**Graphical user interface, text

Description automatically generated with medium confidence**

**Purpose**

Relational Database Service (RDS) is a popular AWS tool that allows a platform to create databases on the web. This database can be integrated alongside other AWS tools such as EC2 in order to provide full front-end and back-end environments for an application. This expands the sphere of AWS tools and offers another traditional data center service on the cloud.

**Lab Summary**

An Aurora/MySQL database was created and tested. A security group was created to make sure a predefined EC2 instance was able to access the data from the database and display to the front-end interface. The setup was tested through the web application pre-installed on the EC2 instance.

**Create Security Group:**

1. In the EC2 service, click the Security Groups in the left menu.

2. Click create security group.

3. Under basic details, configure it with the following:

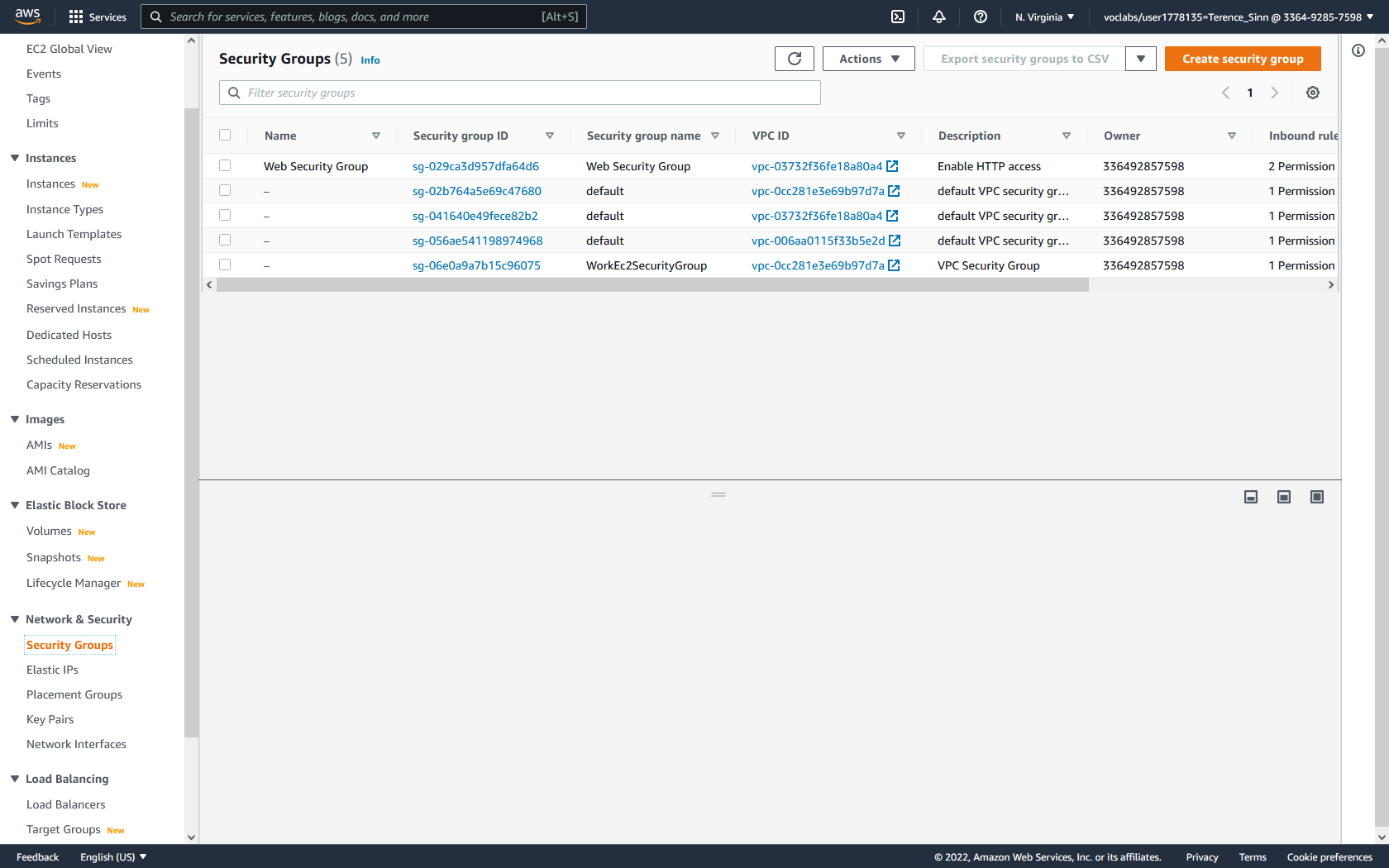
Security Group Name: DB Security Group

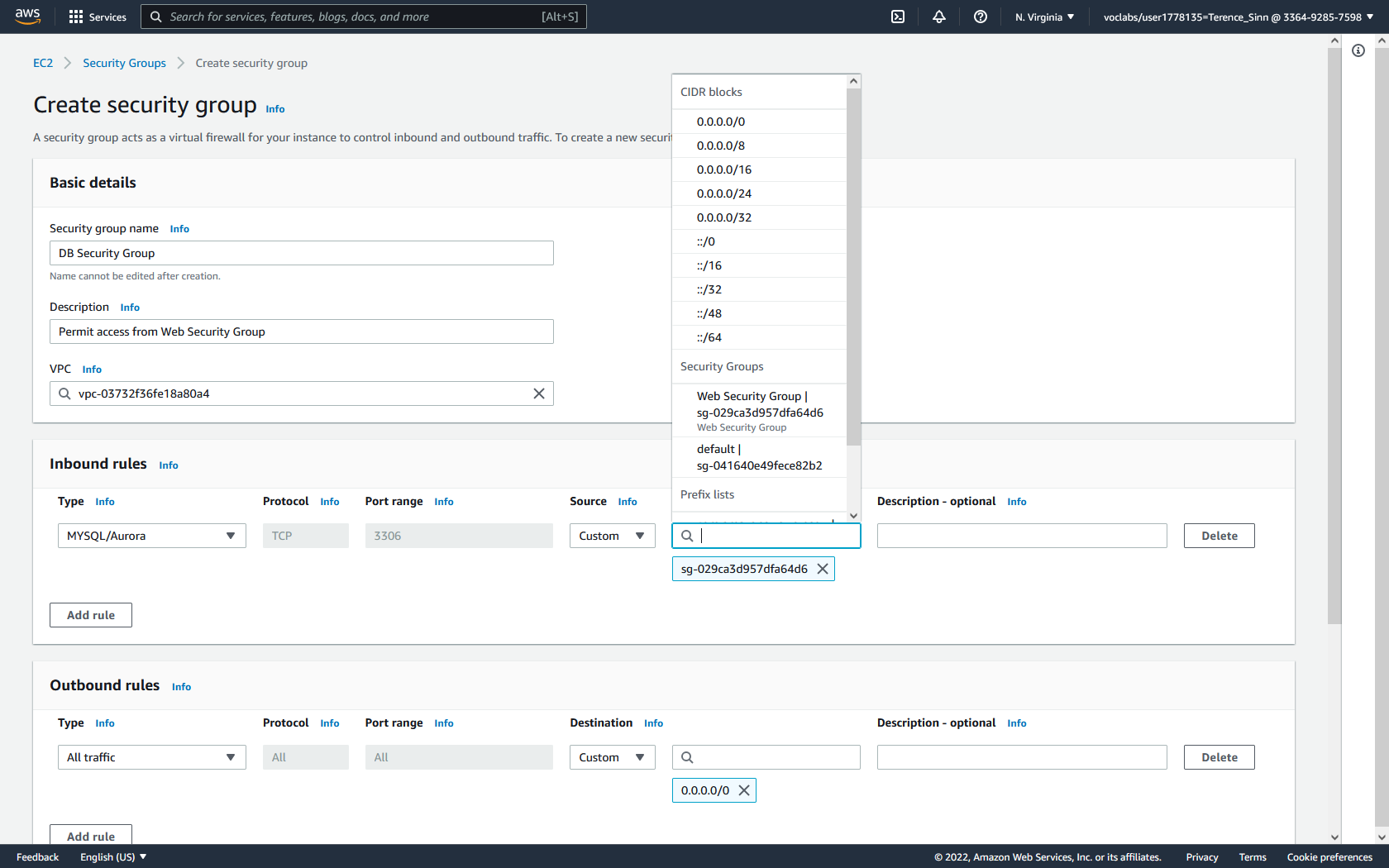
Description: Permit Access from Web Security Group

