Charlie Qu

Period 0-2

Mason/Hansen

1/10/23

***AWS Labs 4-6***

A logo with a smile

Description automatically generated  
Purpose:

Lab 4: EBS Operations - This lab focuses on creating an Amazon EBS volume that can be connected and mounted on an EC2 instance.

Lab 5: Setting up a Database Server - The aim of this lab is to start an Amazon RDS Database and enable connections to and from the web server.

Lab 6: Scaling and Balancing Architecture - In this lab, you'll create an Amazon Machine Image (AMI) and establish a load balancer that works with a scaling template, allowing for automated instance launches.

Background Information:

Lab 4: Exploring Amazon EBS - Amazon Elastic Block Store (Amazon EBS) is an AWS storage service commonly used for EC2 instances. EBS volumes, connected through the network, remain separate from the instance's life cycle. These volumes are consistently available and can be used as a boot partition. An advantage of the EBS/EC2 setup is that during an EC2 instance reboot, EBS usage is inactive, resulting in charges based on the instance's state. Additionally, EBS volumes are resilient as they are automatically replicated in the backend. EBS also allows for real-time snapshots stored in Amazon S3, serving as starting points for new volumes. These snapshots can be easily shared, promoting efficient communication among network engineers.

Key advantages of Amazon EBS include:

* Independence from any EC2 instance.
* Usability across various operating systems due to its raw, unformatted nature.
* Low annual failure rates (0.1-1% on average).
* Scalability from 1GB to 16TB, catering to diverse purposes.
* User-friendly GUI suitable for full-stack developers, cloud engineers, or network engineers.

Lab 5: Establishing a Database Server - Amazon Relational Database Service (Amazon RDS) is an AWS service for creating scalable databases that seamlessly integrate with other AWS features. Offering engines such as Amazon Aurora, Oracle, Microsoft SQL Server, PostgreSQL, MySQL, and MariaDB, RDS provides flexibility in choosing the top market database package. The primary aim is to simplify the setup, operation, and scaling of relational databases in the cloud.

Key features of Amazon RDS include:

* Automated backups and snapshots for data recovery.
* Scalability of resources and loads to handle varying workloads.
* Comprehensive security features, including Amazon Key Management Service (KMS) and SSL/TLS.
* Monitoring capabilities using Amazon CloudWatch for debugging and general analysis.

Lab 6: Scaling and Balancing Architecture - Elastic Load Balancing automatically distributes incoming application traffic across multiple Amazon EC2 instances, ensuring fault tolerance. Autoscaling enables scalability to set parameters, providing versatility. Auto scaling adjusts the number of EC2 instances based on demand for effective load management and fault tolerance.

Key features of AWS Elastic Load Balancing include:

* WebSocket support for communication between clients and servers over a single, long-lived connection.
* Health checks on both registered and unregistered targets, automatically removing unhealthy instances.
* Access logs for detailed information on requests sent to the load balancer, aiding in issue diagnosis.

Use cases include:

* Game Hosting: Dynamically distributes traffic across game servers for a smooth gaming experience.
* API Gateway: Ensures high performance and availability by distributing incoming API requests.
* Web Applications: Distributes web traffic across multiple EC2 instances, improving response times for applications like college portals.

Lab Summary:

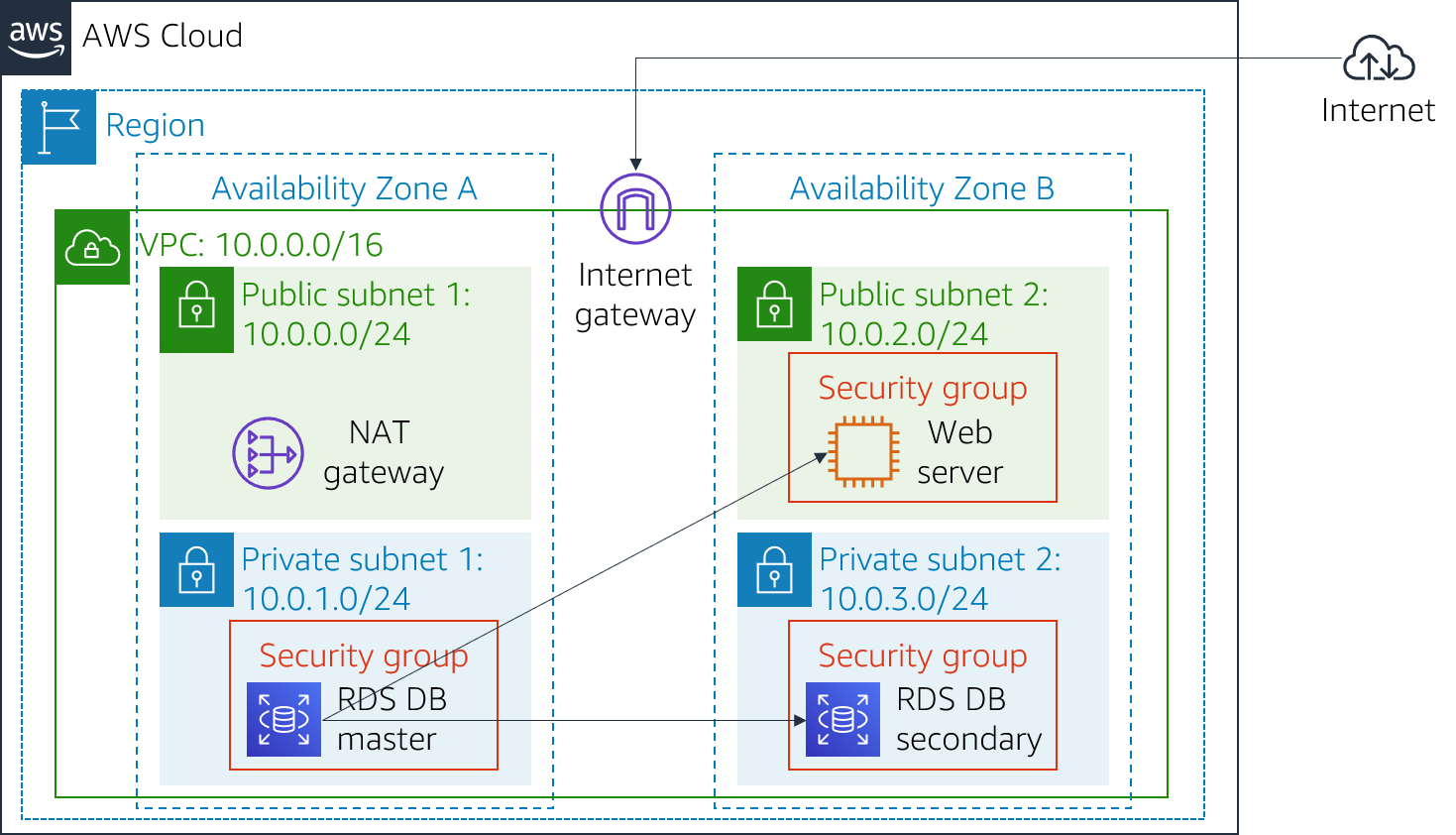
Lab 4: EBS Operations - In this lab, we start by creating an Amazon EBS volume and then attach and mount it onto an EC2 instance. Afterward, we make a snapshot of the volume and reverse the process to generate a new volume from the snapshot. The last step involves attaching and mounting the new volume onto the EC2 instance.

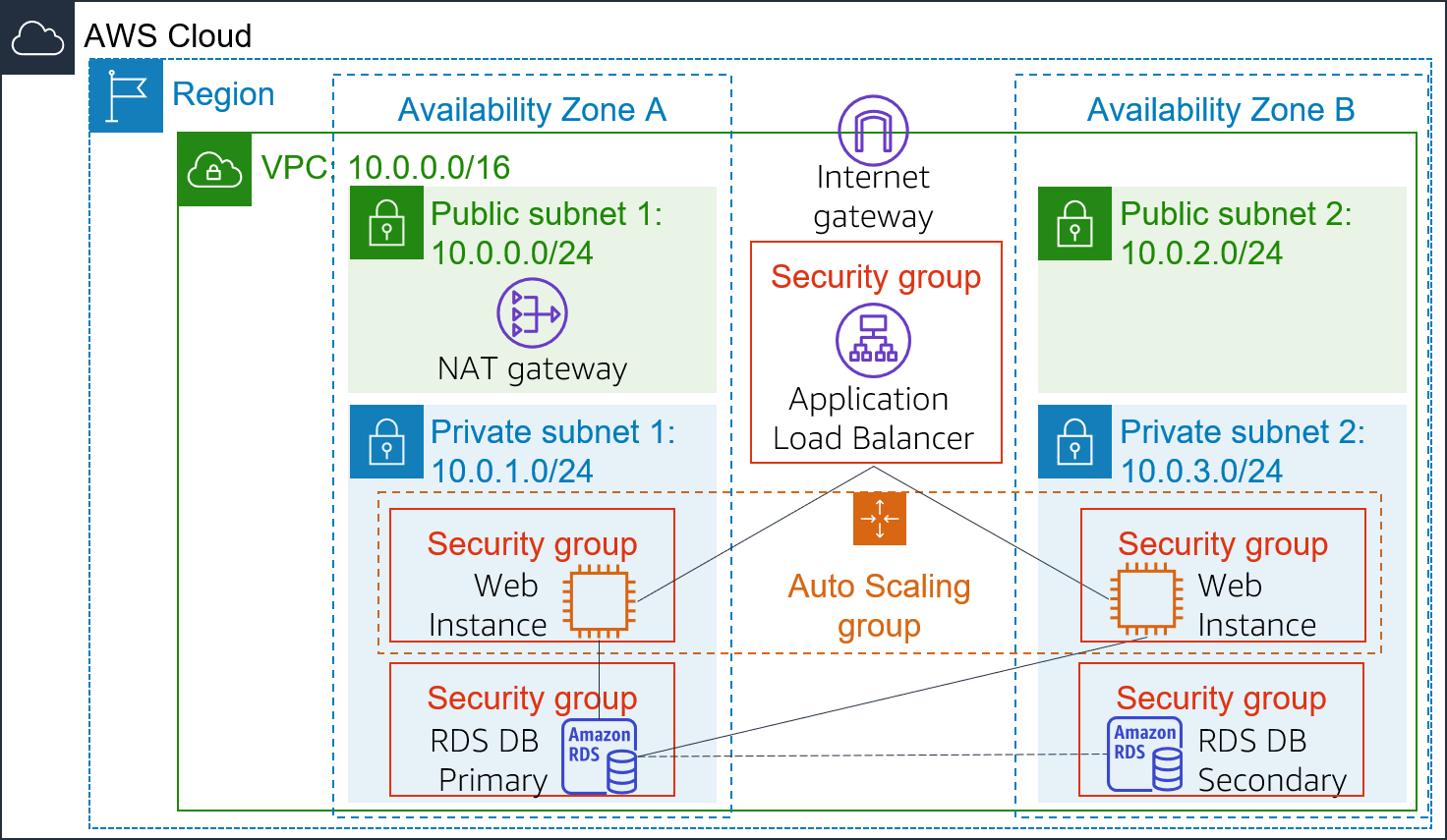
Lab 5: Creating a Database Server - This lab requires launching a pre-configured database using an Amazon RDS DB instance designed for high availability. We then configure the DB instance to allow connections from the web server. To finish, we open a web application to interact with the established database.

Lab 6: Architectural Scaling and Balancing - Starting with the creation of an Amazon Machine Image (AMI) from a running instance, we proceed to set up a load balancer. Using a launch template and an auto-scaling group, we establish a system that automatically scales new instances. The final step involves using Amazon CloudWatch for its alarm system to monitor the performance of the infrastructure created during the lab.

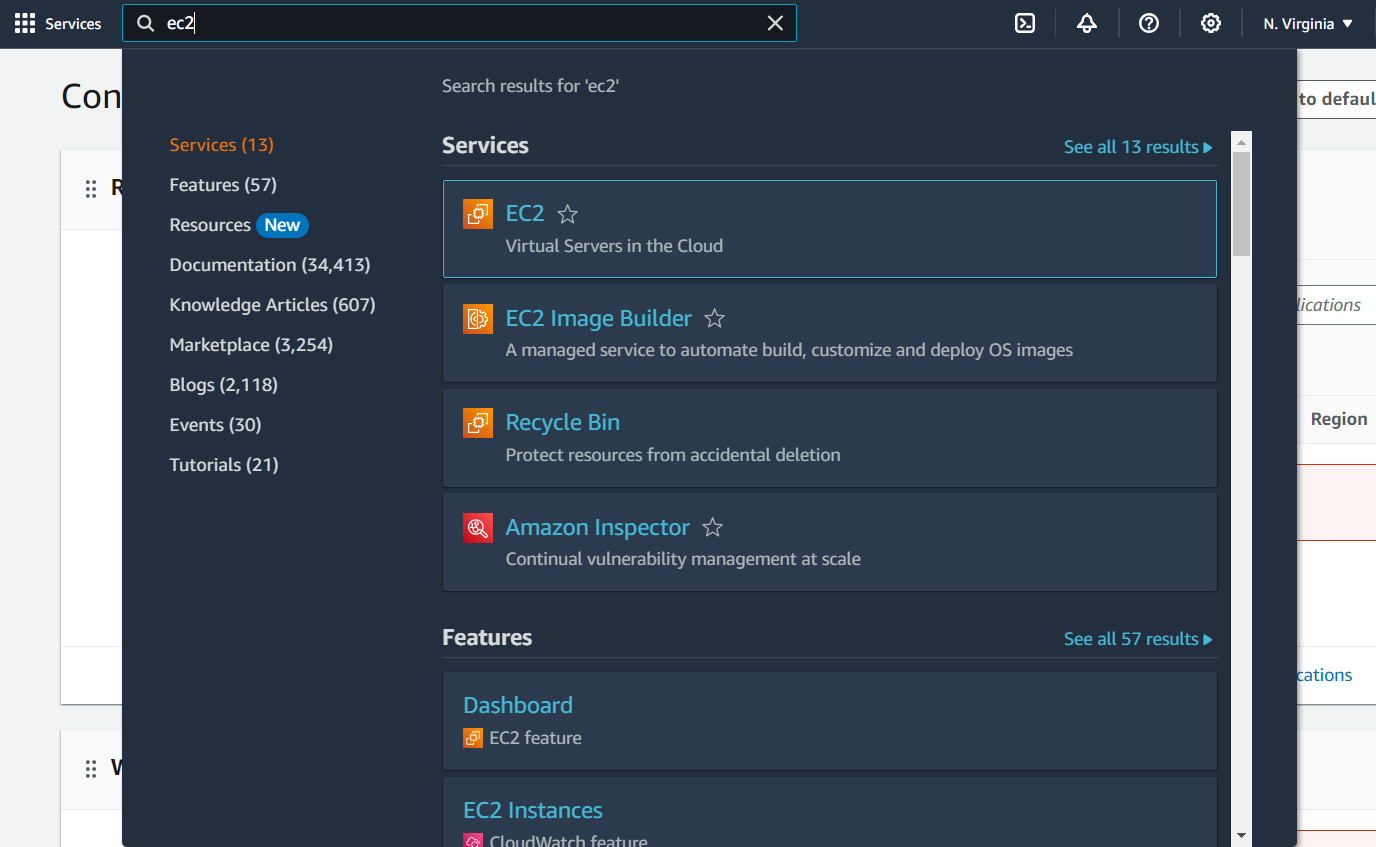
Lab Commands:

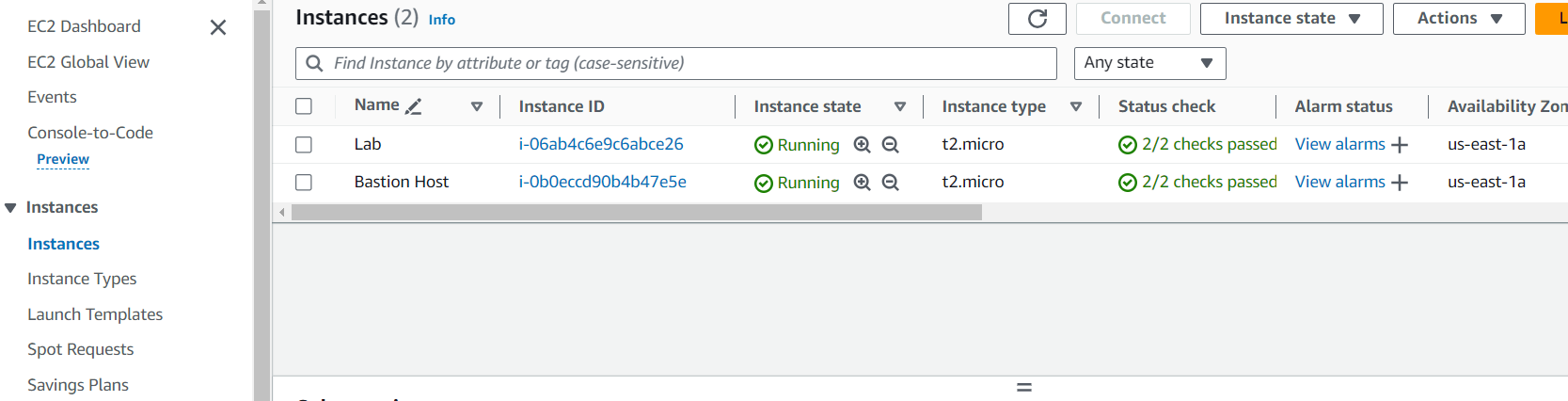
None.

