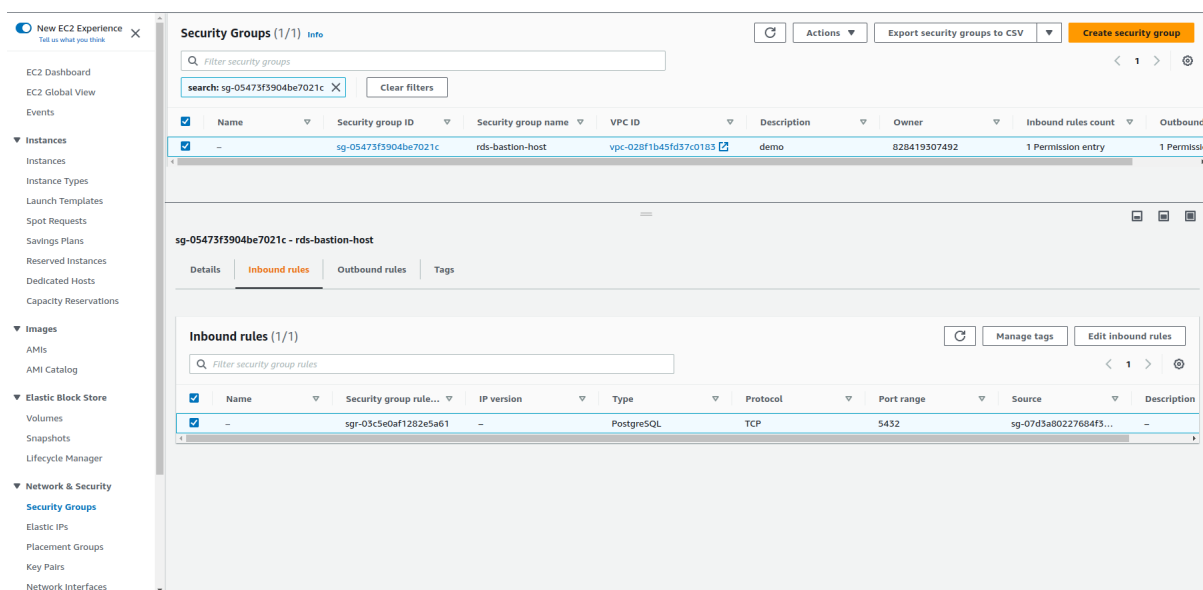


How to connect to RDS instance using AWS Console and Session Manager or AWS CLI with SSM plugin

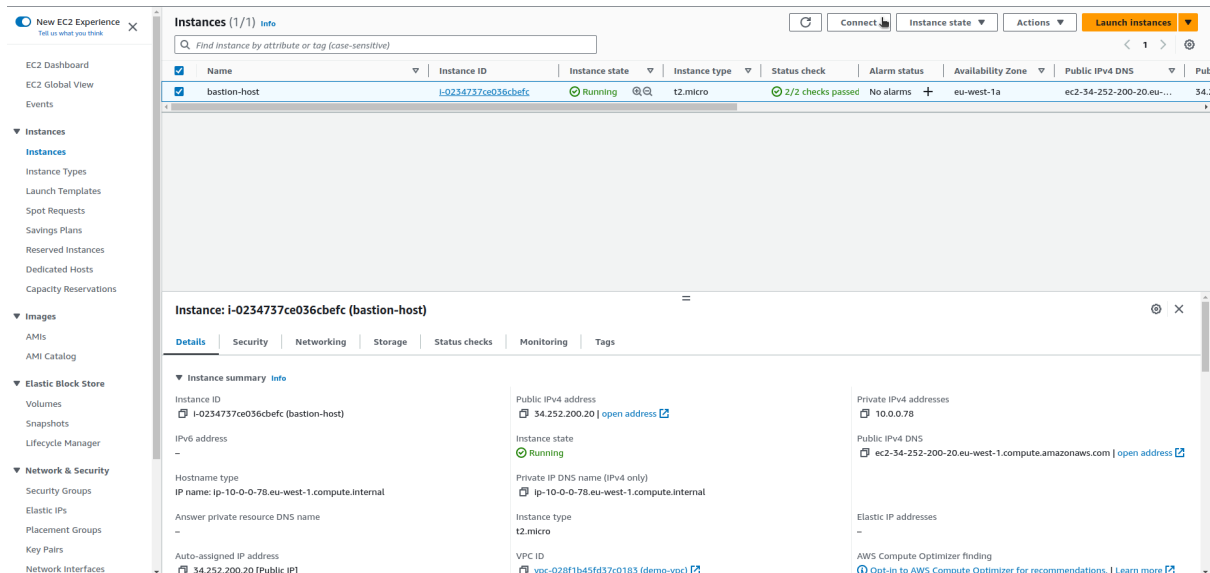
In order to connect to RDS instance through bastion host, there has to be Inbound rule in the database Security group which allows inbound traffic to specific port from private IP or security group of the bastion host.