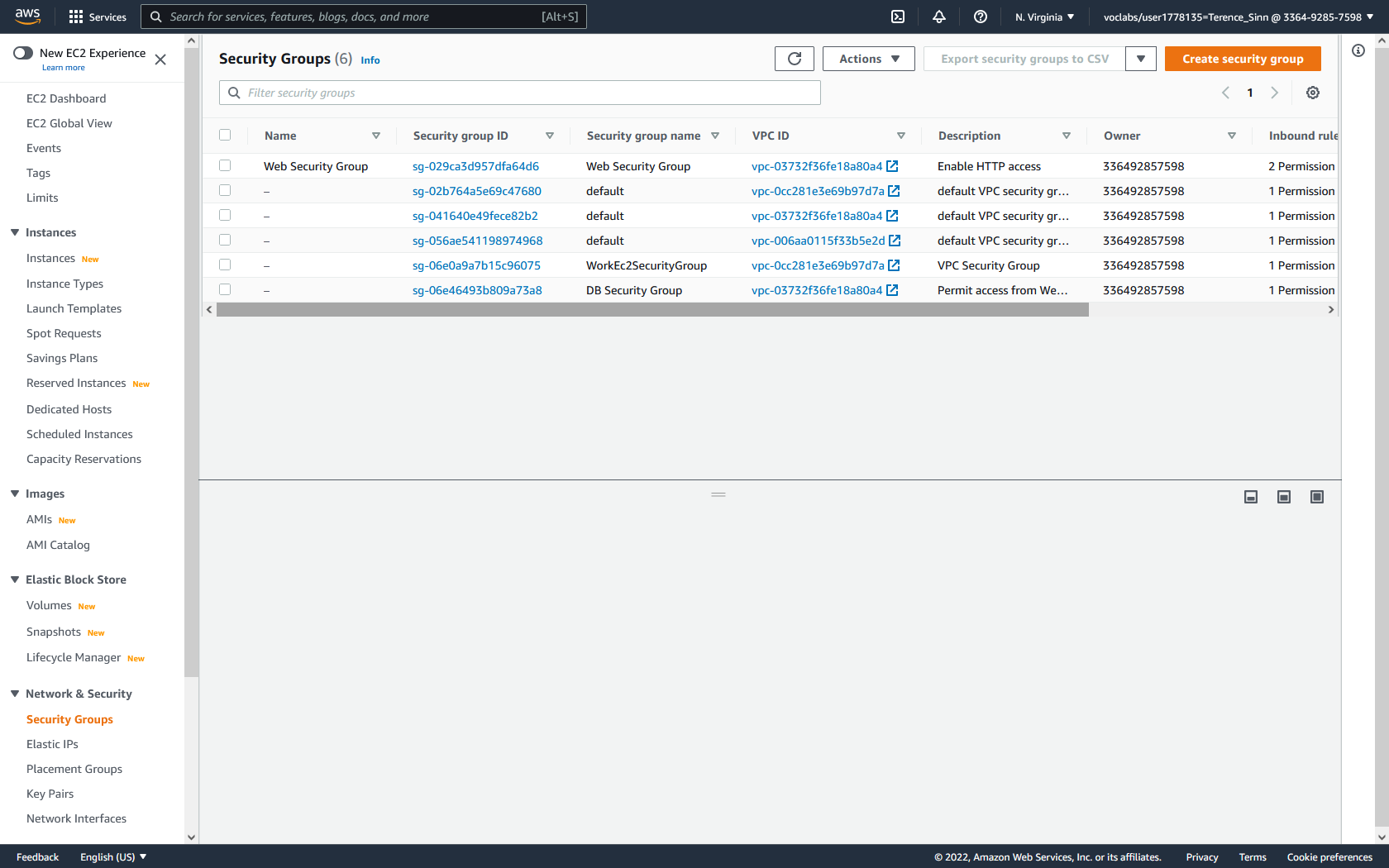
VPC: Lab VPC

4. Under inbound rules, create an inbound rule with the type of MySQL/Aurora(3306) and with the source IPv4 address of the Web Security Group.

5. Click create security group.







**Create Subnets:**

1. Locate RDS by typing RDS into the services search bar at the top of the page. Then click RDS to reach the RDS dashboard. Then click Subnet groups in the left menu to reach the Subnet groups dashboard.