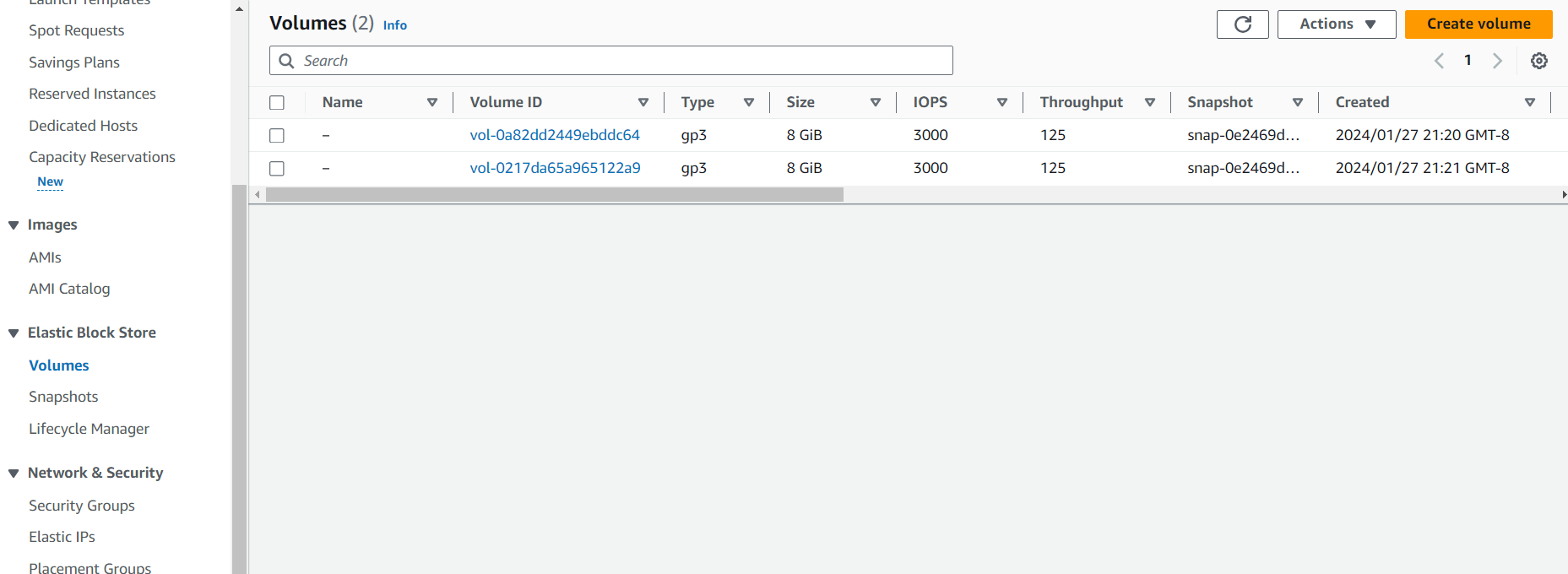
Network Diagrams:****

****

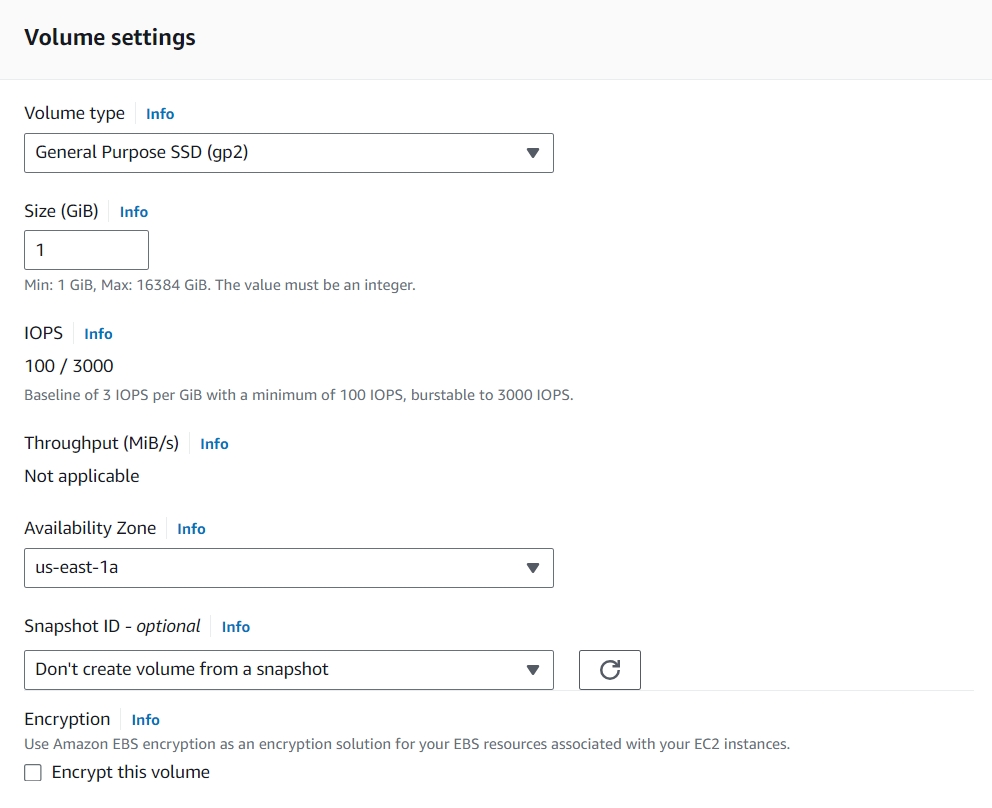
AWS Module 7 Lab 4

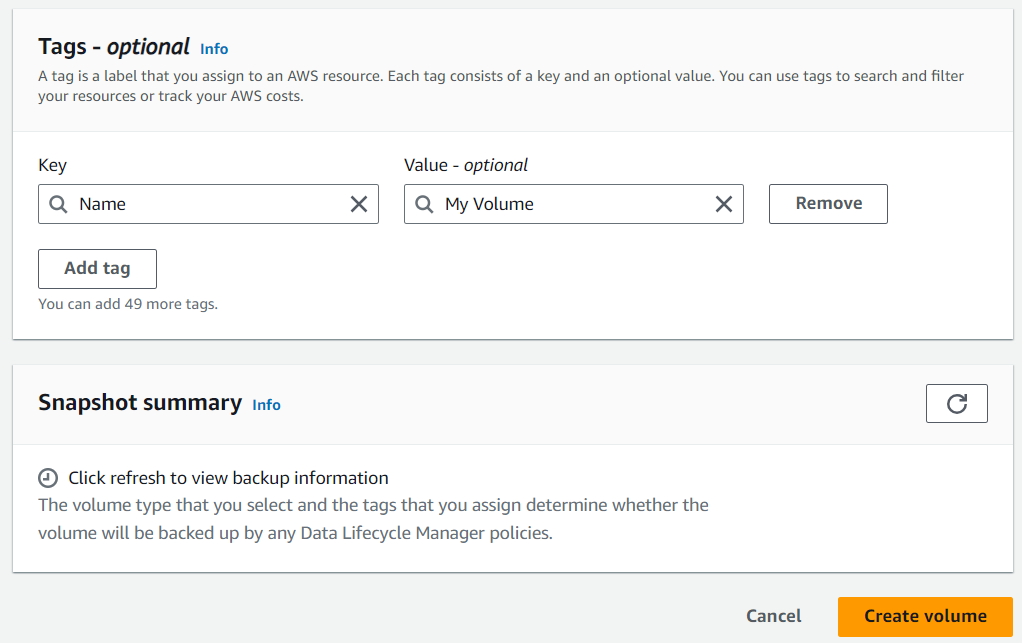
Step 1: Under the *Services* menu, select EC2.

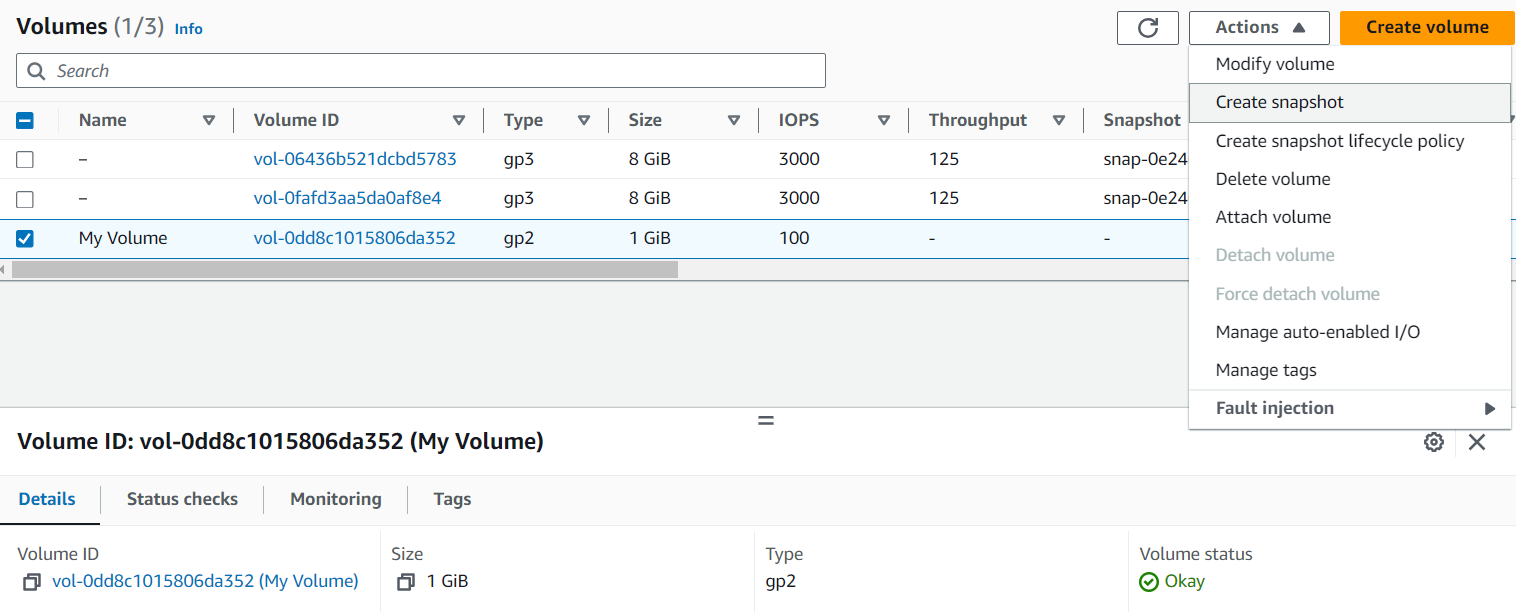
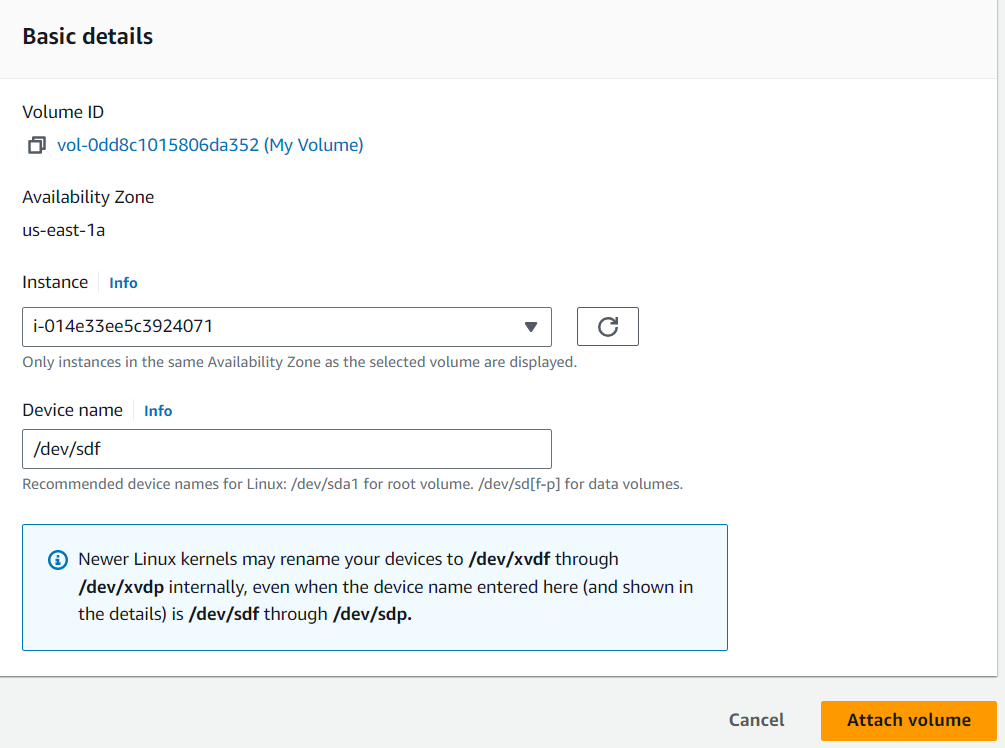
Step 2: Once you enter the EC2 console select instances to verify that the *Lab* instance is still intact and the availability zone is us-east-1a. If not, reset the lab.