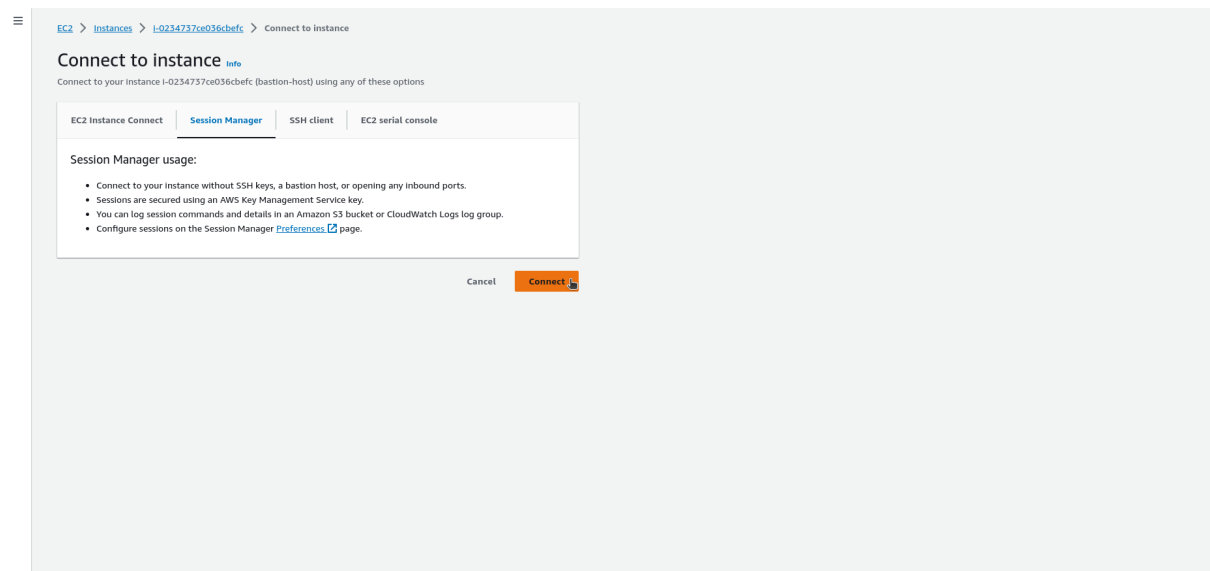
The screenshot displays the AWS Management Console interface for configuring a Security Group. The left sidebar shows the navigation menu with categories like EC2 Dashboard, Images, Elastic Block Store, and Network & Security. The main content area is titled 'Security Groups (1/1)' and shows a table with one security group: 'sg-05473f3904be7021c' named 'rds-bastion-host' with a description 'demo'. Below this, the 'Inbound rules' tab is selected, showing a table with one rule: 'sg-03c5e0af1282e5a61' of type 'PostgreSQL' on port '5432' with source 'sg-07d3a80227684f3...'. The interface includes search filters, action buttons, and pagination controls.

One way to connect is through AWS Console Session Manager:

1. Navigate to EC2 service, choose **bastion-host** instance and click on “**Connect**” in the upper right corner.



2. Choose Session Manager tab and click on “**Connect**”.



3. Now You can connect using, for example, **mysql** or **psql** command:

`psql -h demo.cjbsflodxtzy.eu-west-1.rds.amazonaws.com -U postgres`

Session ID: dawid.tomczynski-03d45aa59ea5de659

Instance ID: i-0234737ce036cbefc

```
[root@ip-10-0-0-78 ~]# psql -h demo.cjbsflodxtzy.eu-west-1.rds.amazonaws.com -U postgres
Password for user postgres:
psql (15.0, server 15.3)
SSL connection (protocol: TLSv1.2, cipher: ECDHE-RSA-AES256-GCM-SHA384, compression: off)
Type "help" for help.

postgres=>
```

The other way to connect is through AWS CLI with SSM plugin installed.

[AWS CLI installation](#)[AWS CLI credentials configuration](#)[SSM plugin installation](#)

AWS SSM plugin, like AWS CLI, uses AWS credentials for connection so it's important to set them up first.

Open the terminal on Your local machine and start the connection between Your local host and RDS instance with this command:

```
aws ssm start-session --document-name AWS-
StartPortForwardingSessionToRemoteHost --parameters '{"host":
["RDS_INSTANCE_ENDPOINT"], "portNumber":["5432"], "localPortNumber":
["5432"]}' --region eu-west-1 --target BASTION_HOST_INSTANCE_ID
```

Change **RDS_INSTANCE_ENDPOINT** to Your RDS endpoint, e.g.
demo.cjbsflodxtzy.eu-west-1.rds.amazonaws.com

Change **BASTION_HOST_INSTANCE_ID** to Your bastion host EC2 instance ID,
e.g. *i-0234737ce036cbefc*

Set RDS instance and local **ports** depending on database engine, and the **AWS region** of RDS instance and bastion host.

```
dawid@dawid-IdeaPad5: ~  
dawid@dawid-IdeaPad5:~$ aws ssm start-session --document-name AWS-StartPortForwardingSessionToRemoteHost --parameters '{"host":["demo.cjbsflodxtzy.eu-west-1.rds.amazonaws.com"], "portNumber":["5432"], "localPortNumber":["5432"]}' --region eu-west-1 --target i-0234737ce036cbefc  
  
Starting session with SessionId: terraform-0f14f84486dc76338  
Port 5432 opened for sessionId terraform-0f14f84486dc76338.  
Waiting for connections...  
  
Connection accepted for session [terraform-0f14f84486dc76338]  
█
```

Next You will be able to connect to the database with, for example, **mysql** or **psql** command in a new terminal or any database client with **localhost** as host:

psql -h 127.0.0.1 -U postgres

```
dawid@dawid-IdeaPad5: ~  
dawid@dawid-IdeaPad5:~$ psql -h 127.0.0.1 -U postgres  
Password for user postgres:  
psql (14.9 (Ubuntu 14.9-0ubuntu0.22.04.1), server 15.3)  
WARNING: psql major version 14, server major version 15.  
         Some psql features might not work.  
SSL connection (protocol: TLSv1.2, cipher: ECDHE-RSA-AES256-GCM-SHA384, bits: 256, compression: off)  
Type "help" for help.  
  
postgres=> █
```