2. Under subnet group details, configure the following:

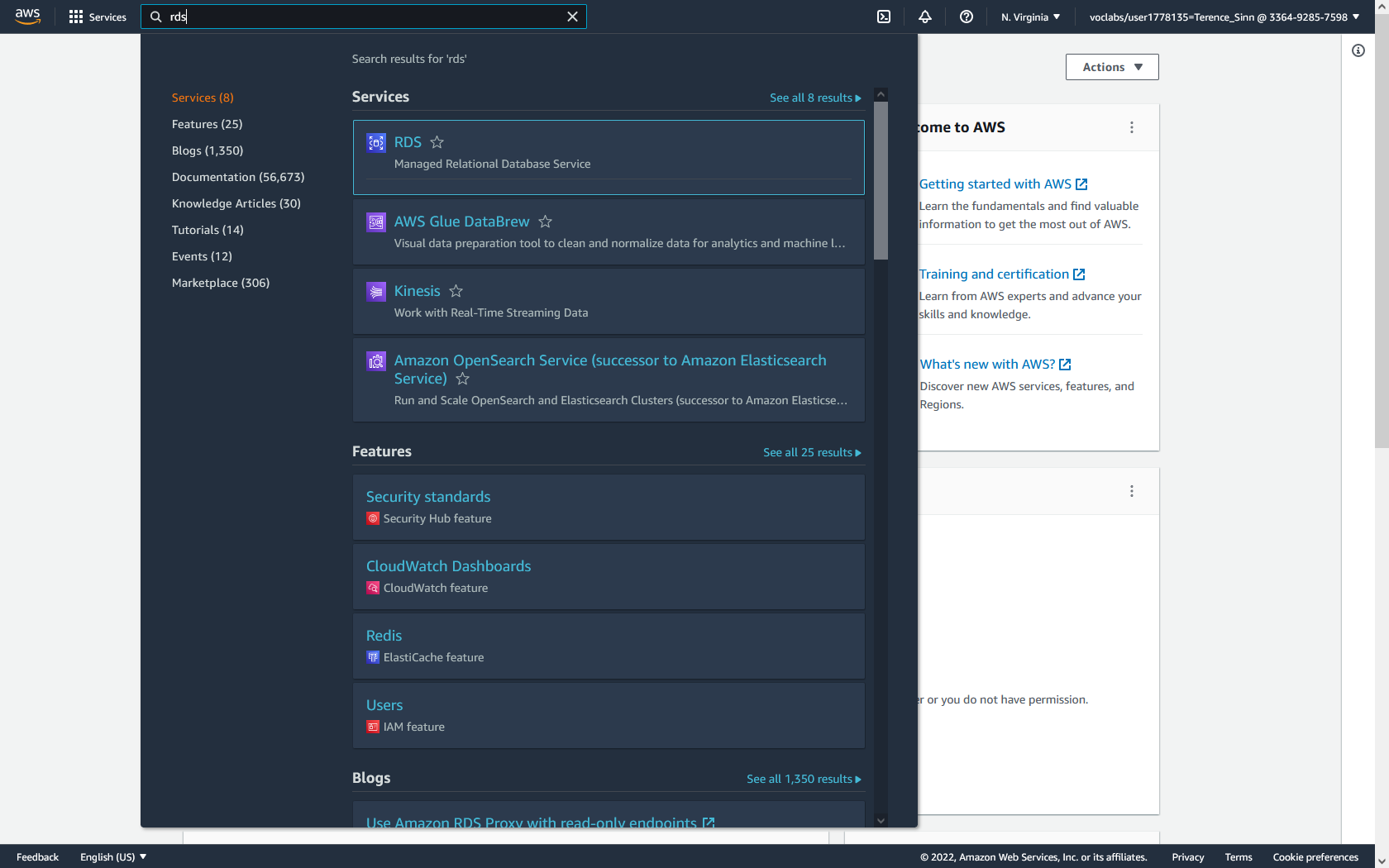
Name: DB-Subnet-Group

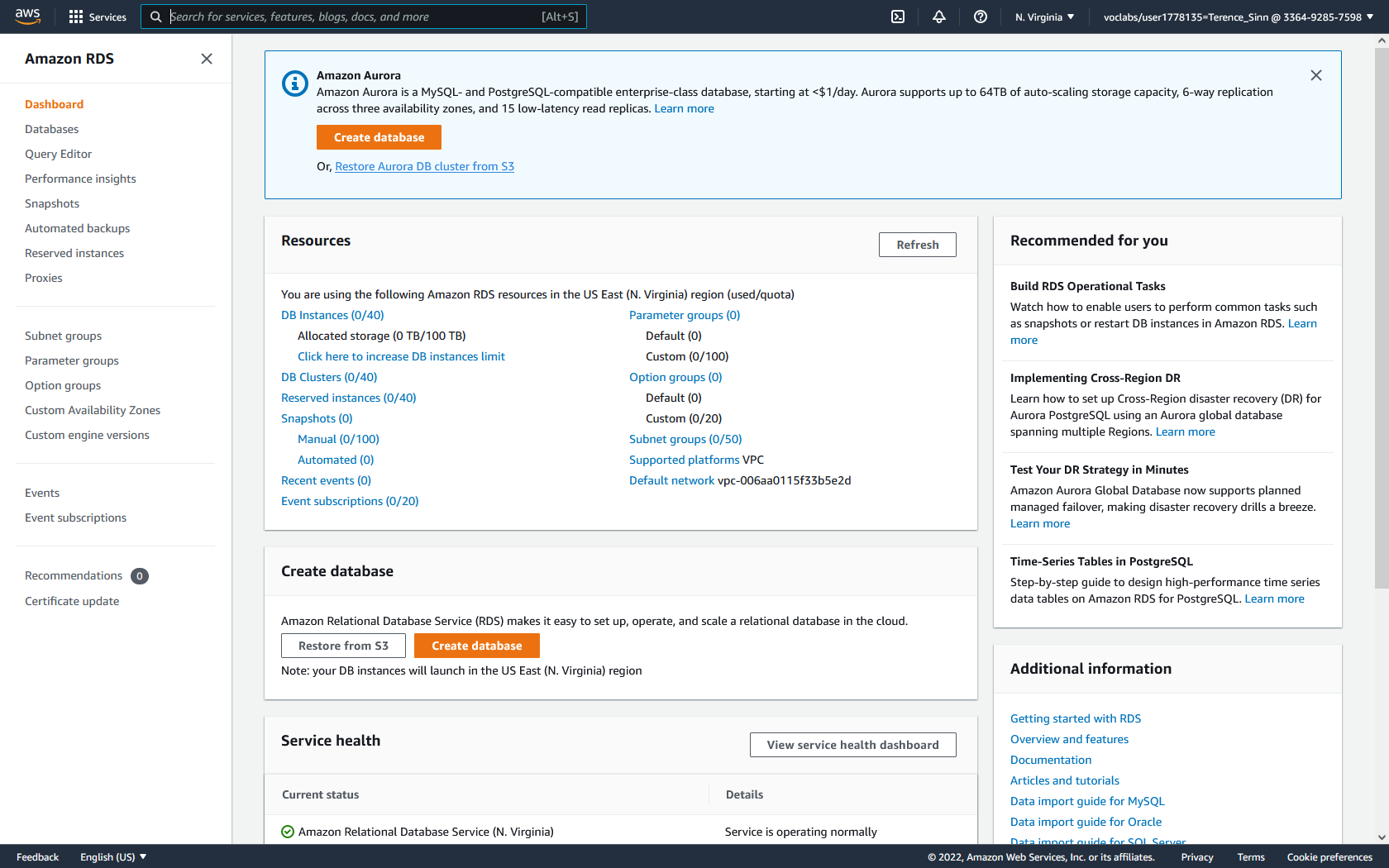
Description: DB Subnet Group

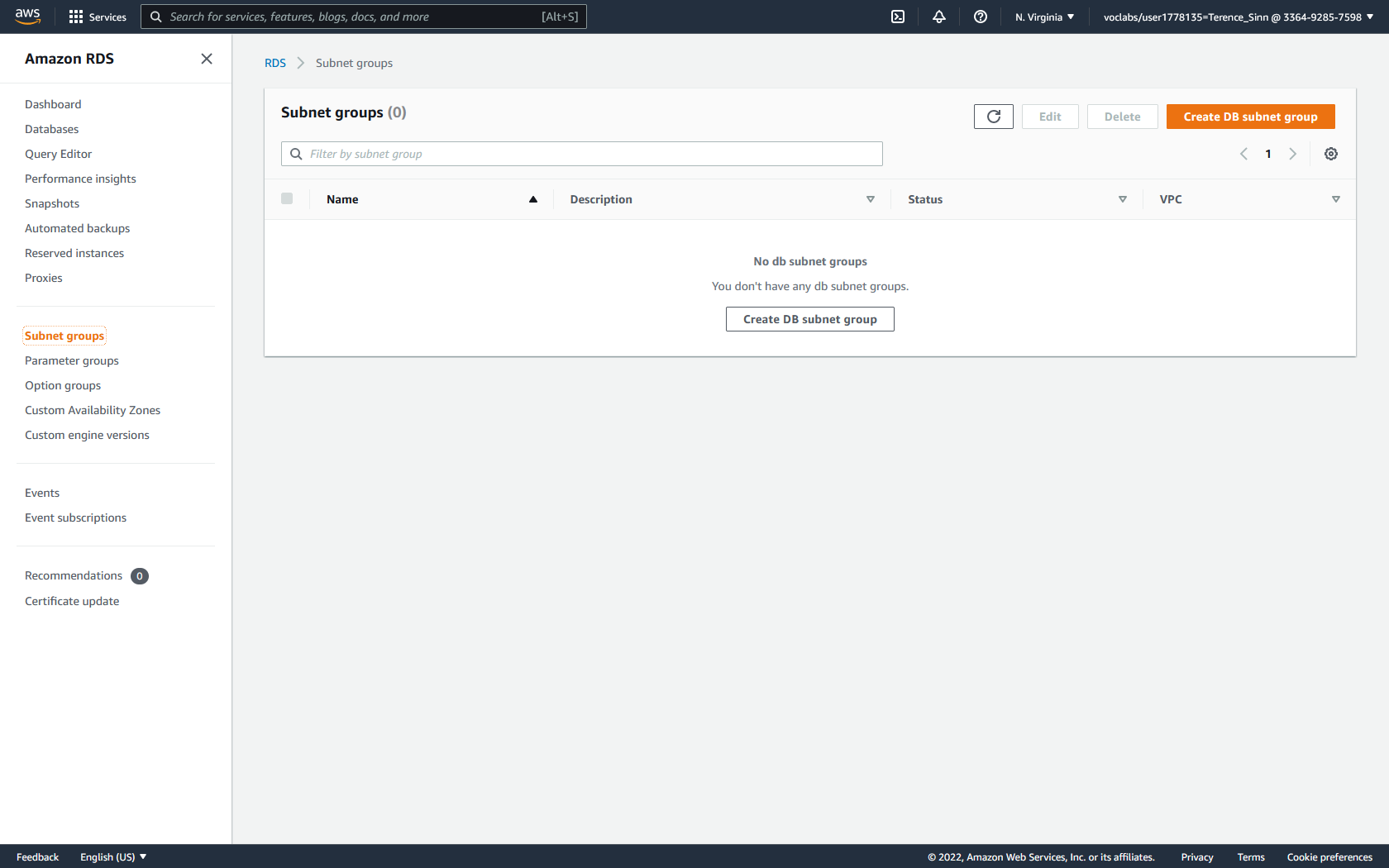
VPC: Lab VPC

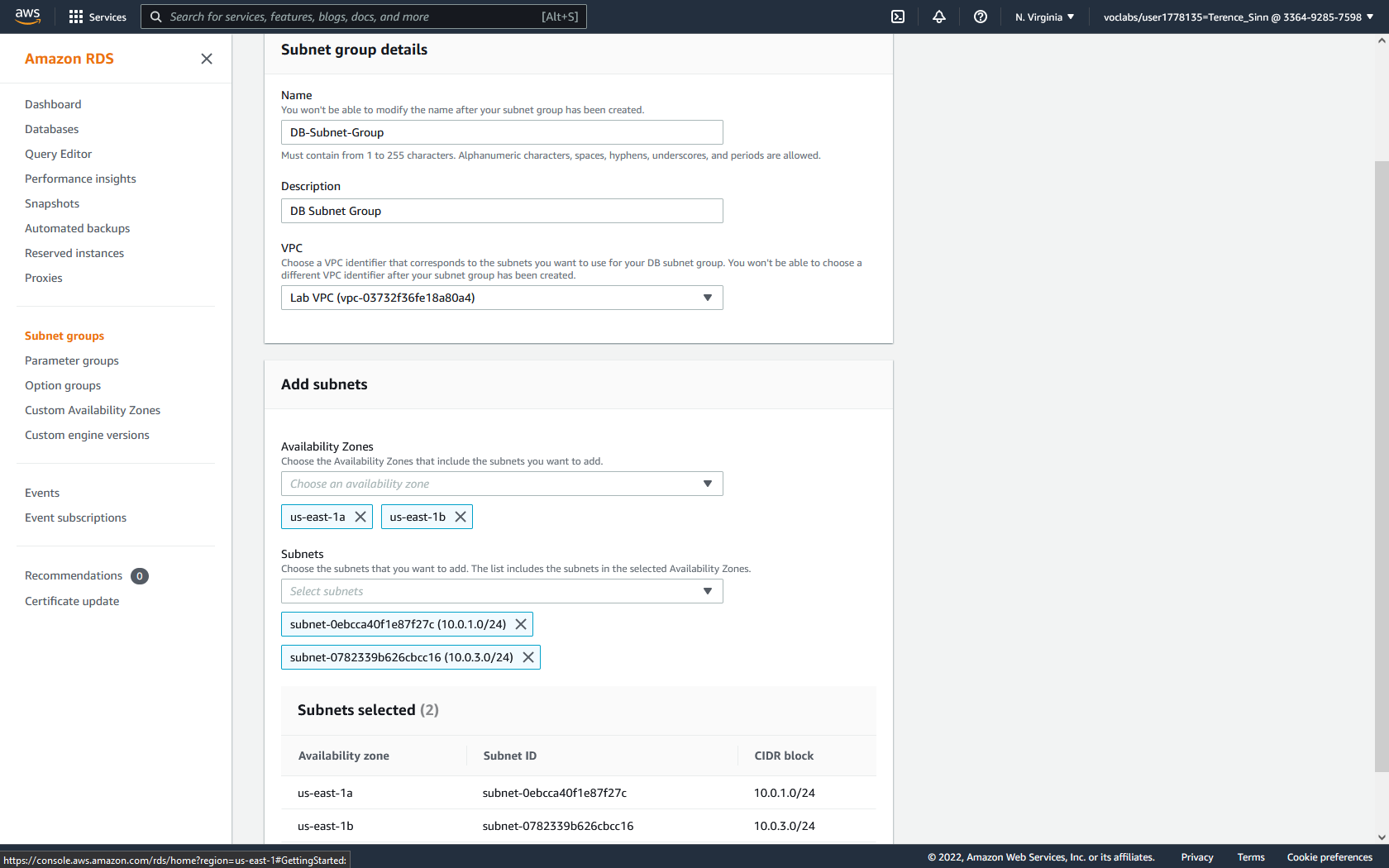
3. Under add subnets, select both the us-east1a and us-east1b availability zones. Then choose the subnets 10.0.1.0/24 and 10.0.3.0/24. These should be private networks since databases should not be reachable directly but indirectly through a web server.