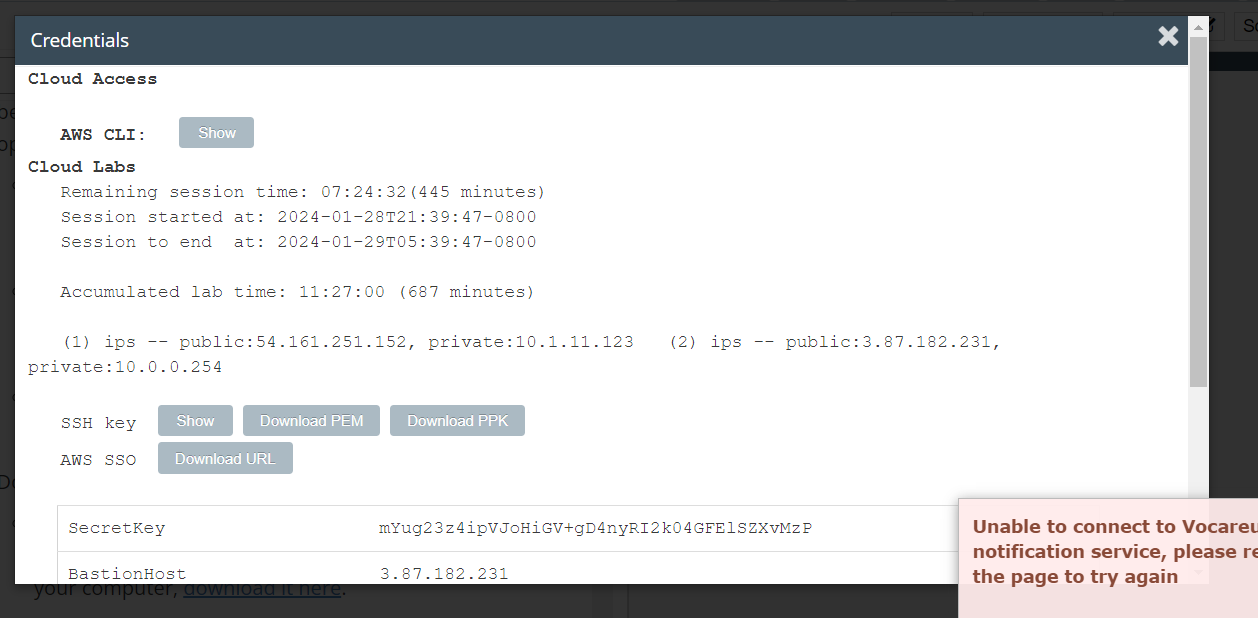
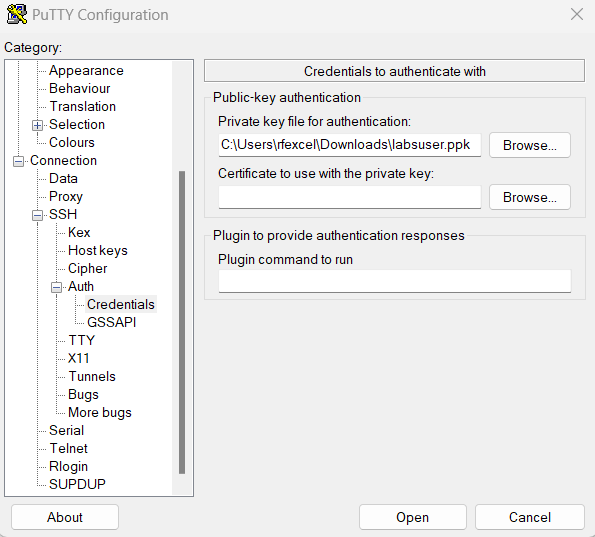
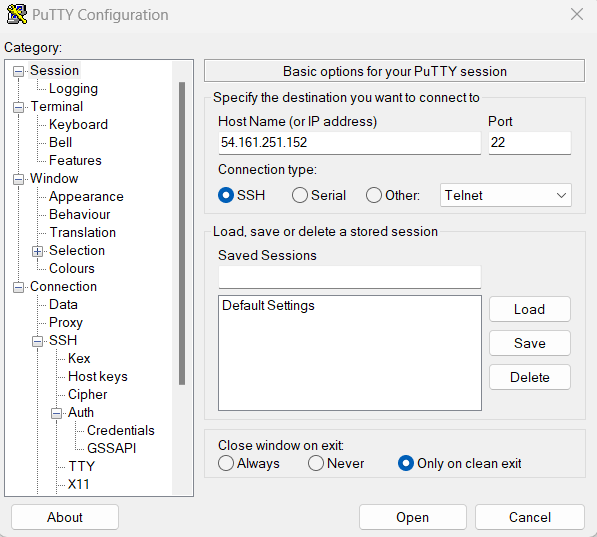
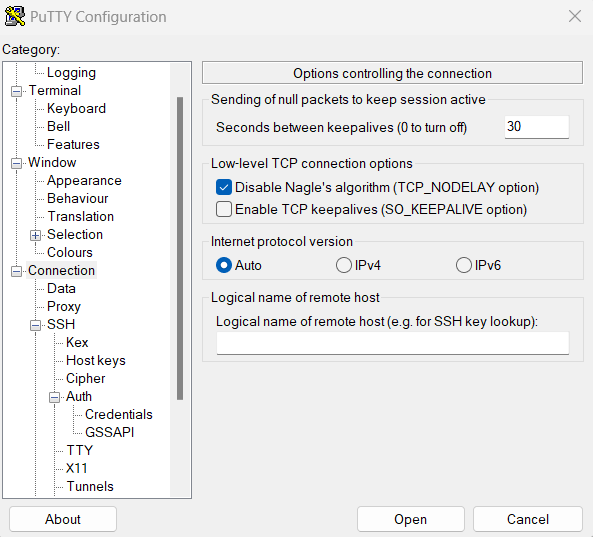
Step 3: After verifying the instance, scroll to *Volumes* under the *Elastic Block Store* section in the navigation panel on the left. Then click *Create Volume*. 

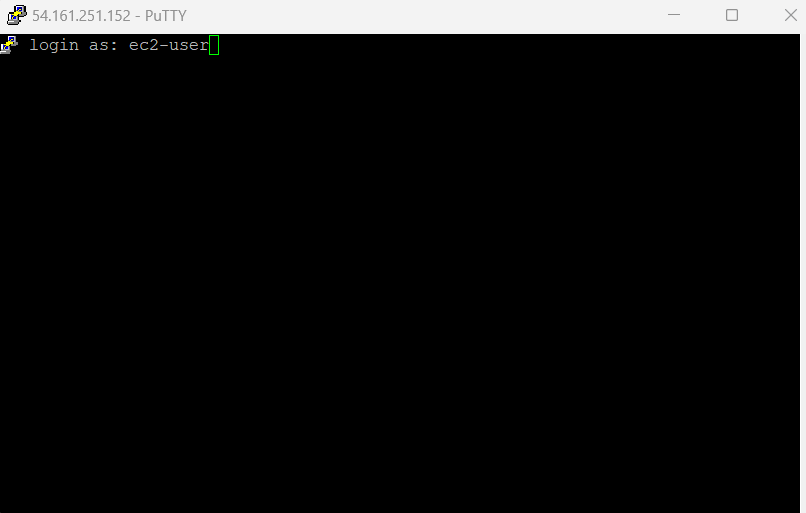
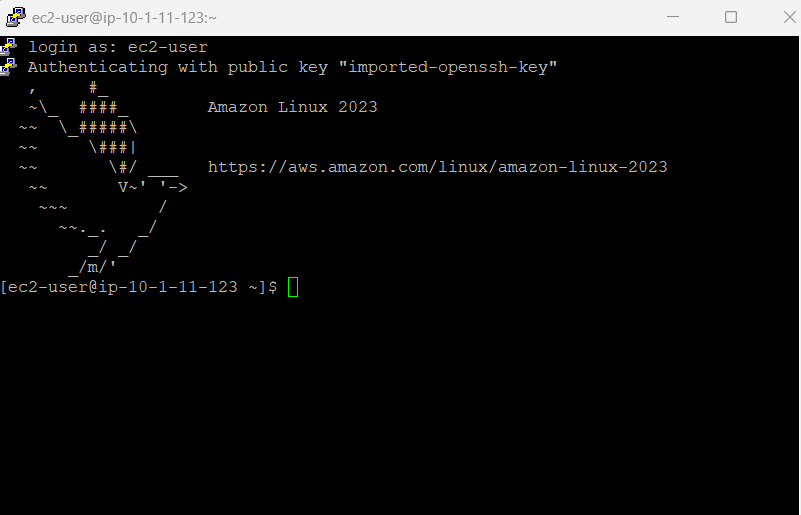
Step 4: Under Volume type, select *General Purpose SSD (gp2).* Change the size field to 1. Change availability zone to us-east-1a. Choose add tag to create a new tag, and under its menu make the key, *Name* and set the value to *My Volume*. Then select *Create volume*.

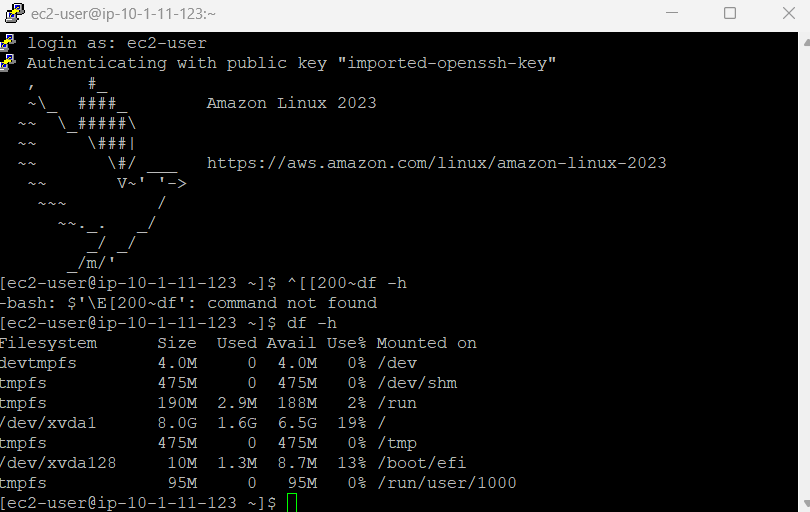


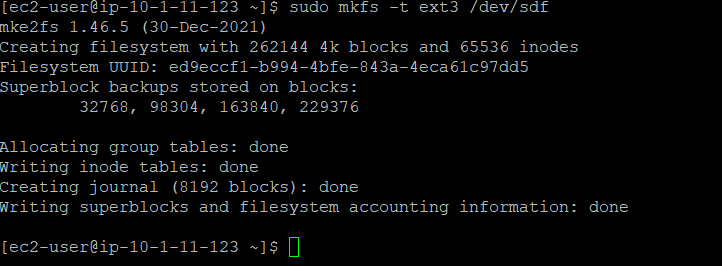
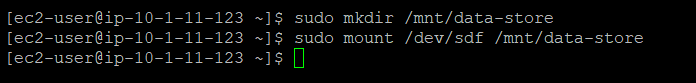
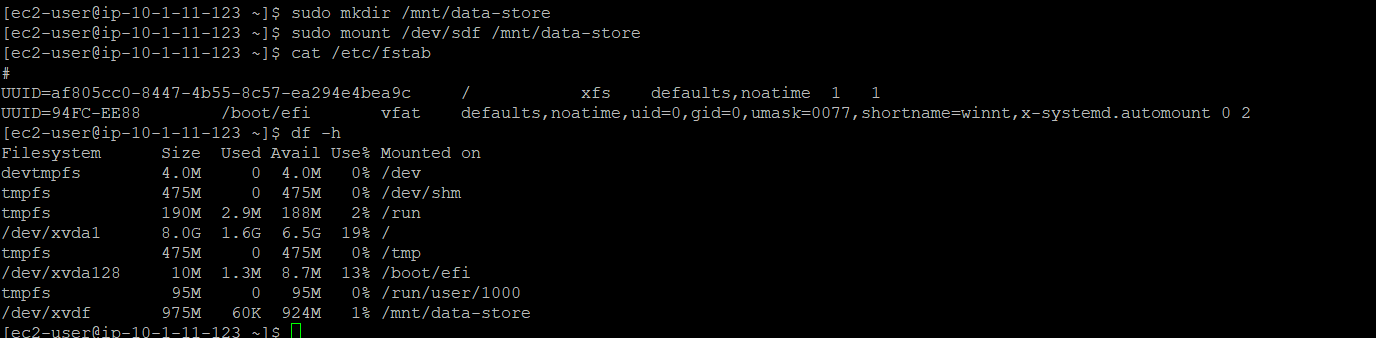
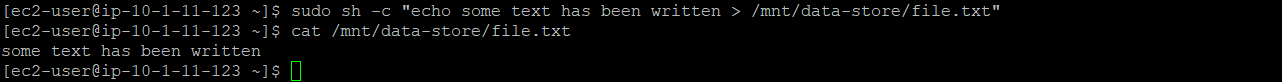


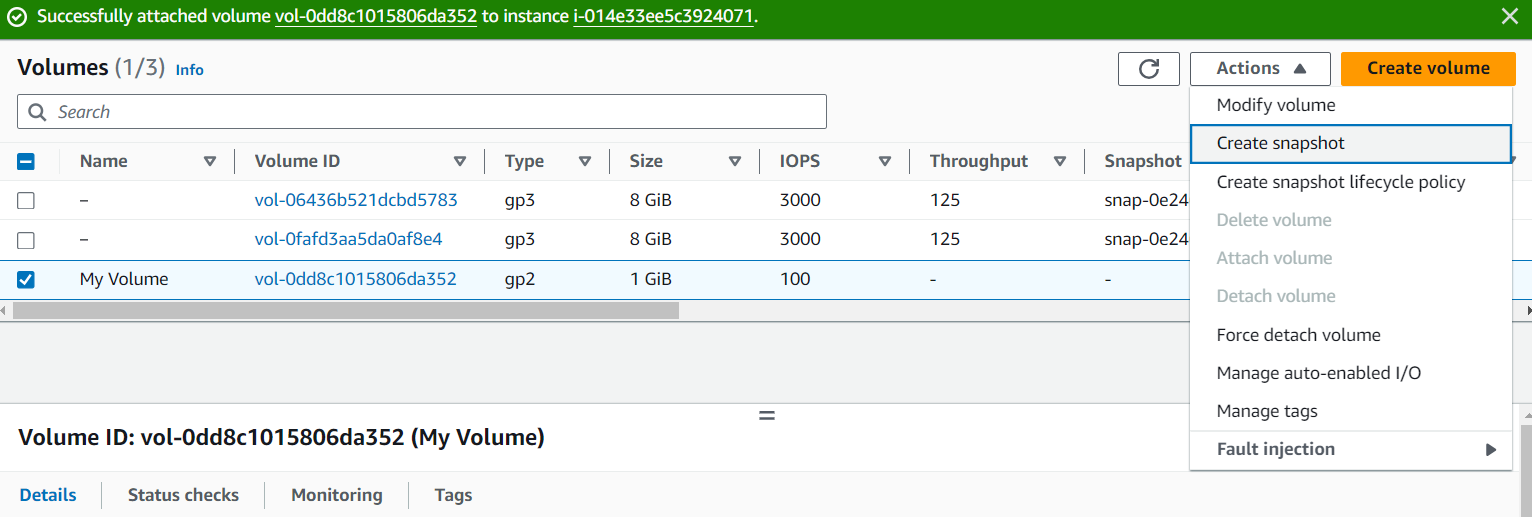
Step 5: Select *My Volume.* Then under the *Actions* menu, select *Attatch Volume.* This will take you to a new section to configure the attatchment of the volume. Then, select the “Lab” instance through the dropdown menu and keep the device name as is. 

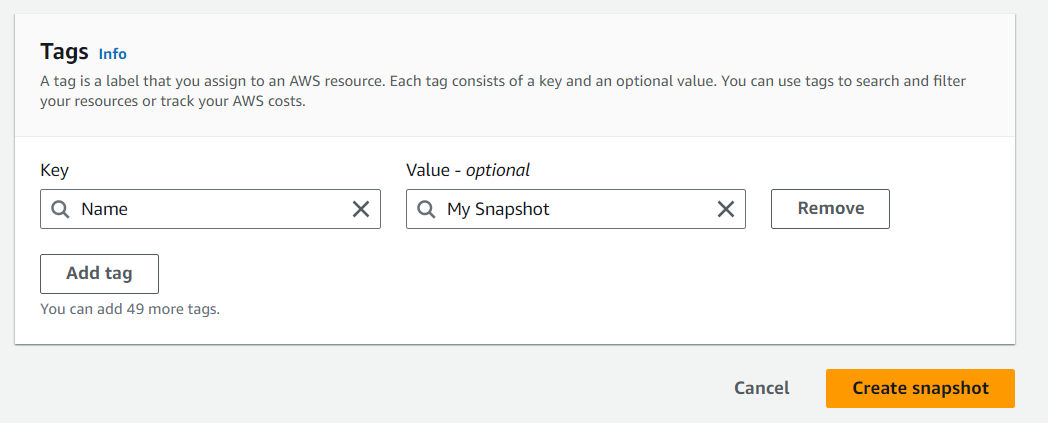
Step 6: Select Details in the AWS Lab Menu. Then select show. Finally, click *Download PPK*. Download and install Putty. Then, open the putty application. Scroll down to connection and change the seconds between keepalives to 30. Then, copy the public IPv4 address for the Lab instance. Paste it under *Hostname* in the *Session* section. Finally scroll down to *Connections,* again, expand it, expand the *Auth* section, and select *Credentials.* Over there, click browse and choose the file you just downloaded. Finally, select open to start the console session. 

