QATIP Intermediate
AWS Lab08
Deploy SQL Server on AWS

Contents

Overview

This lab will guide you through deploying an AWS EC2 instance running SQL Server 2022 Standard using Terraform. You will be provided with a skeleton Terraform file and must complete the missing sections following the outlined steps.

By the end of this lab, you will:

- Define and configure an AWS Virtual Private Cloud (VPC).

- Create a subnet for the EC2 instance.
- Assign an Elastic IP and configure networking.
- Deploy a Windows-based EC2 instance with SQL Server 2022 Standard Edition.
- Implement security rules to allow RDP and SQL access.
- Upload and restore a sample database backup (demo.bak).
- Output the public IP of the deployed instance.

Before you begin

Ensure you have completed Lab0 before attempting this lab.

In the IDE terminal pane, enter the following command...

```
cd /aws-tf-int/labs/08
```

This shifts your current working directory to labs/08. Ensure all commands are executed in this directory

Close any open files and use the Explorer pane to navigate to and open the labs/08 folder.

Solution

There are no step-by-step instructions for this lab. A solution can be at /aws-tf-int/labs/solutions/08 but try to use this only as a last resort if you are struggling to complete the steps outlined below.

Reference Documentation

Terraform AWS Provider Documentation

AWS EC2 Documentation

SQL Server on AWS

Scenario

You are a cloud engineer setting up a **SQL Server 2022 Standard** instance on AWS. The company requires a cost-effective solution using a small EC2 instance while ensuring secure access via RDP and SQL ports.

Your goal is to:

- Define AWS infrastructure components in Terraform.
- Deploy a Windows-based EC2 instance with **SQL Server 2022 Standard Edition**.
- Configure security groups to allow controlled access.
- Upload and restore a sample database backup (demo.bak).
- Extract and display the public IP of the EC2 instance.

Instructions

You are provided with a partial Terraform script that includes only the **Provider** block in **/aws-tf-in/labs/09**. Your task is to complete the missing sections.

Step 1: VPC and Subnet

Create a **VPC** (aws_vpc) named vpc-sql with CIDR block 10.0.0.0/16.

Create a **Subnet** (aws_subnet) named subnet-sql with CIDR block 10.0.1.0/24.

Step 2: Elastic IP and Network Interface

Create an **Elastic IP** (aws_eip) named sql-eip.

Create an **Elastic Network Interface (ENI)** (aws_network_interface) named sql-eni.

Associate the **Elastic IP** with the **Network Interface**.

Step 3: EC2 Instance

Create an **EC2 instance** (aws_instance) named sql-server-ec2.

Use **Windows Server 2022 with SQL Server 2022 Standard**:

AMI ID: Use an AMI specific to your AWS region.

Instance Type: t3.medium.

Root Volume: 32GB with **gp3** storage type.

Security Group: Associate the EC2 instance with the security group created in Step 4.

Set administrator credentials:

Username: adminuser

Password: your_choice

Step 4: Security Groups

Create a **Security Group** (aws_security_group) named sql-sg.

Add inbound rules to allow:

RDP (port 3389) from any IP.

SQL Server (port 1433) from any IP.

Attach this security group to the EC2 instance.

Step 5: Output the Public IP Address

Modify the **output** block in Terraform to display the **public IP** of the instance.

Step 6: Deploy the Resources

Run the following commands to deploy the infrastructure:

```
terraform init
```

```
terraform apply -auto-approve
```

Step 7: Upload and Restore Database Backup

Retrieve the **Public IP** from Terraform output.

Use **RDP** to connect to the EC2 instance:

Open **Remote Desktop (mstsc.exe)**.

Enable **Local Drives** access.

Enter the **Public IP** and log in using adminuser credentials.

Copy demo.bak from your local machine to C:\Backup\demo.bak on the EC2 instance.

Restore the **Bank** database using SQLCMD or **SQL Server Management Studio (SSMS)**:

```
RESTORE DATABASE Bank
FROM DISK = 'C:\backup\demo.bak'
WITH
  MOVE 'Bank' TO 'C:\Program Files\Microsoft SQL Server\MSSQL16.MSSQLSERVER\MSSQL\DATA\Bank.mdf',
  MOVE 'Bank_log' TO 'C:\Program Files\Microsoft SQL Server\MSSQL16.MSSQLSERVER\MSSQL\DATA\Bank.ldf',
  REPLACE,
  RECOVERY,
  STATS = 10;
```

Step 8: Lab Cleanup

To remove all deployed resources by running **terraform destroy -auto-approve**

### Congratulations! You have completed this lab. ###