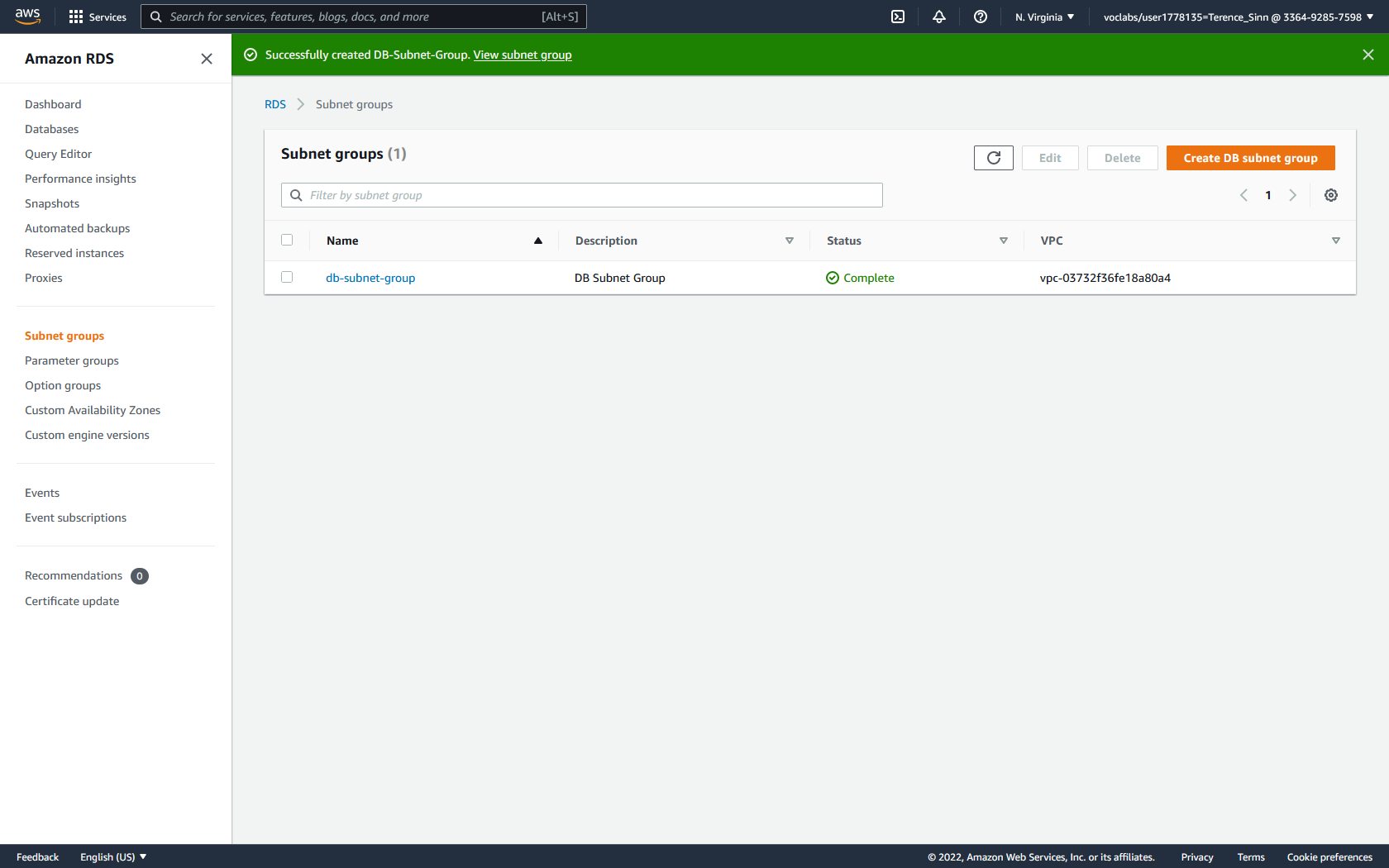
4. Click create.











**Launch Database Server:**

1. Click databases on the left menu to reach the databases dashboard. Click create databases. Select MySQL.

2. Under Settings, configure the following information:

DB Instance Identifier: lab-db

Master Username: main

Master Password: lab-password

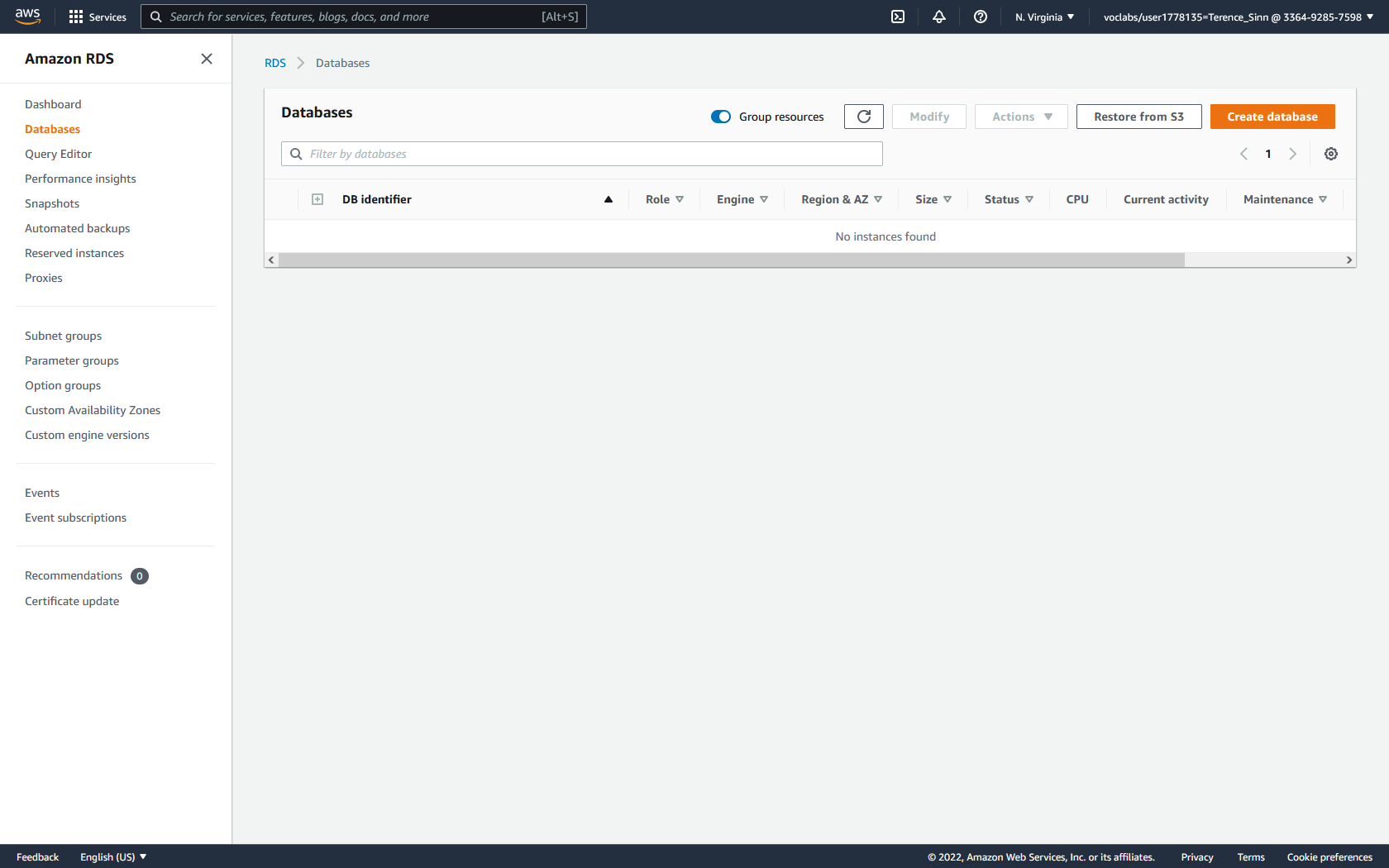
3. Under DB Instance class, select burstable classes and choose the db.t3.micro instance type.