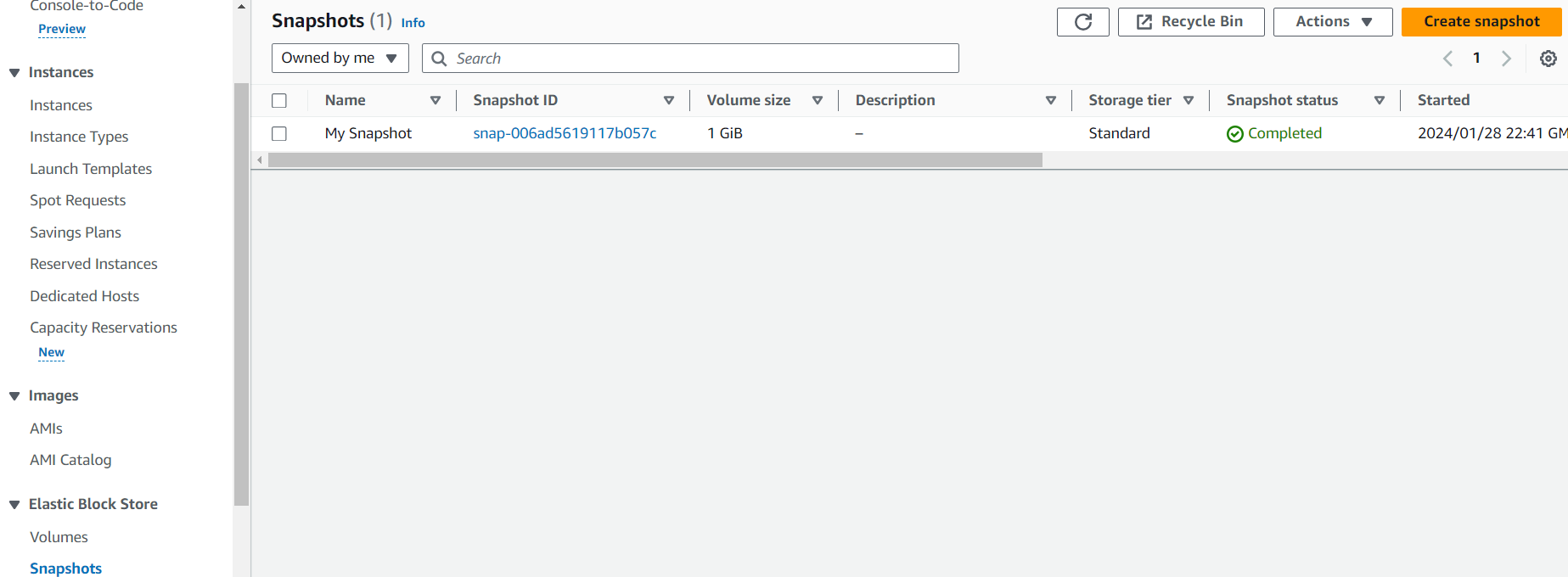
Step 7: Select accept, and log-in as ec2-user. 

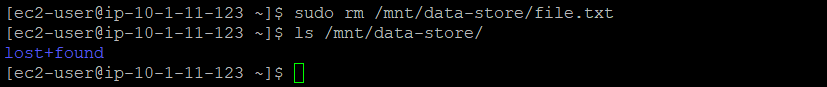
Step 8: Type in the command *df -h*.

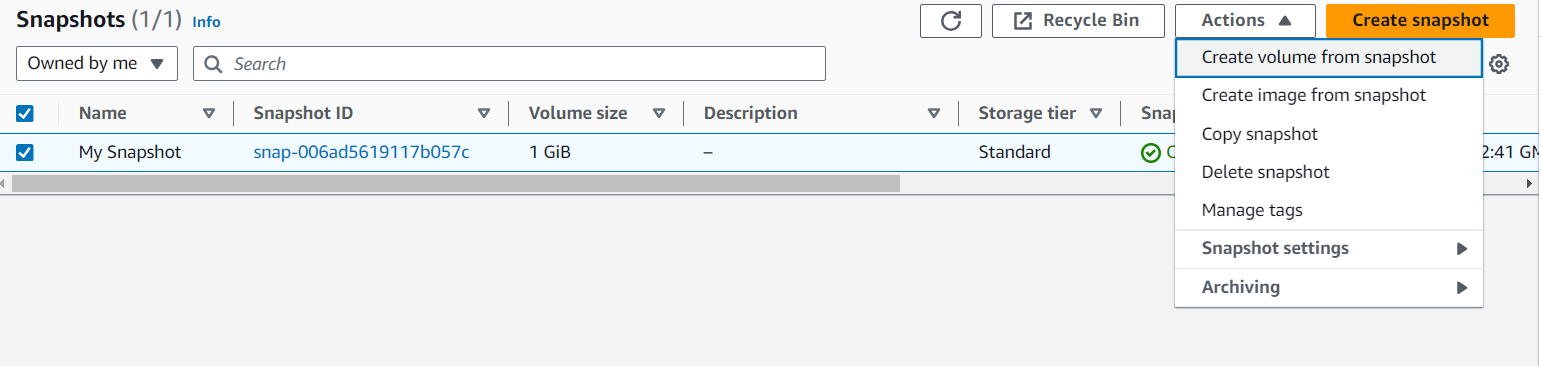
Step 9: Type in the following, highlighted commands.   

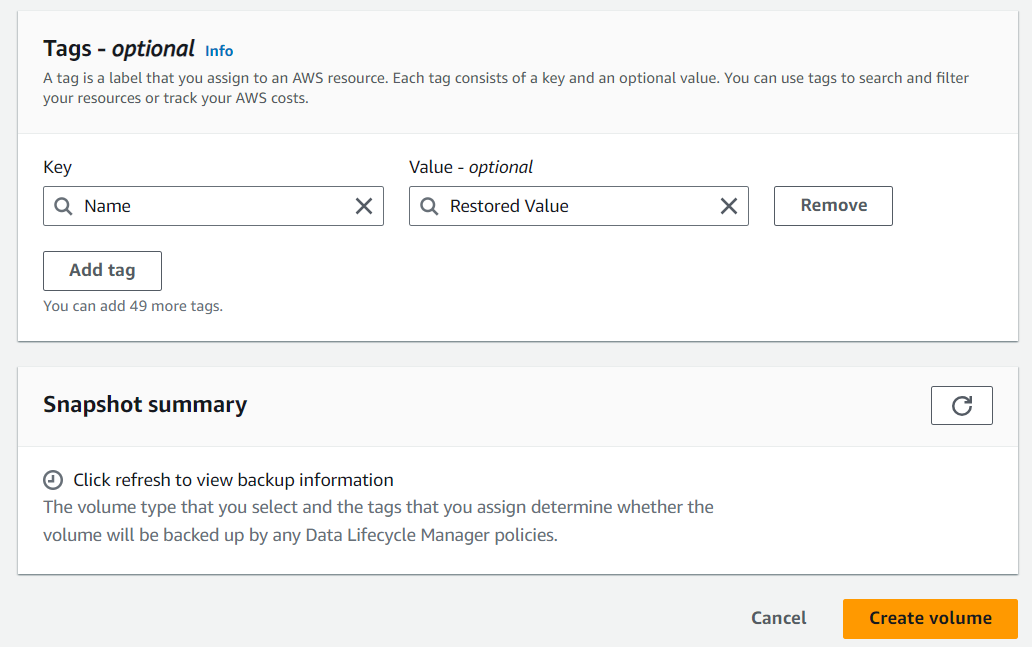
Step 10: Go back to the actions menu, and select *Create Snapshot*.

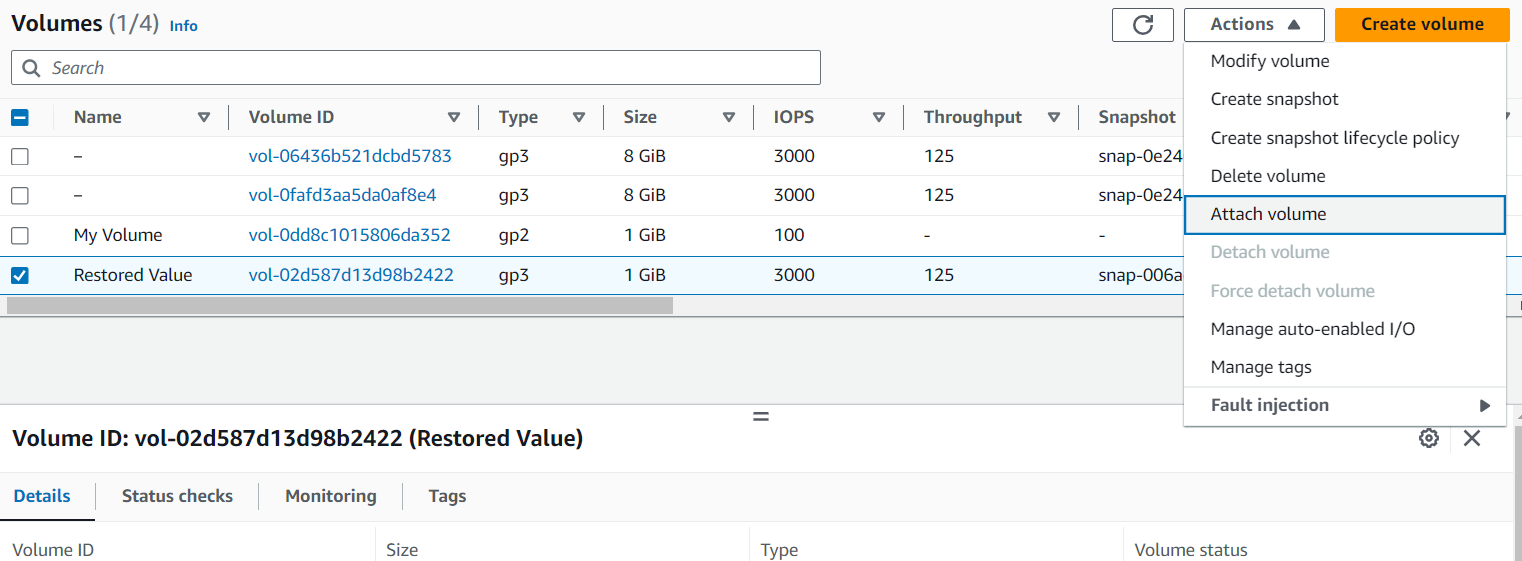
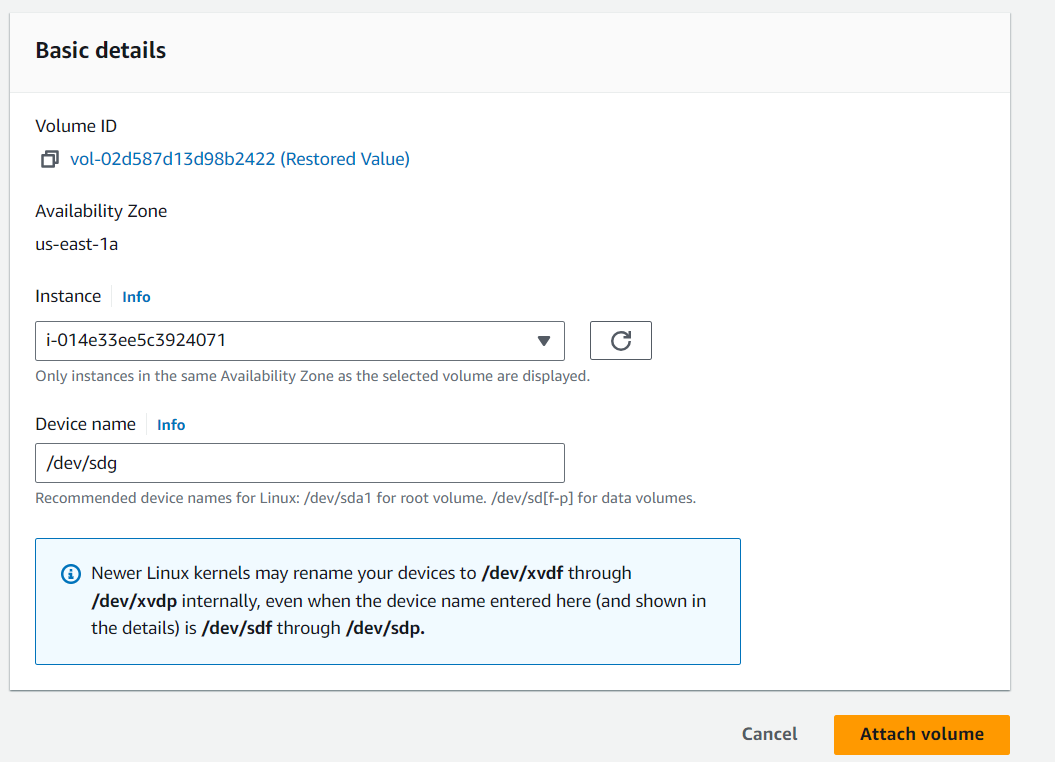
Step 11: Configure a key and value for the snapshot.

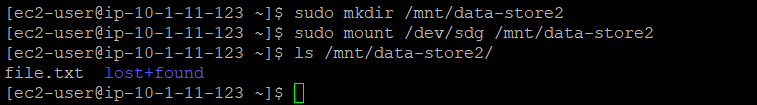
Step 12: A snapshot should be created, and the snapshot status indicates its completetion. This process may take minutes.

Step 13: Add the following commands into Putty to delete the file we creted previously.

Step 14: Select *Create volume from snapshot* under the Actions menu while selectingthe snapshot you created earlier.

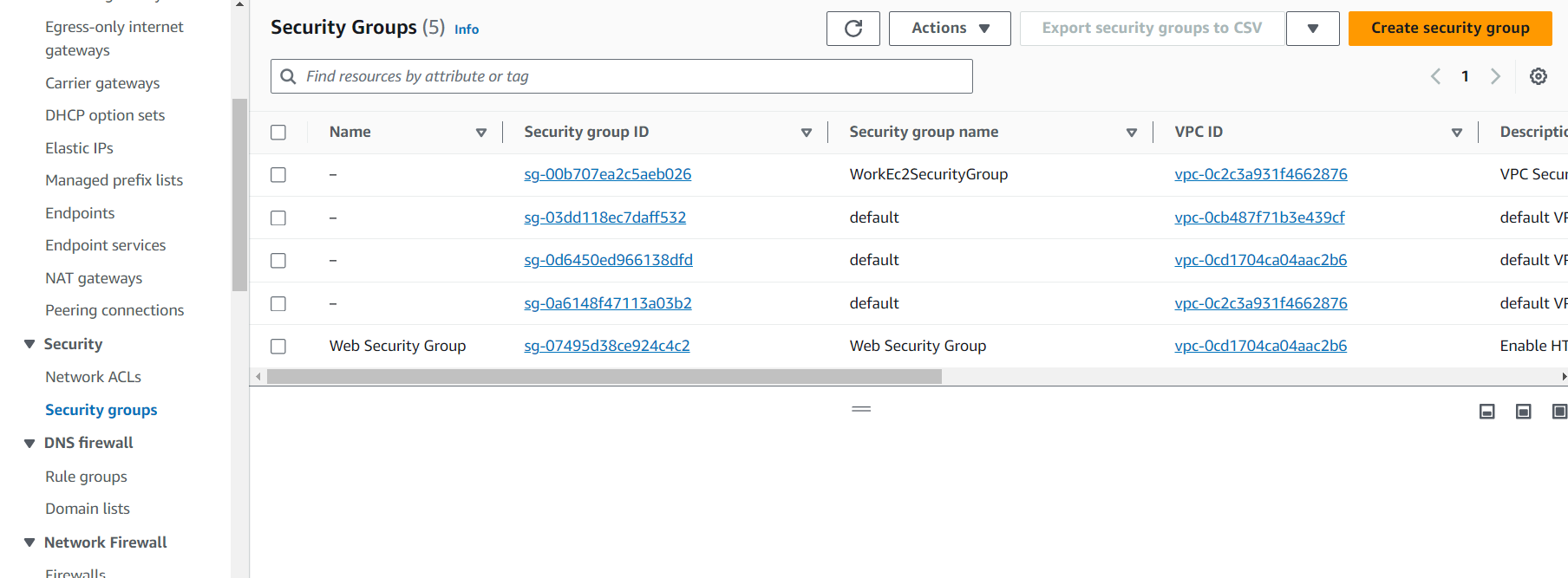
Step 15: Create another tag for the Restored Value, then create the volume.

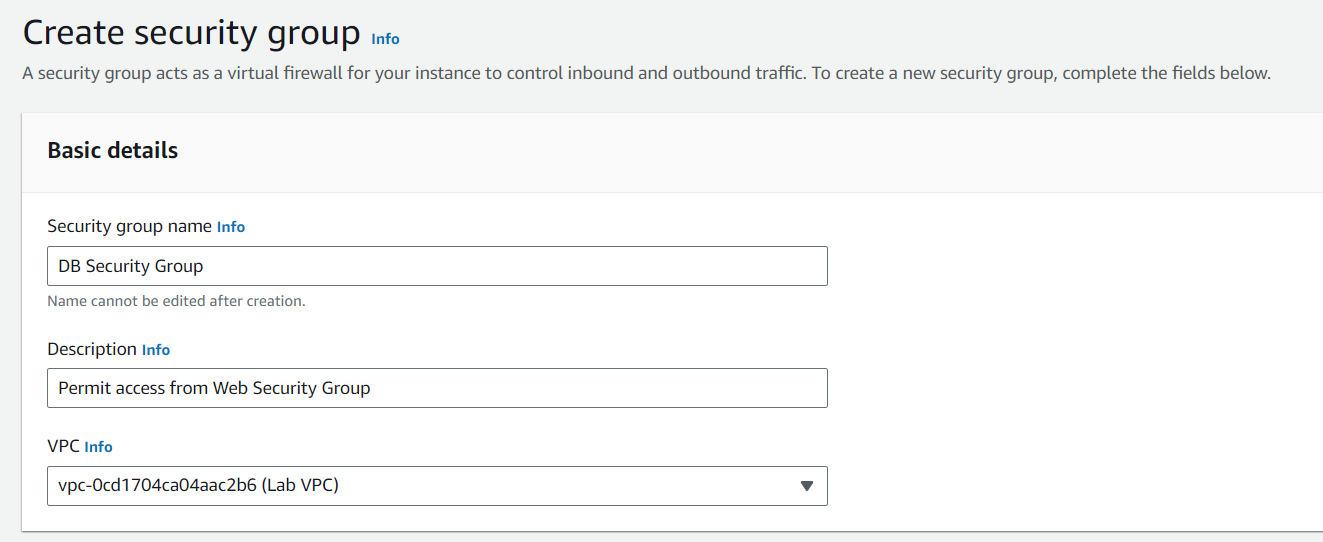
Step 16: Under the volumes section, select the restored value volume and head to the actions section. Then select attatch volume. Then configure its basic details. 

