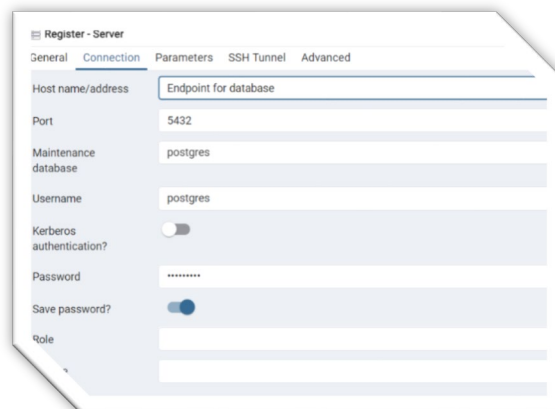
- Host: The private DB endpoint.
- Port: Default PostgreSQL port (5432).
- Username and password.
- Database name.
- Private key.pem "for EC2"

## 2. Guide the user to configure SSH tunneling:

- In PGAdmin, go to File > Add New Server.
- Under the General tab, enter a name for the connection.



- Under the Connection tab:
- Enter 'Endpoint for Database in RDS' as the host.
- Use port '5432'.



- Under the SSH Tunnel tab:
- Enable SSH tunneling and add the IP address for Ec2
- Provide the EC2 instance details (private key file).

The screenshot shows the 'Register - Server' dialog box with the 'SSH Tunnel' tab selected. The 'Use SSH tunneling' toggle is turned on. The 'Tunnel host' field contains 'ip address for EC2', 'Tunnel port' is '22', and 'Username' is 'ubuntu'. Under 'Authentication', the 'Identity file' radio button is selected, with a file path entered in the adjacent field. The 'Password' field is empty, and 'Save password?' is unchecked. The 'Live (seconds)' field is set to '0'.

### 3. Test the connection in PGAdmin.