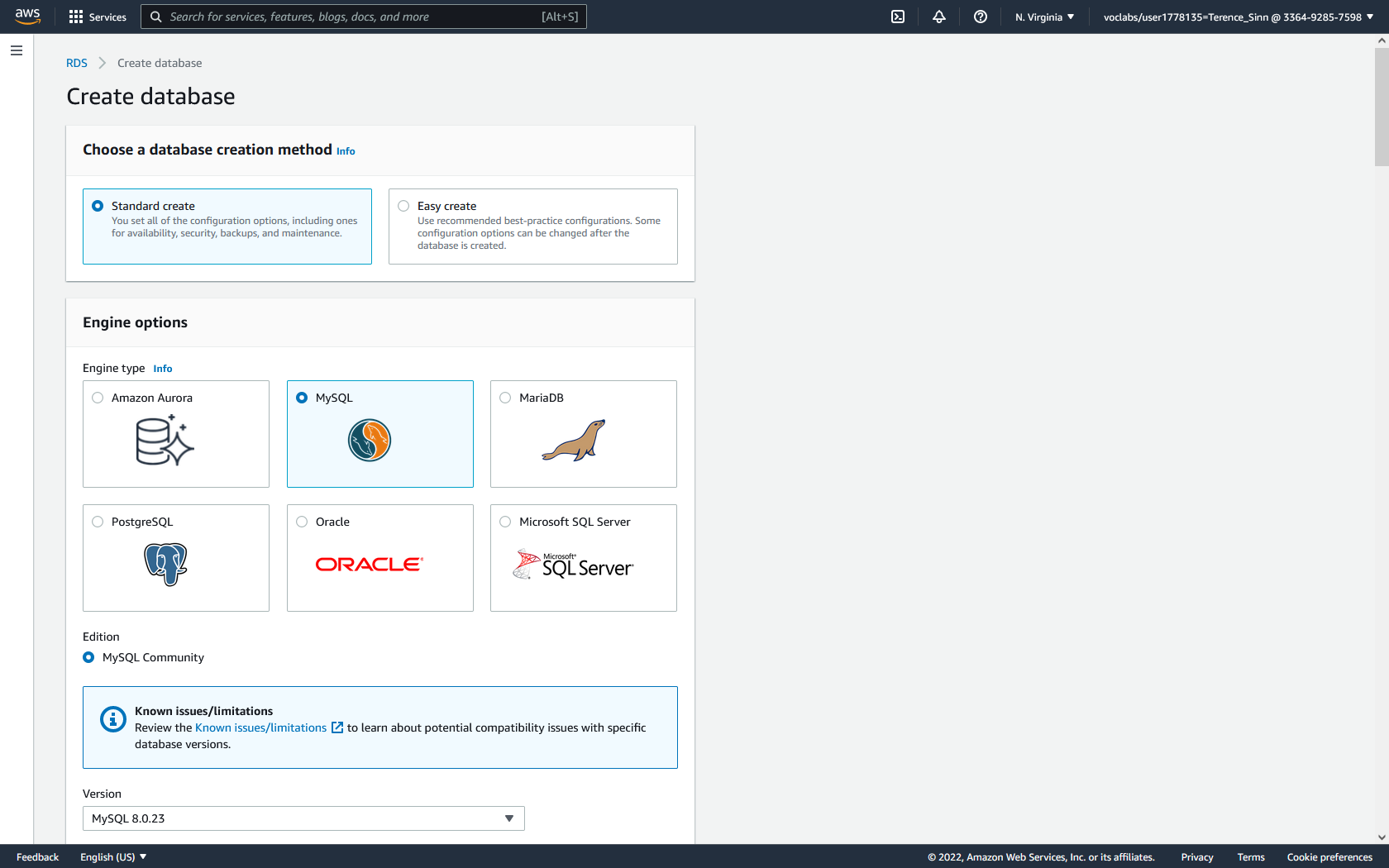
4. Under Storage, choose a General Purpose SSD with 20 Gib of storage.

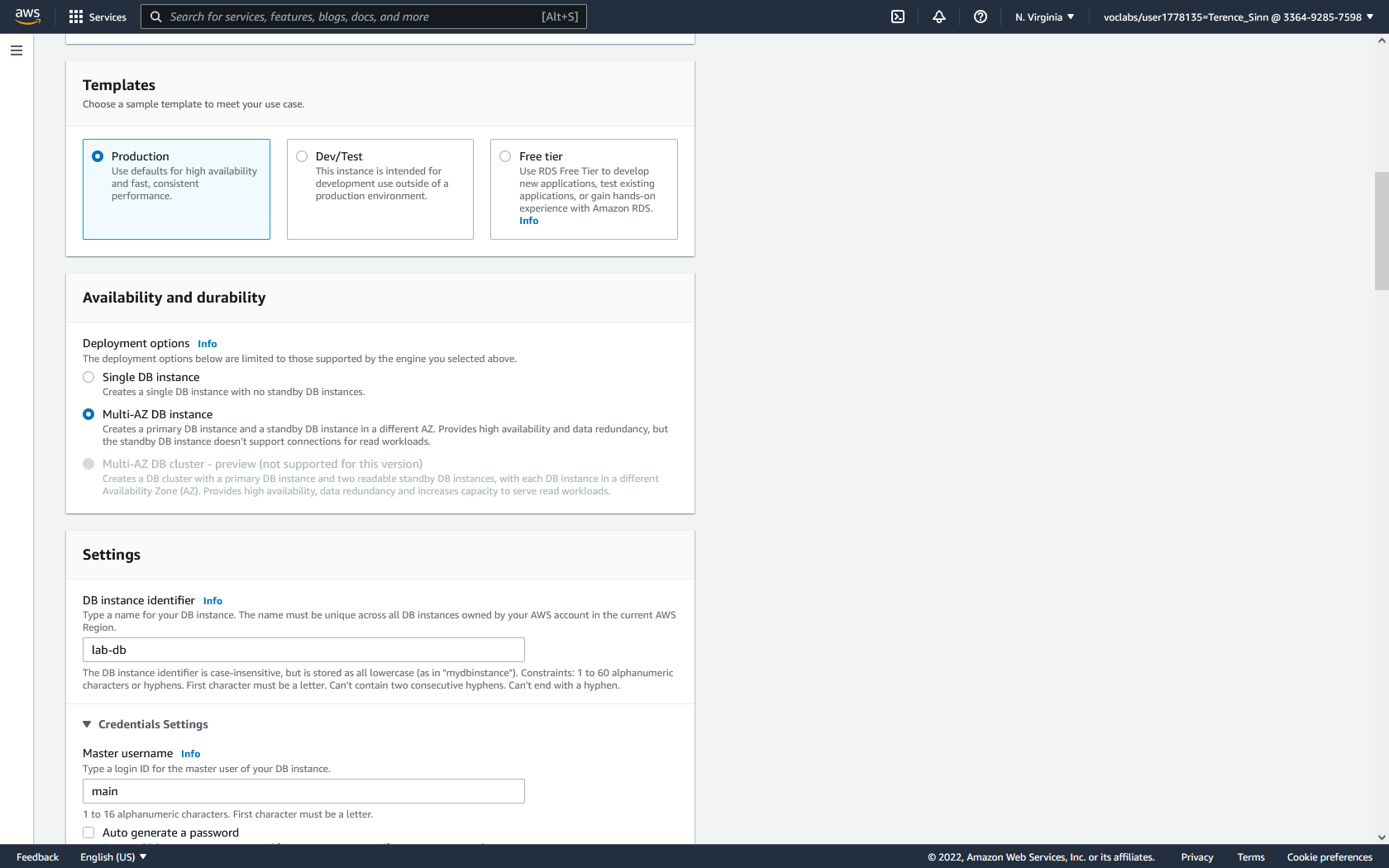
5. Under Connectivity, select the database to use the Lab VPC. Then use the existing DB Security Group.