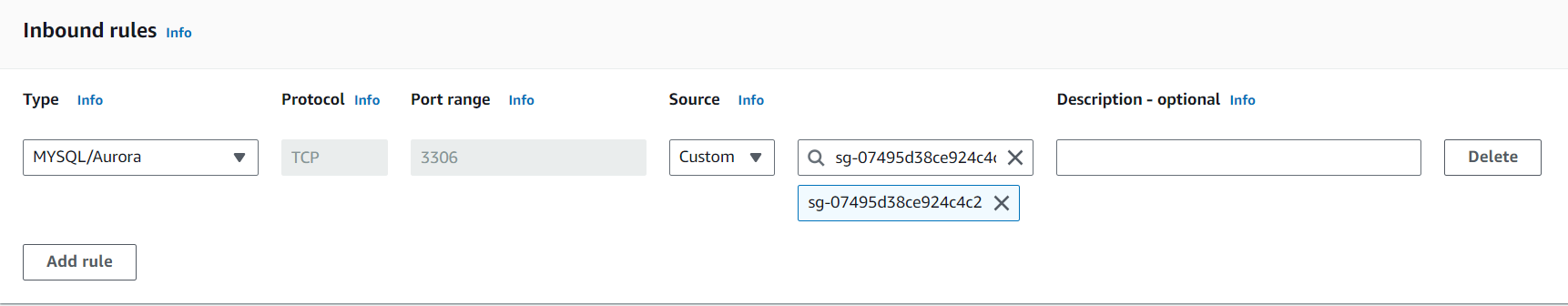
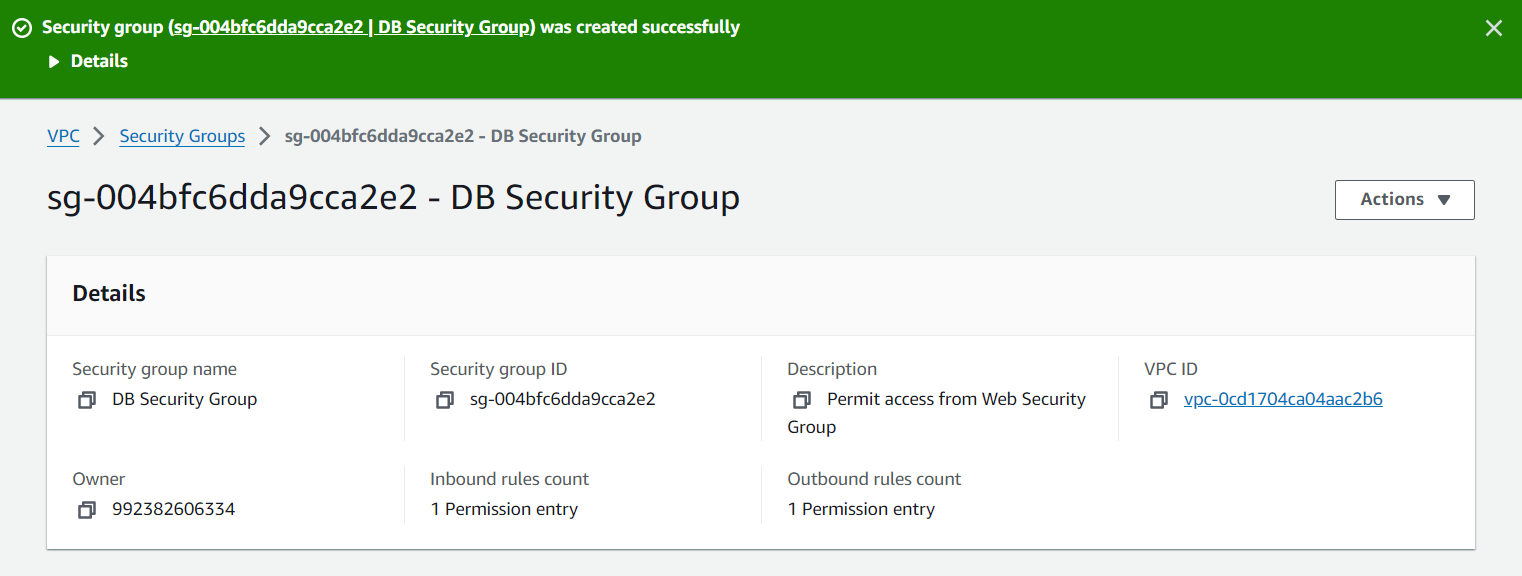
Step 17: Paste the following commands in Putty.

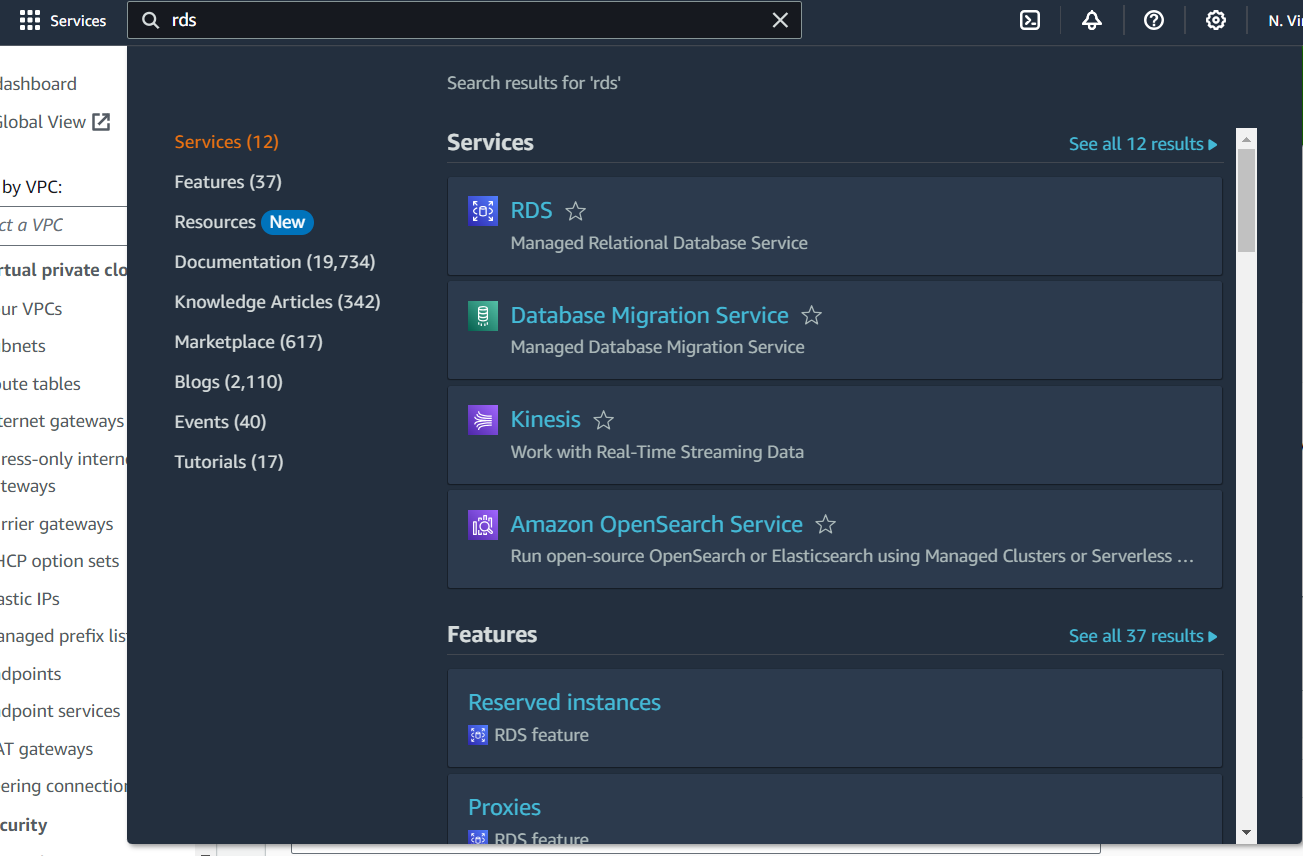
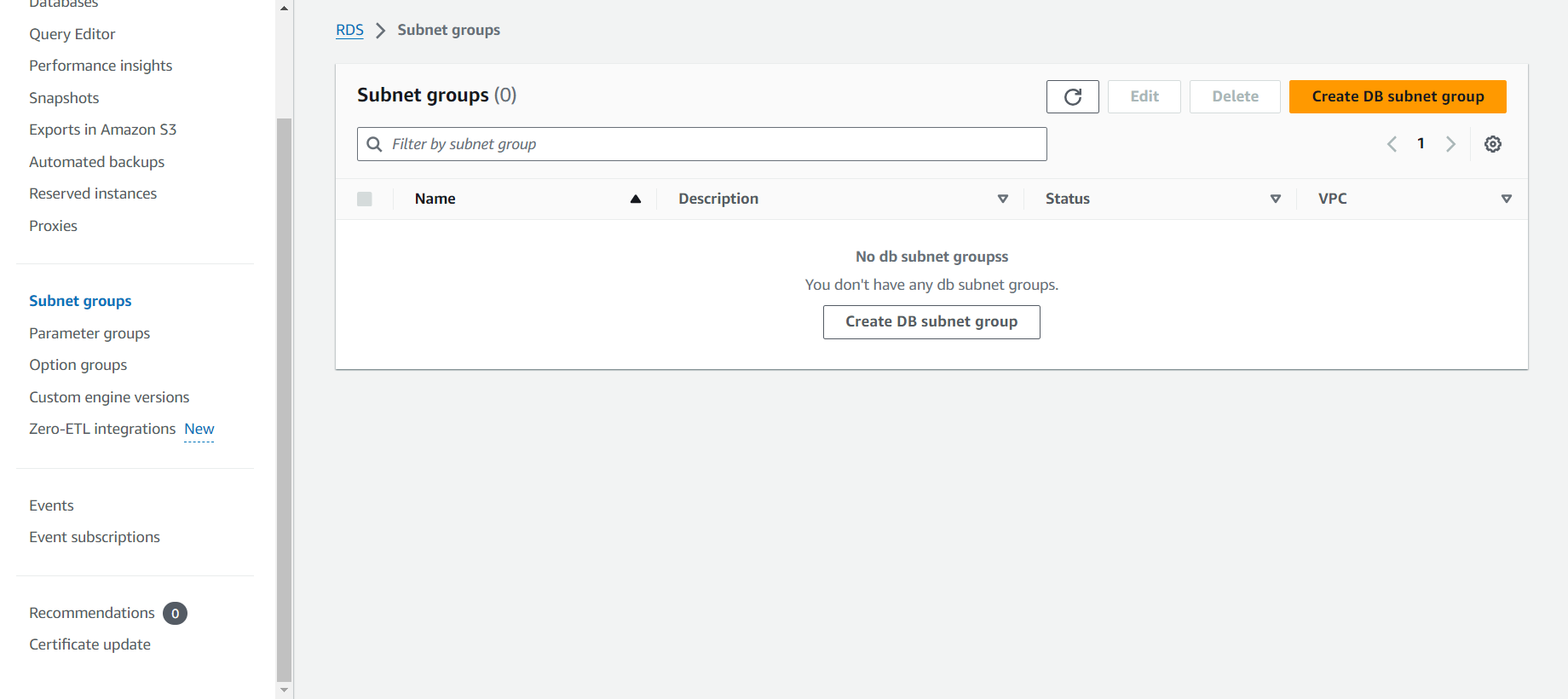
**AWS Module 8 Lab 5**

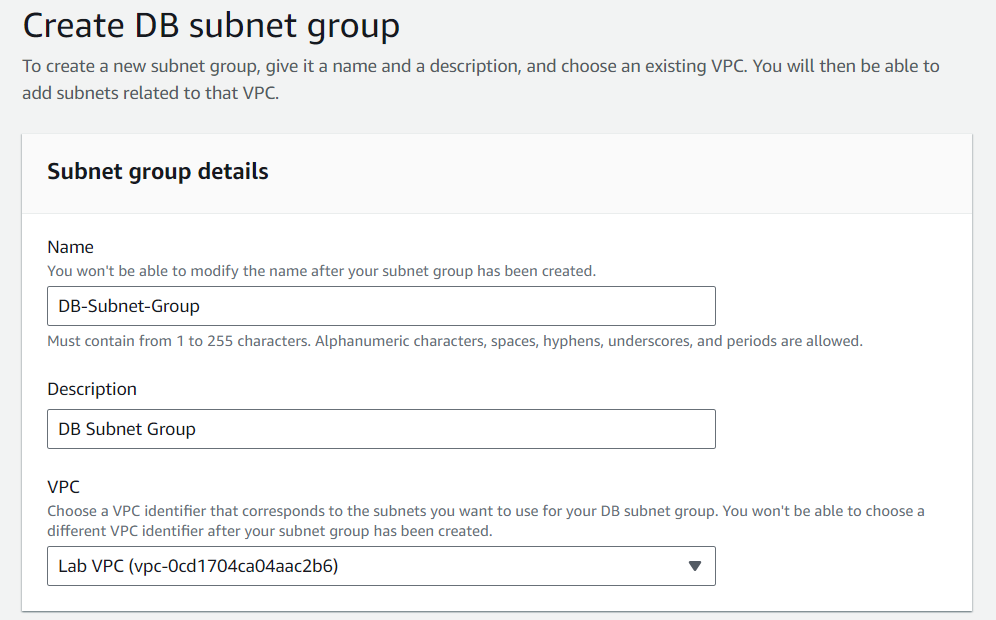
Step 1: Under the Services menu, select *VPC*. 

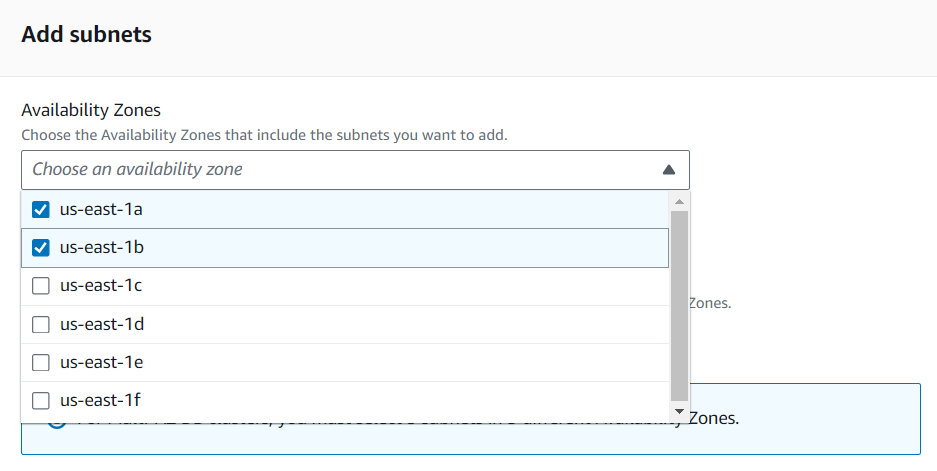
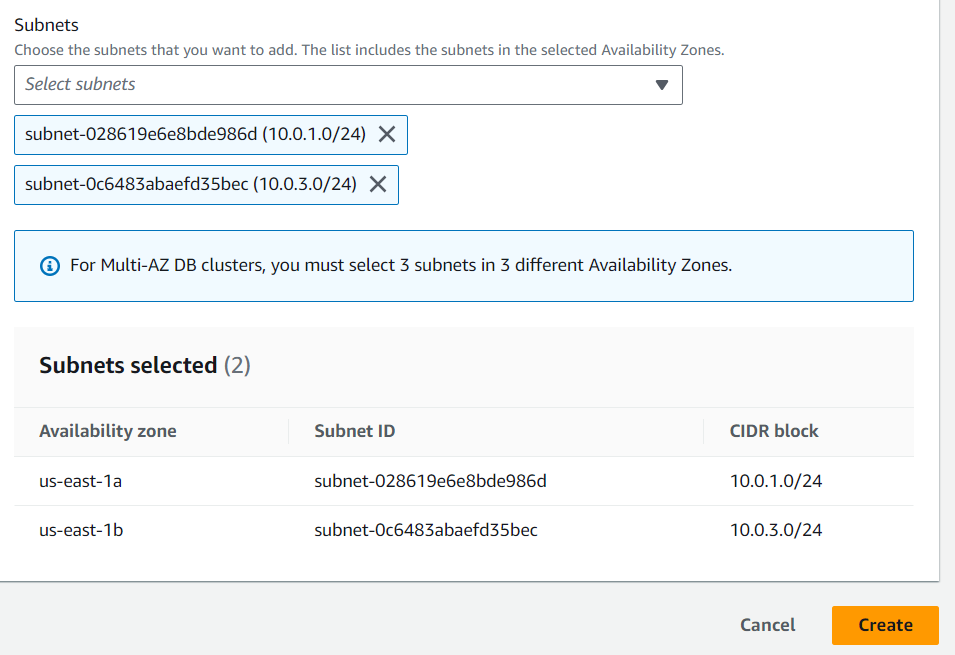
Step 2: Under the navigation panel on the left in the VPC service, select *Security Groups*. Then click *Create security group*.