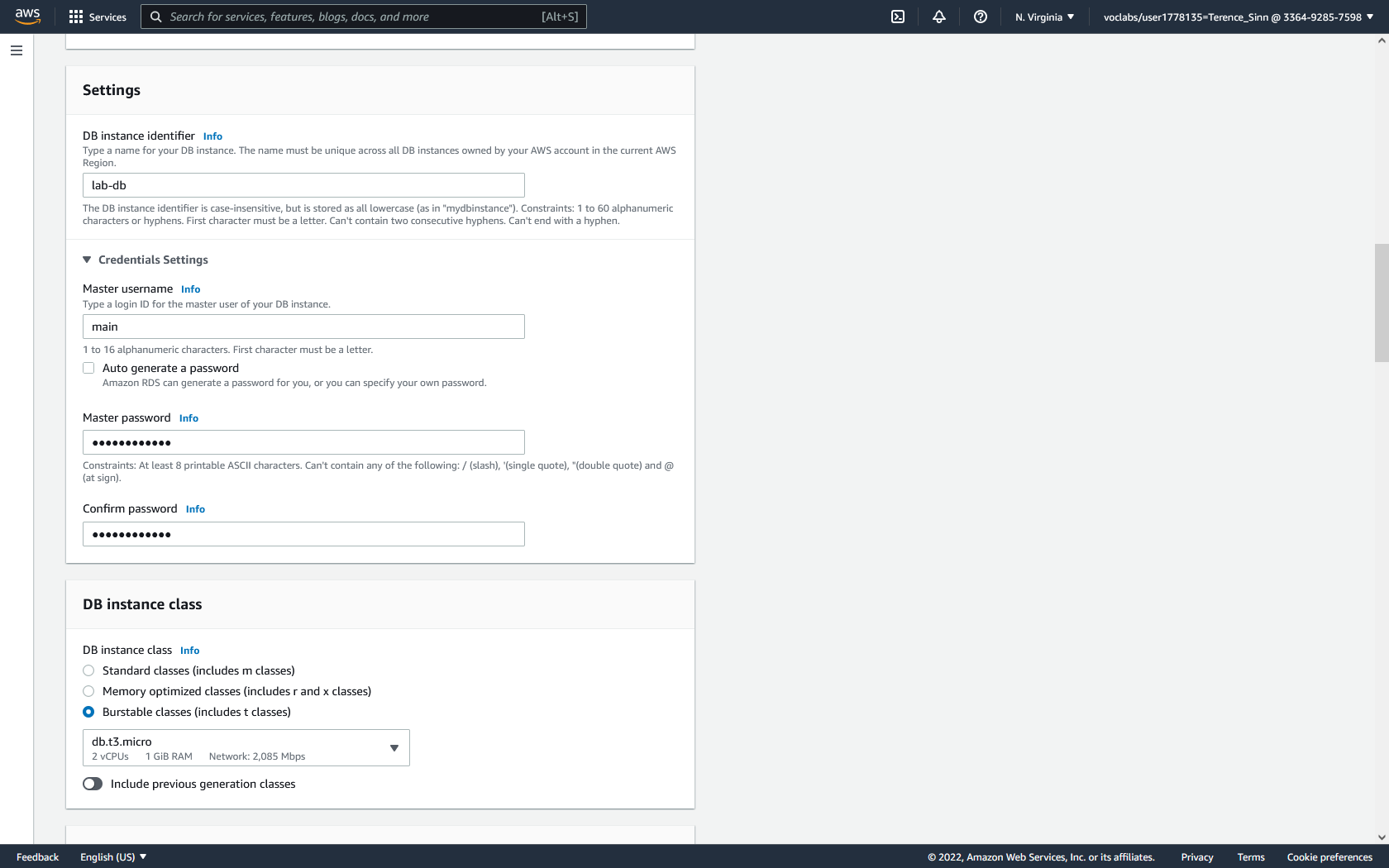
6. For lab purposes, under Additional Configuration, give the database an Initial Database Name of lab. Then uncheck enable automatic backups, enable encryption, and enable enhanced monitoring. This will speed up the creation process of the database instance.

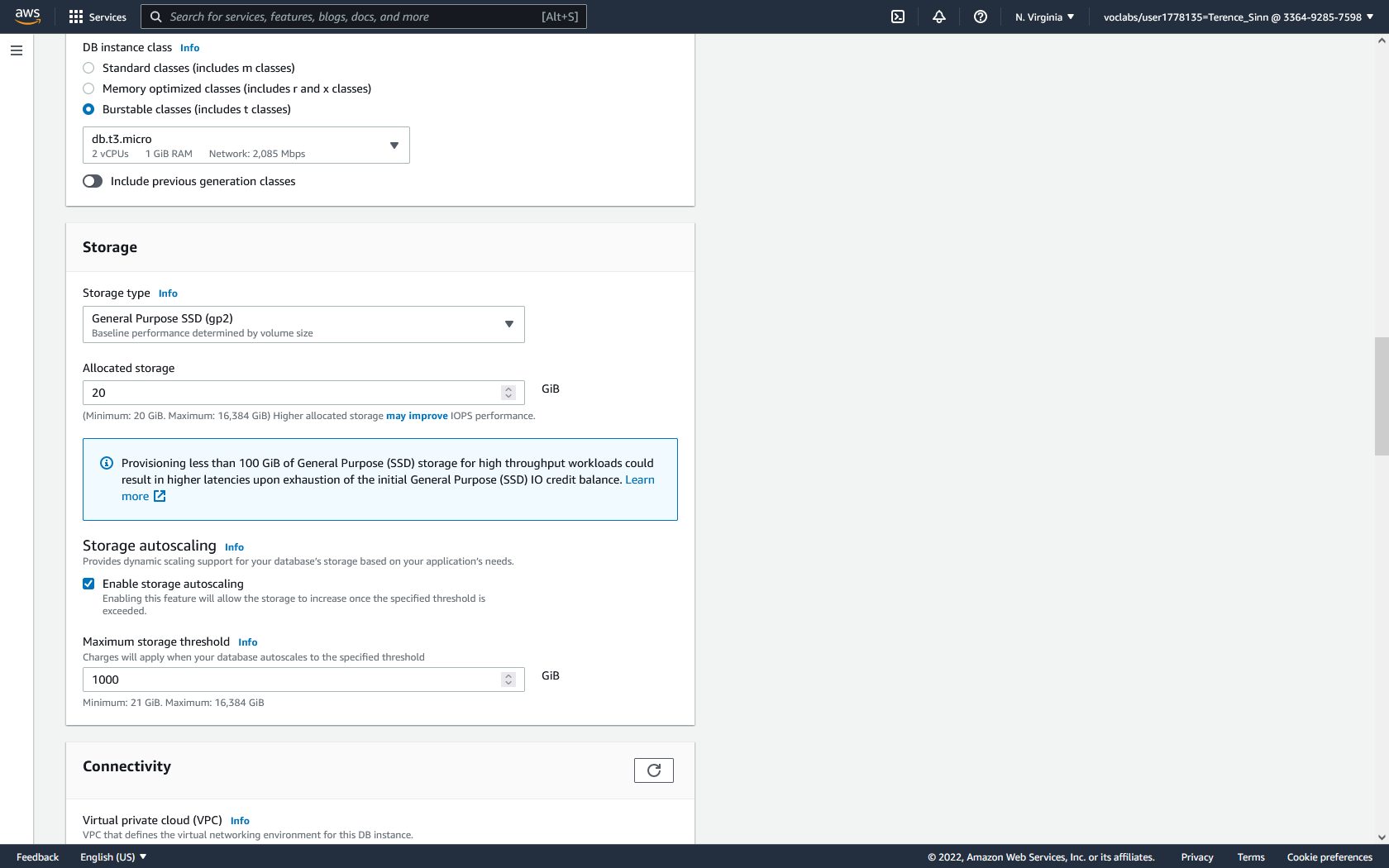
7. Click create database. Go back to the databases dashboard and wait until the database is seen as available.