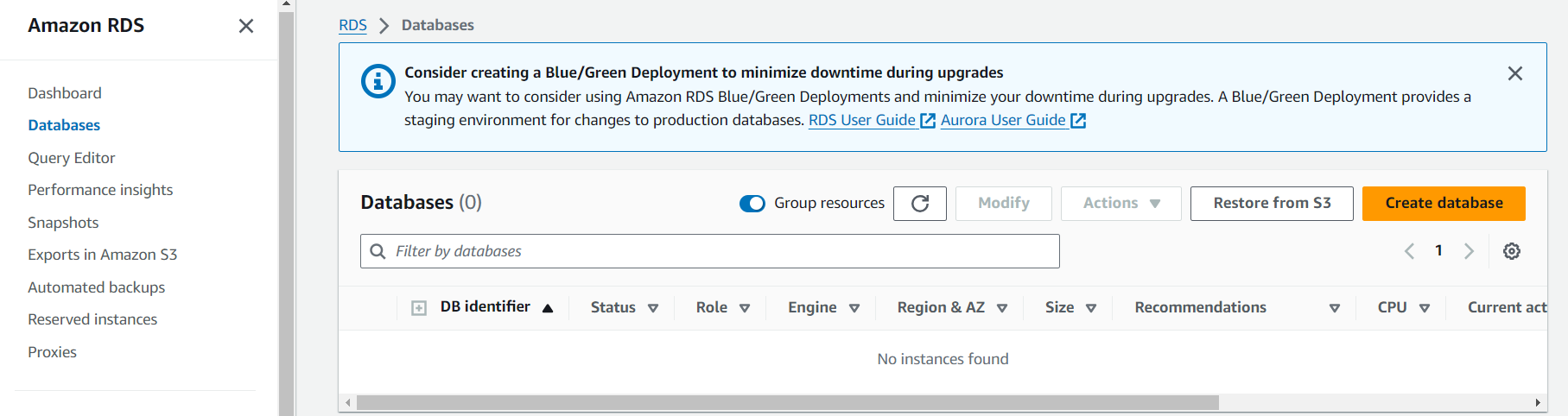
Step 3: Create a security group name, set a description and select the *Lab* VPC.

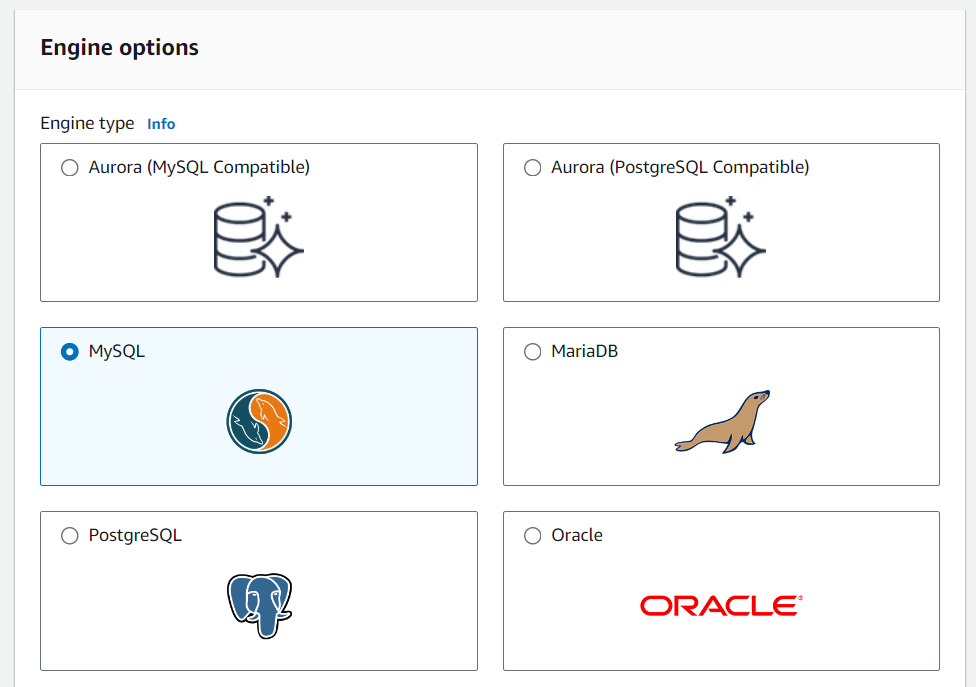
Step 4: Select *add rule* under the inbound rules section. Then select the type to be *MYSQL/Aurora*. Then, under CIDR block, search up VPC. Then choose the one that states *Web Security Group.* Then scroll down and select *Create Security Group*. 

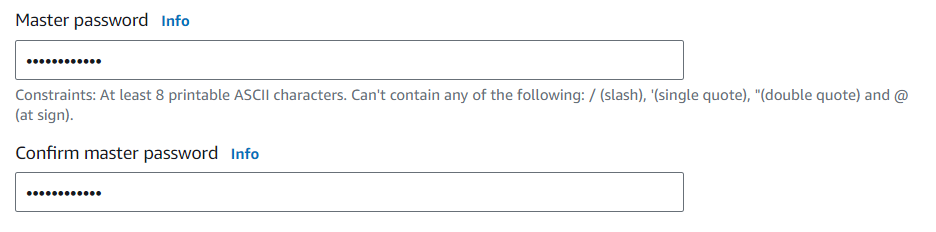
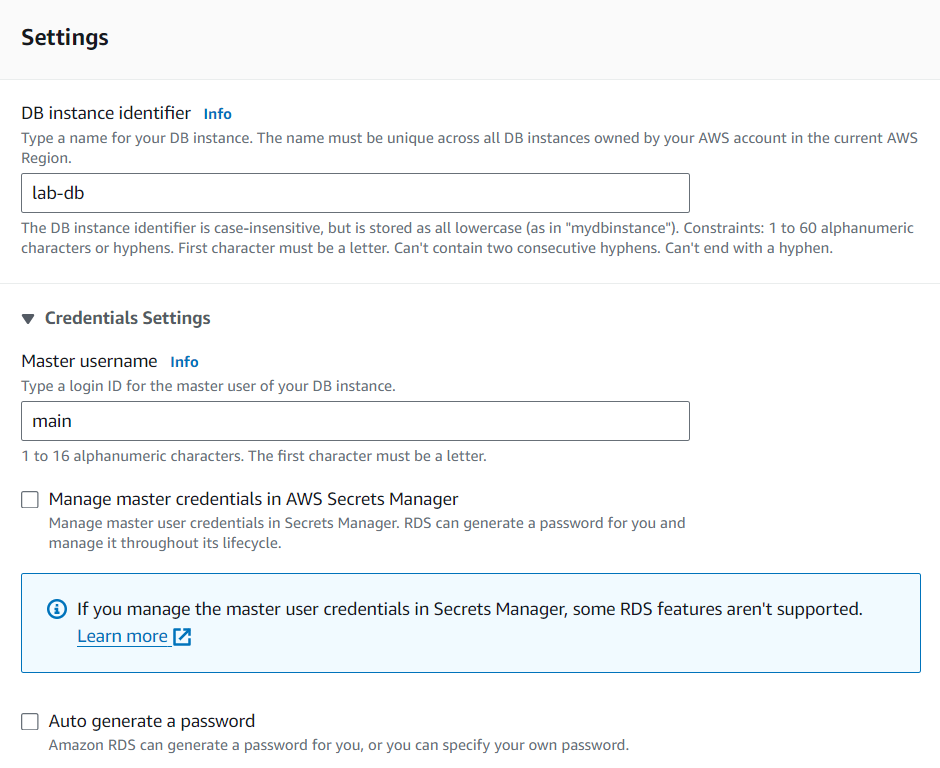
Step 5: Under the services menu, select *RDS*.  
Step 6: Select *Subnet Groups* in the navigation panel on the left. Then select *Create DB subnet* group.

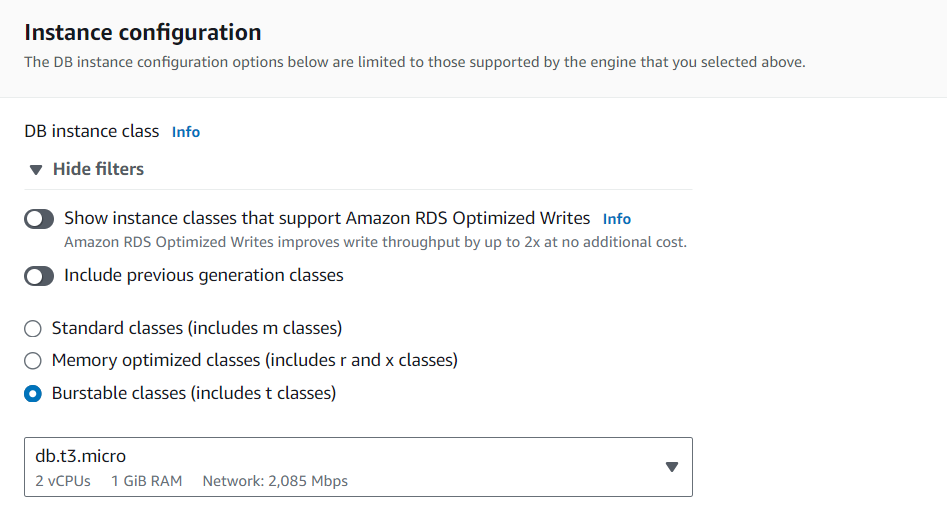
Step 7: Configure a name, description, and select Lab VPC under the dropdown menu.

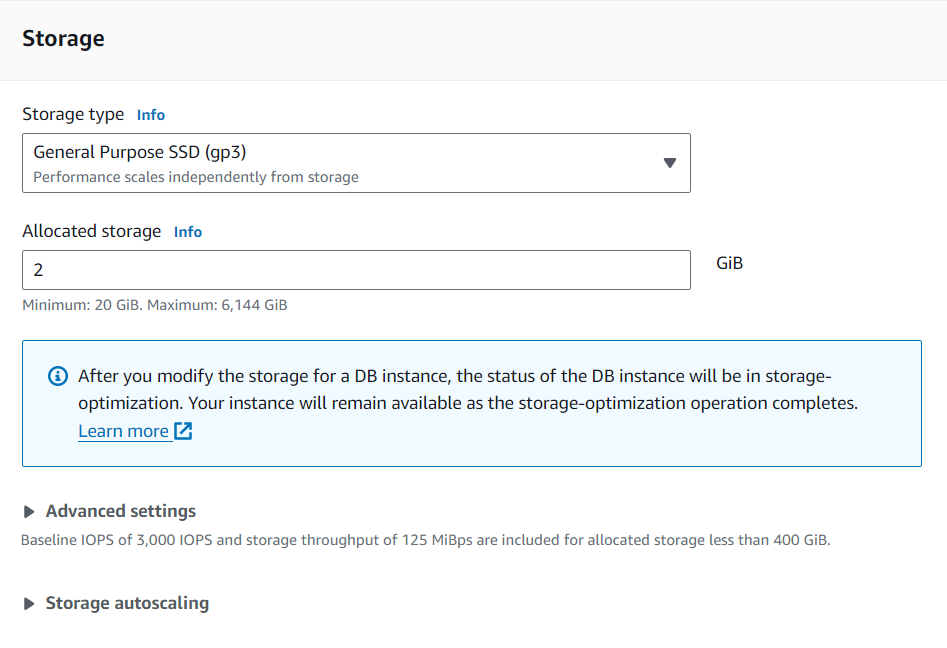
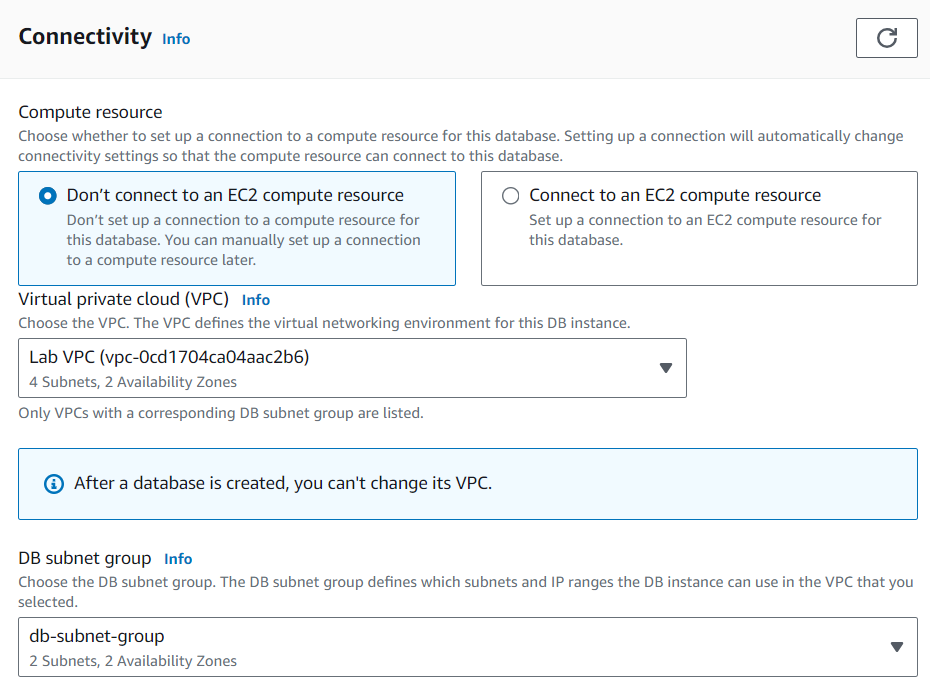
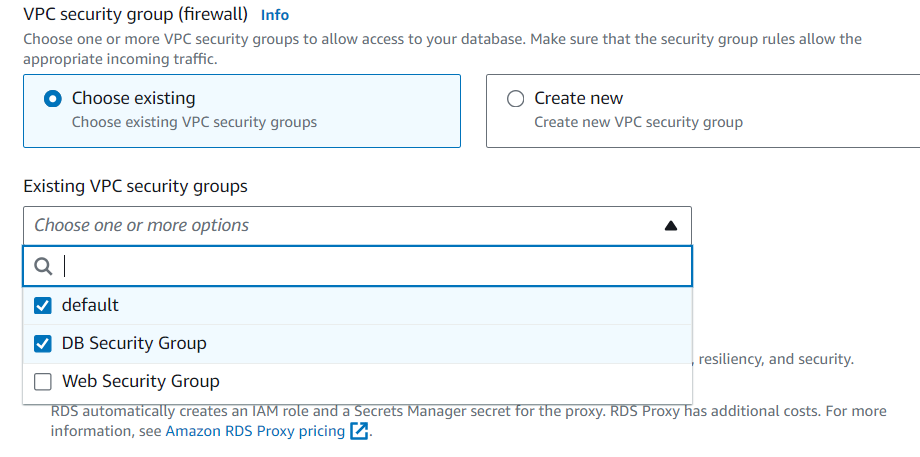
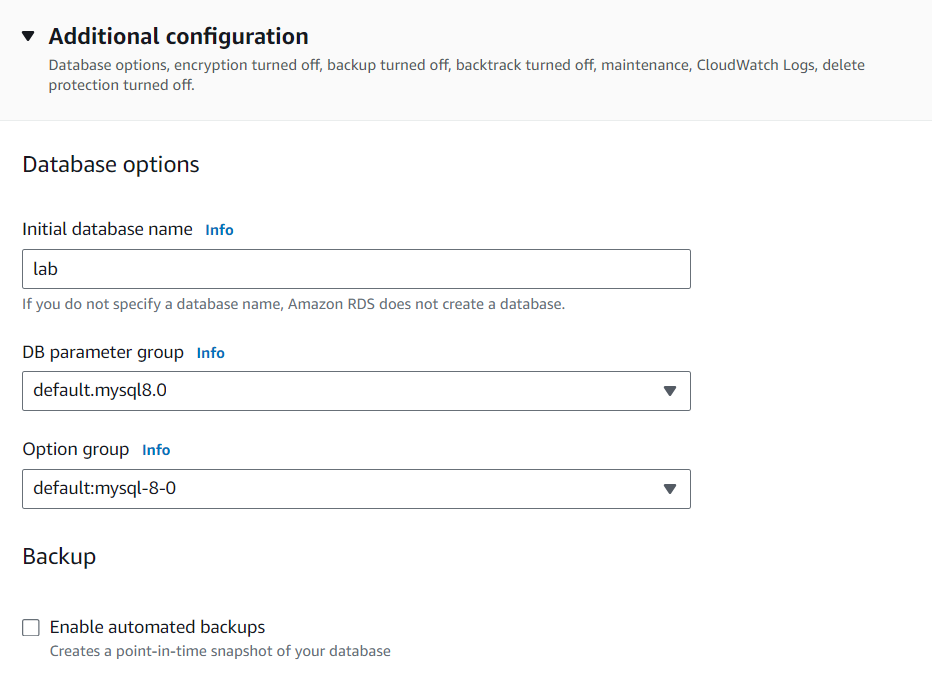
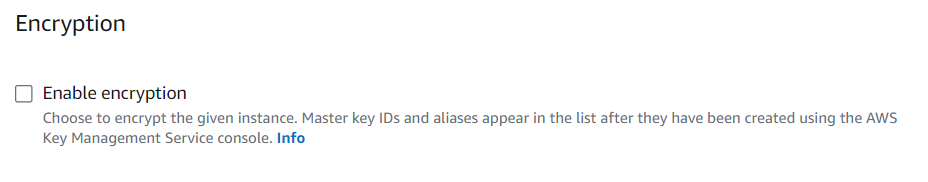
Step 8: Expand the dropdown menu under availability zones and select us-east-1a and us-east-1b. Then under Subnets, select the subnets that are in the IP ranges: *10.0.1.0/24, and 10.0.3.0/24.* Then click *create.* 