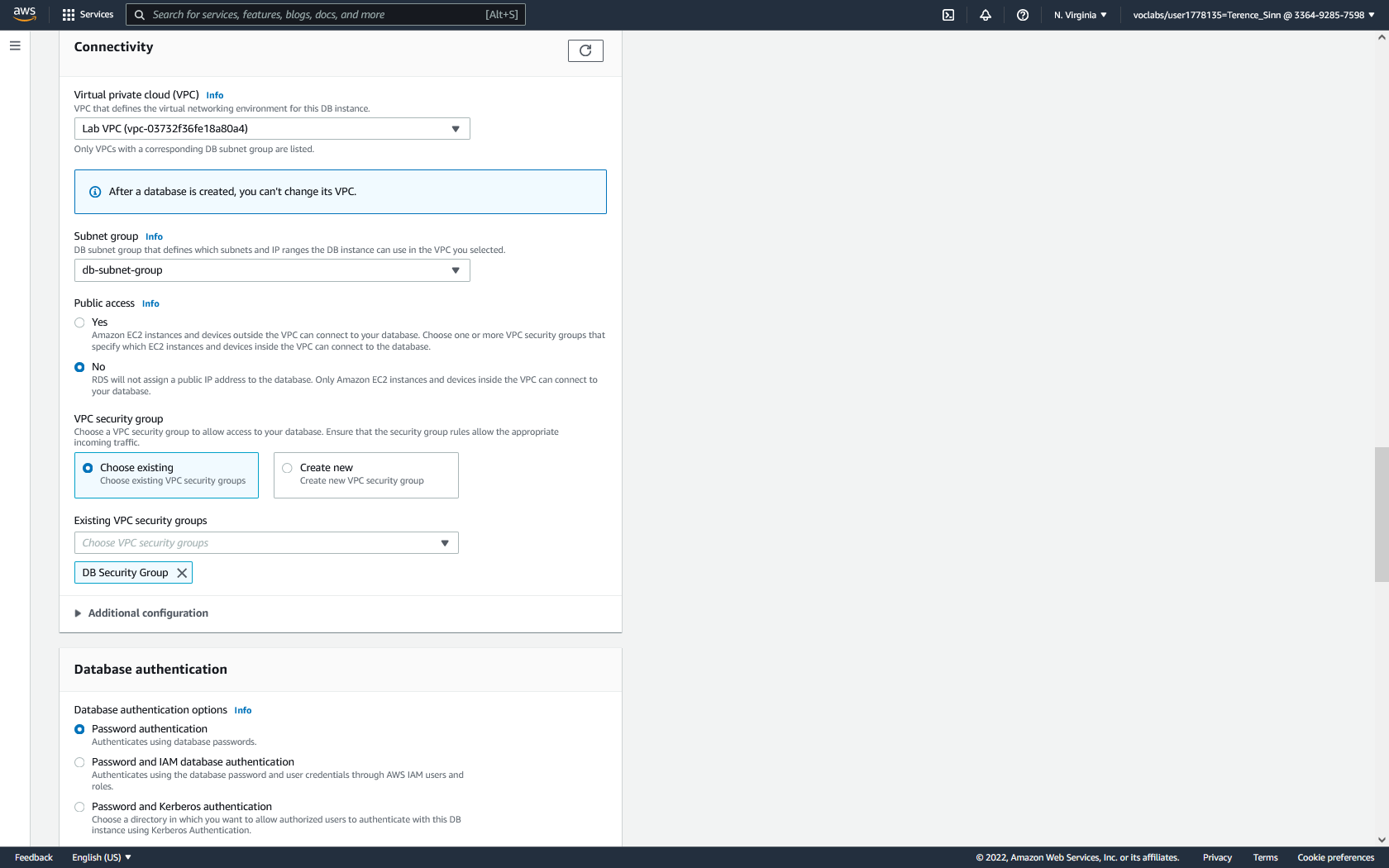


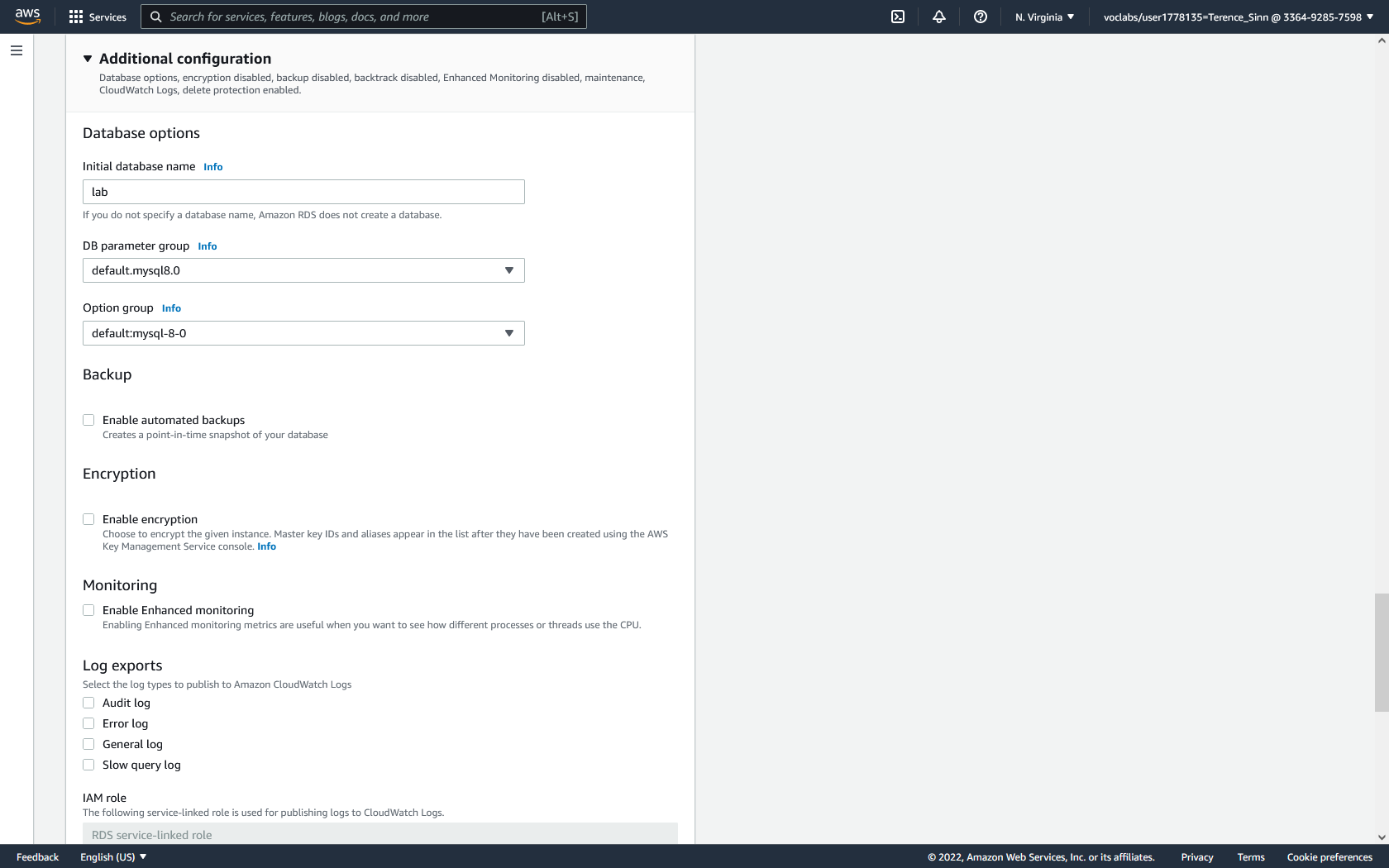


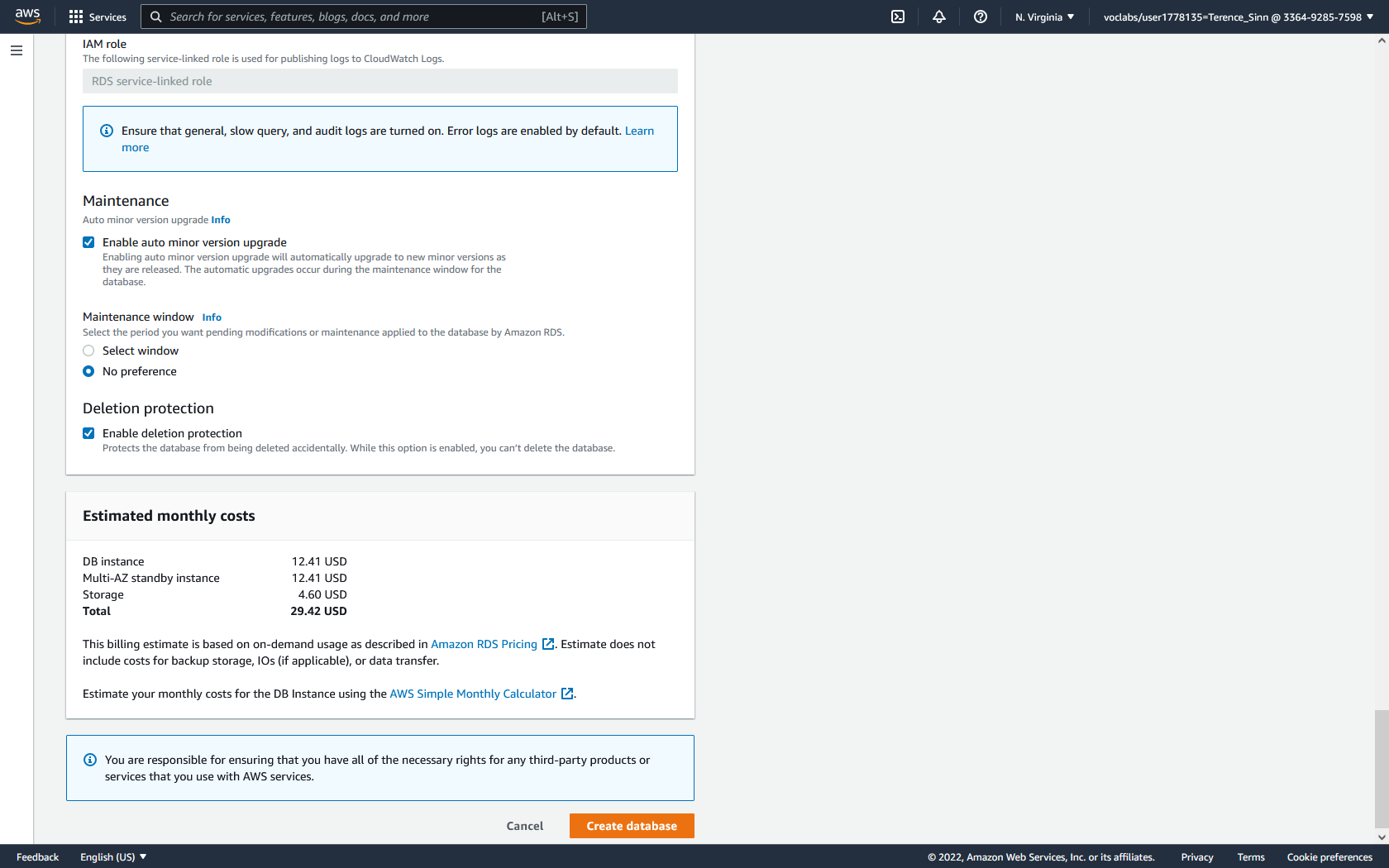


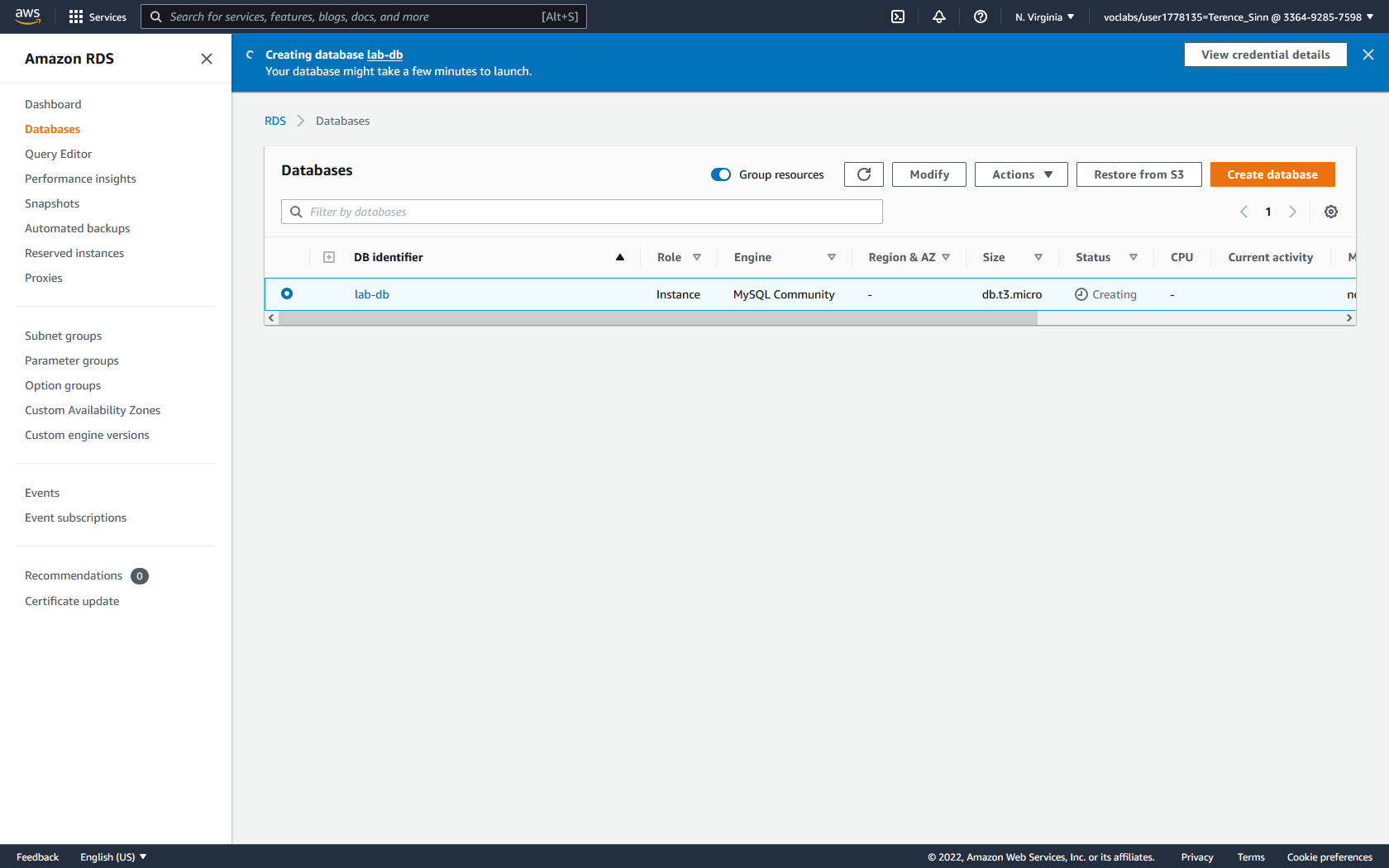


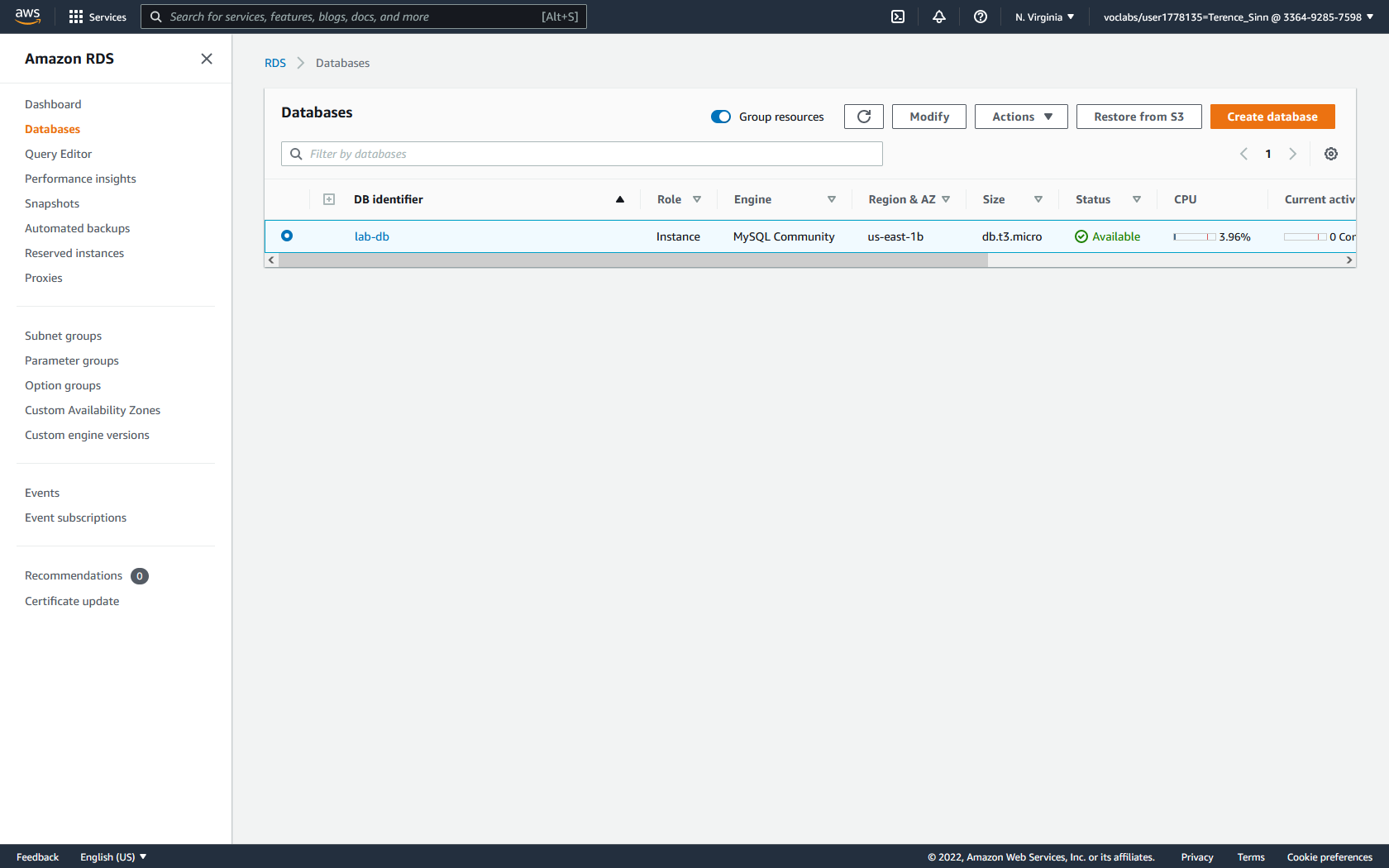












**Test Database Server:**

1. Click into the lab-db to view further detail. Under the connectivity & security tab, locate and copy the endpoint URL into a text file.

2. Go back to the EC2 services and reach the instances dashboard. Click into the Web Server to view further details. Copy its public IPv4 address and paste it into a web browser.

3. After the web page loads, click the RDS button in the top banner.

4. Login using the following information:

Endpoint Security: the previously copied endpoint URL of the lab-db

Database: lab

Username: main

Password: lab-password

5. Change the address book by editing or creating new entries to test the functionality of the database.