Step 9: Scroll up to the *Databases* section and select *Create Database*.

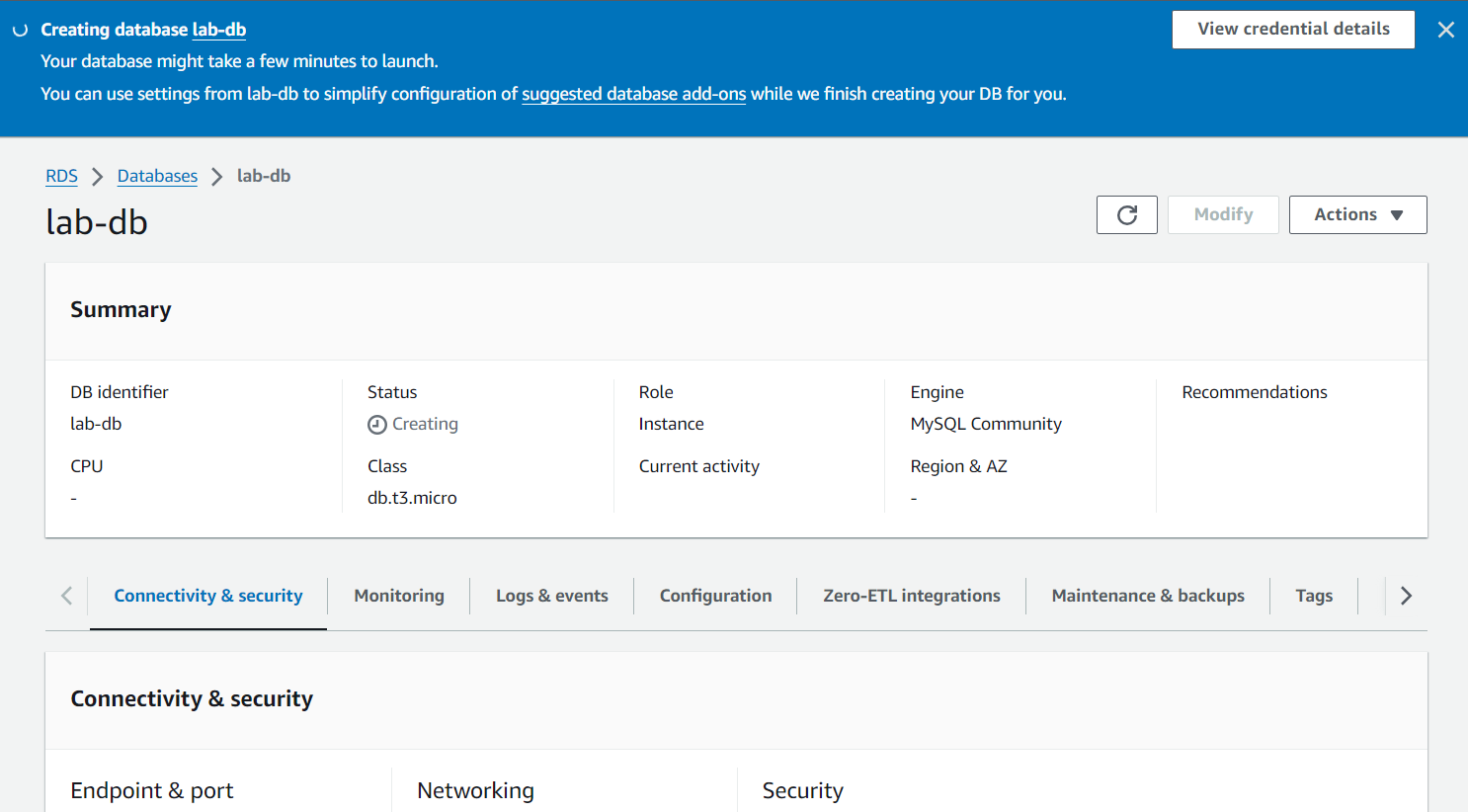
Step 10: Under *Engine options* select, *MySQL.* Then, under *templates*, select *dev/test*. Finally, choose *multi-AZ DB instance* under *availability and durability*. 

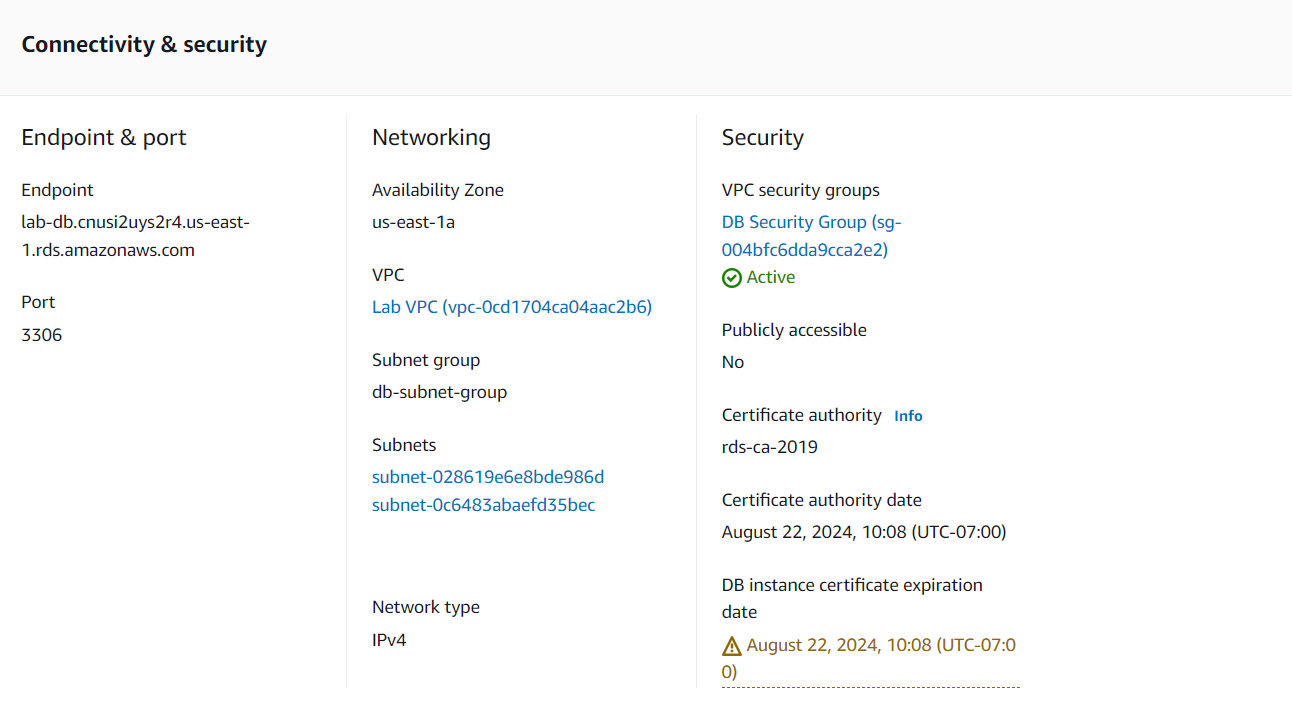
Step 11: Under the settings section, configure the DB instance identifier. Create a master username, password, and then confirm your password by re-typing it in. 

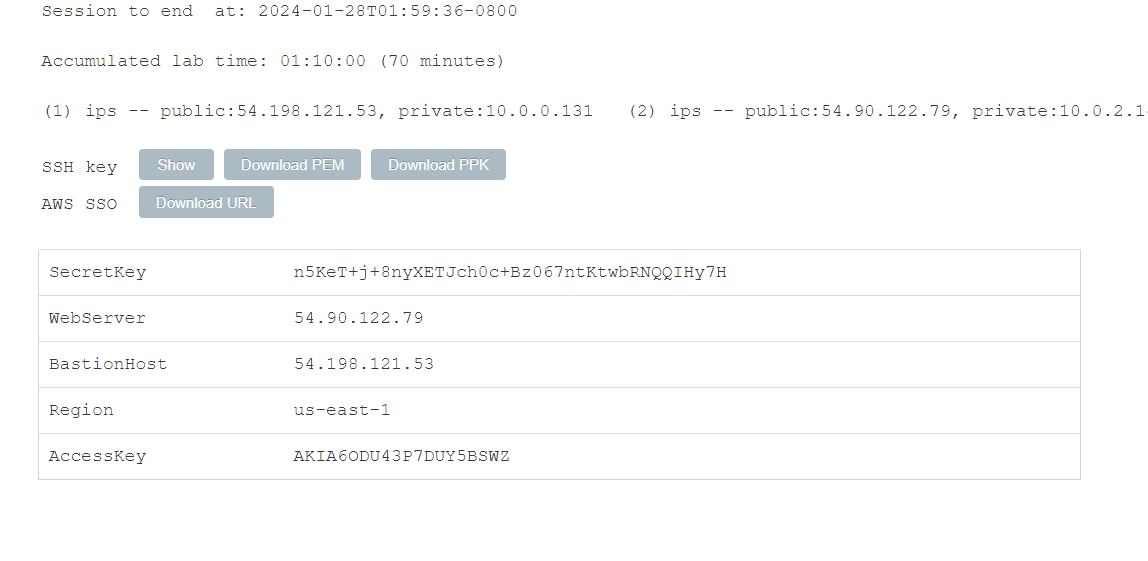
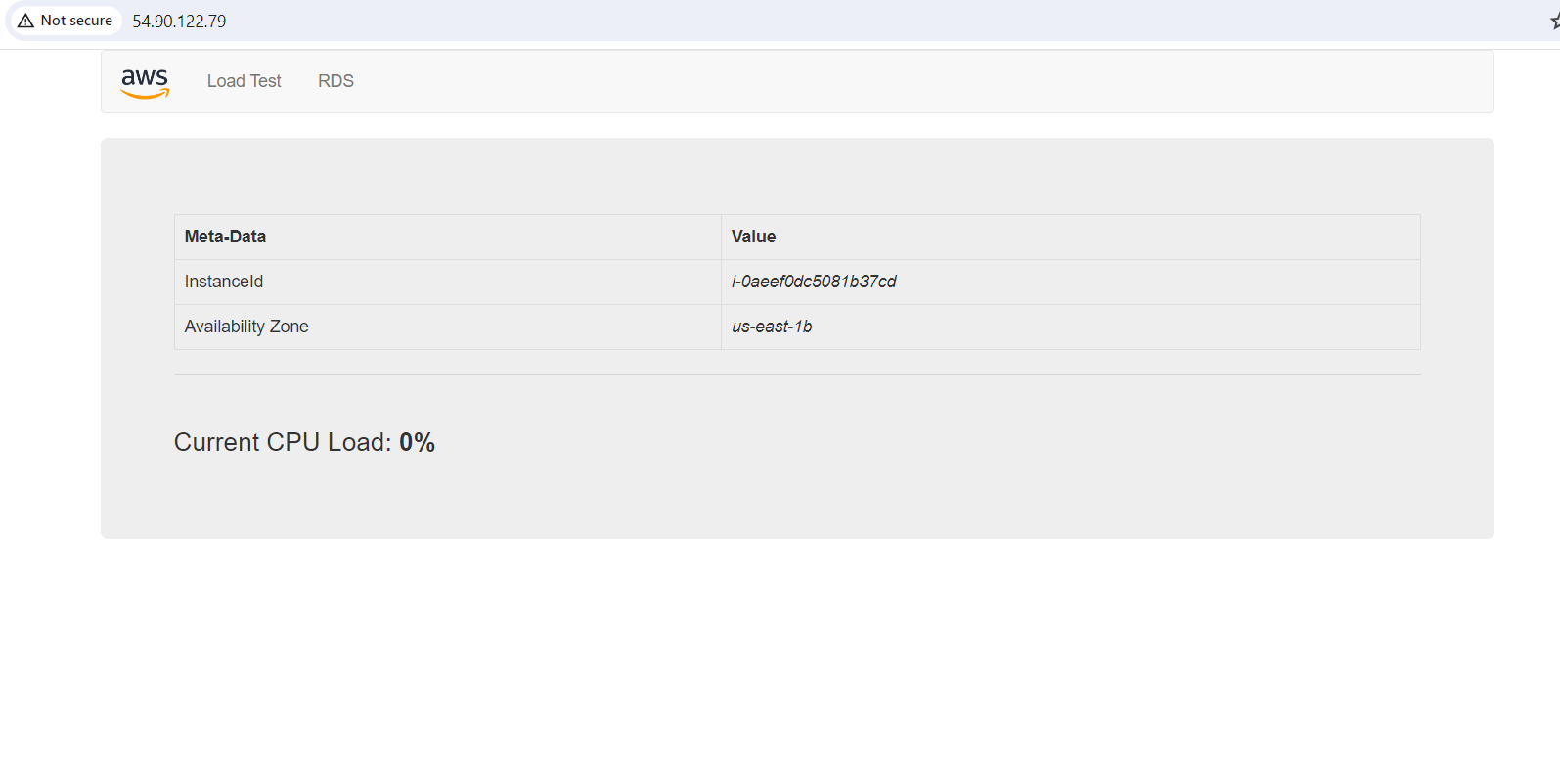
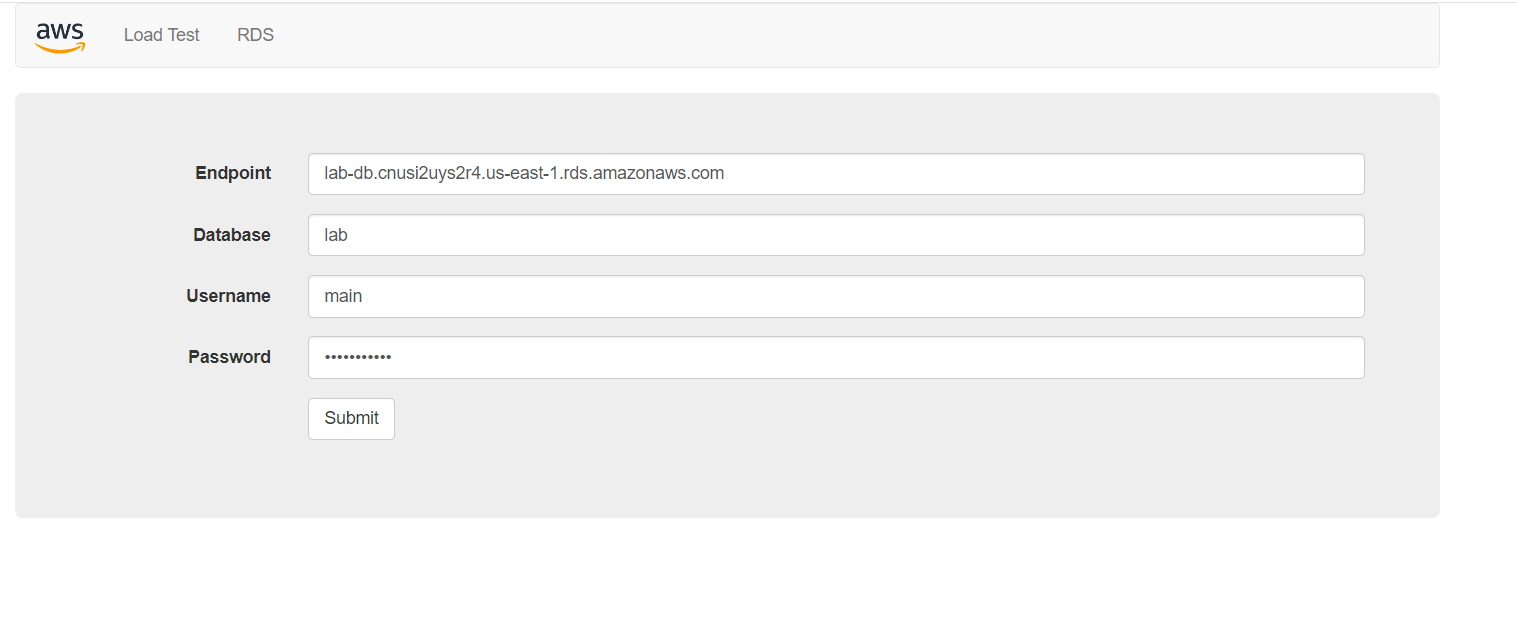
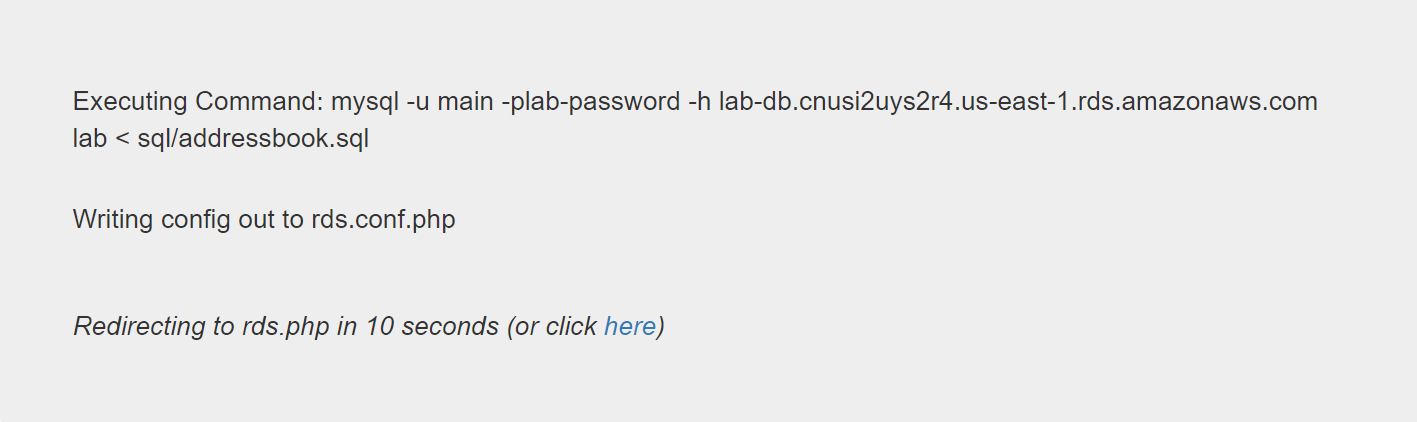
Step 12: Under DB instance class, select *Burstable classes*. Then choose, *db.t3.micro* under the dropdown menu. 

Step 13: Under *storage*, select *SSD*. Then set the *allocated storage to 20*. Then under *connectivity,* select *the Lab-VPC*. Then, under *Existing VPC Security Groups,* select, *DB Security Group.* De-select, *default* in the same menu.Finally, under *additional configuration,* set the initial database name. Then, uncheck automated backups, encryption, and enhanced monitoring inside of the monitoring section.     

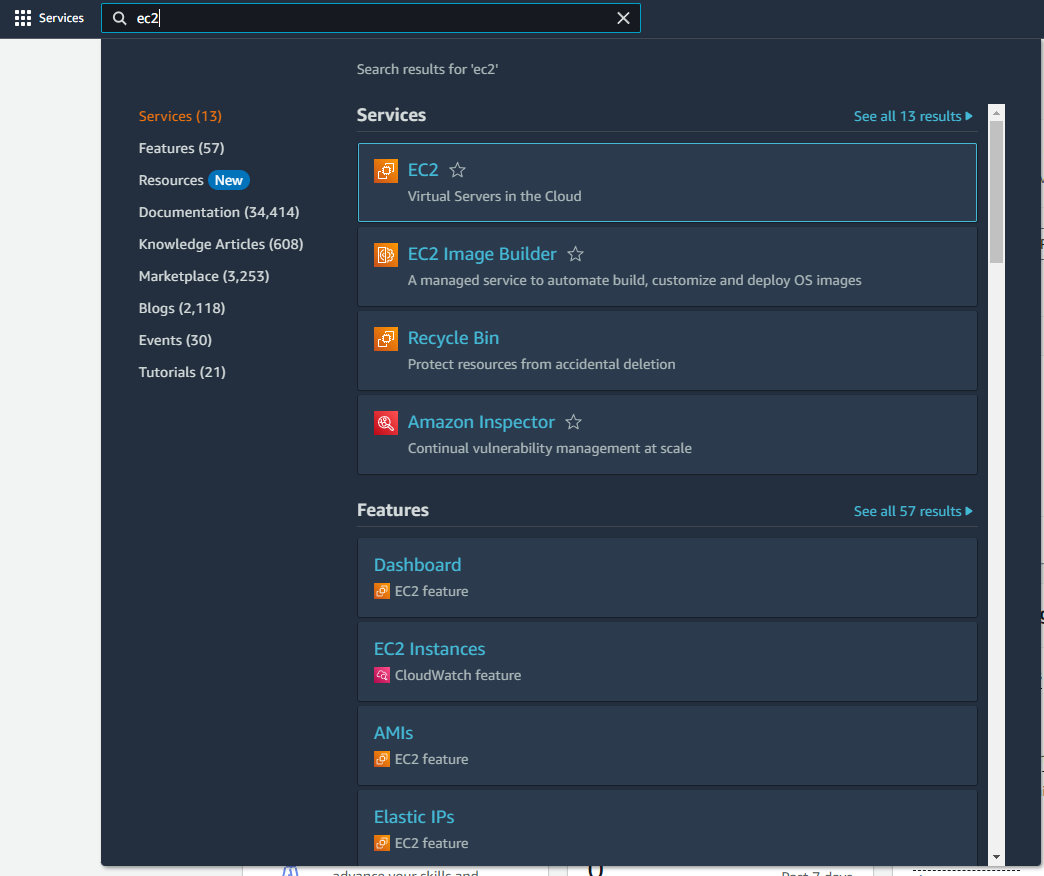
Step 14: Choose *Create Database. *

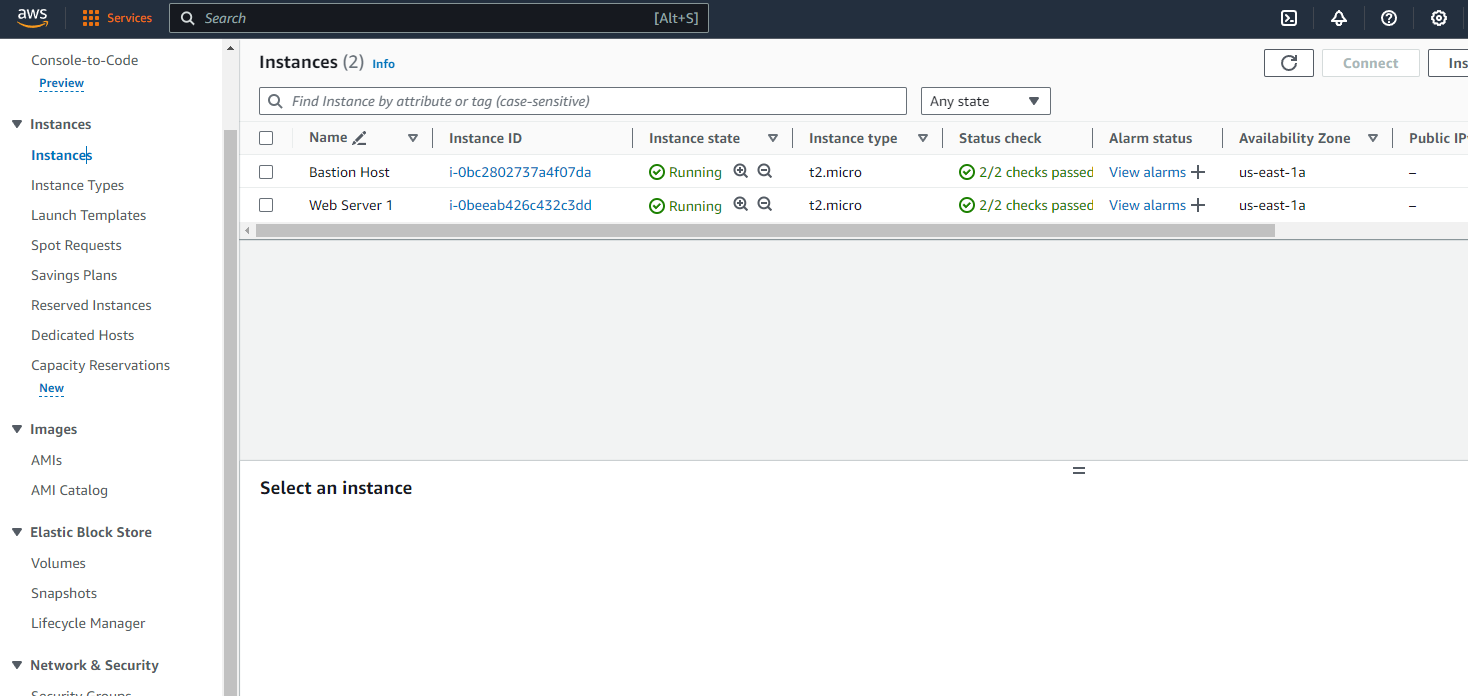
Step 15: The database creation will take approximately 4 minutes as it deploys to multiple databases.

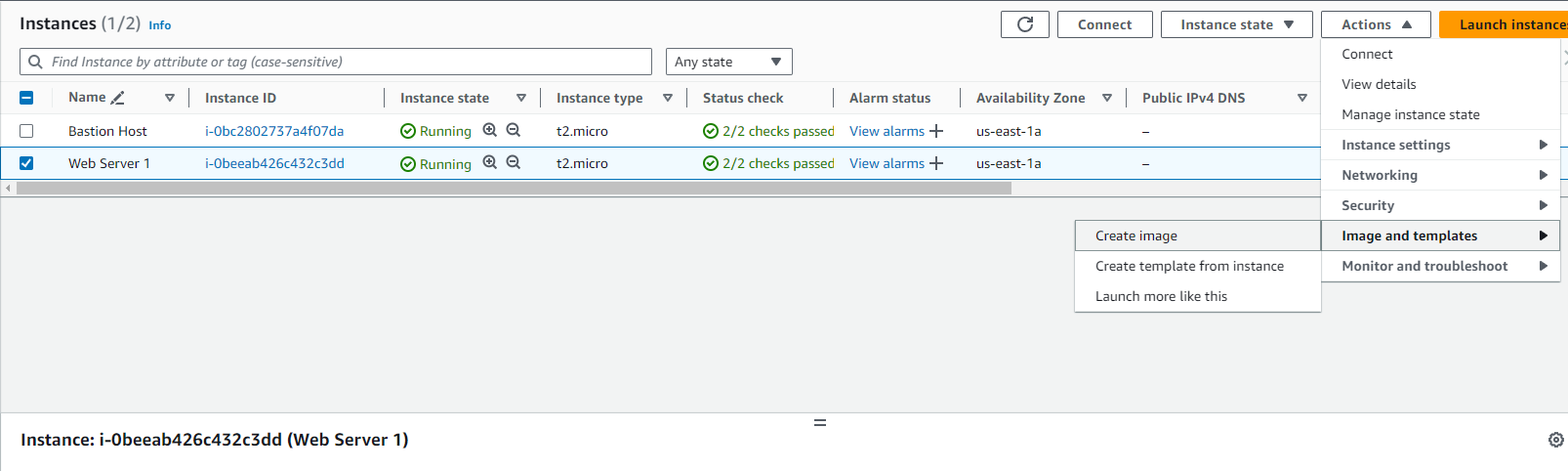
Step 16: Copy the endpoint link into a notepad file.

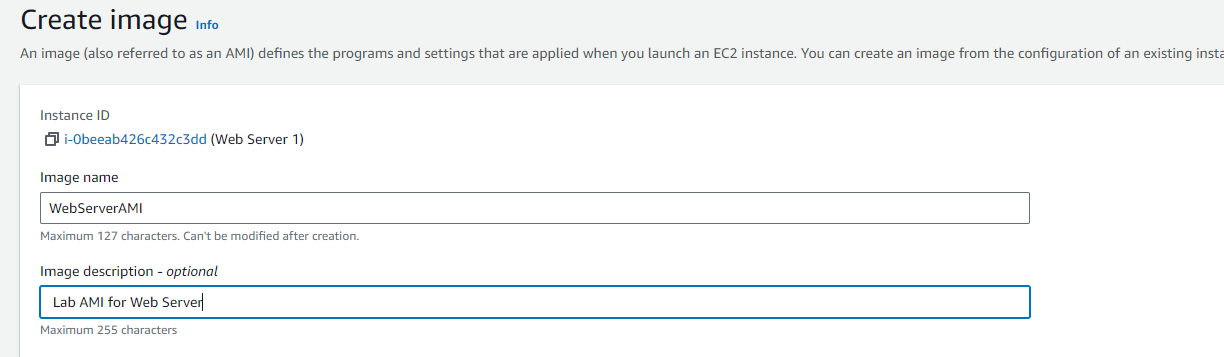
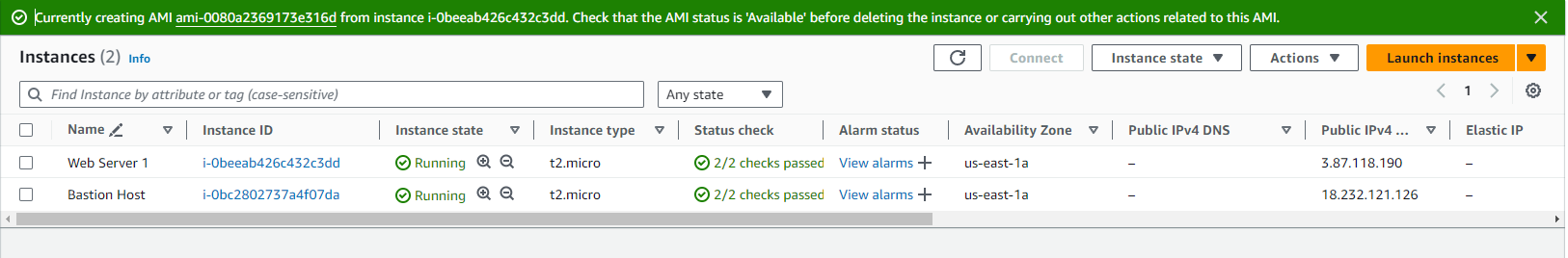
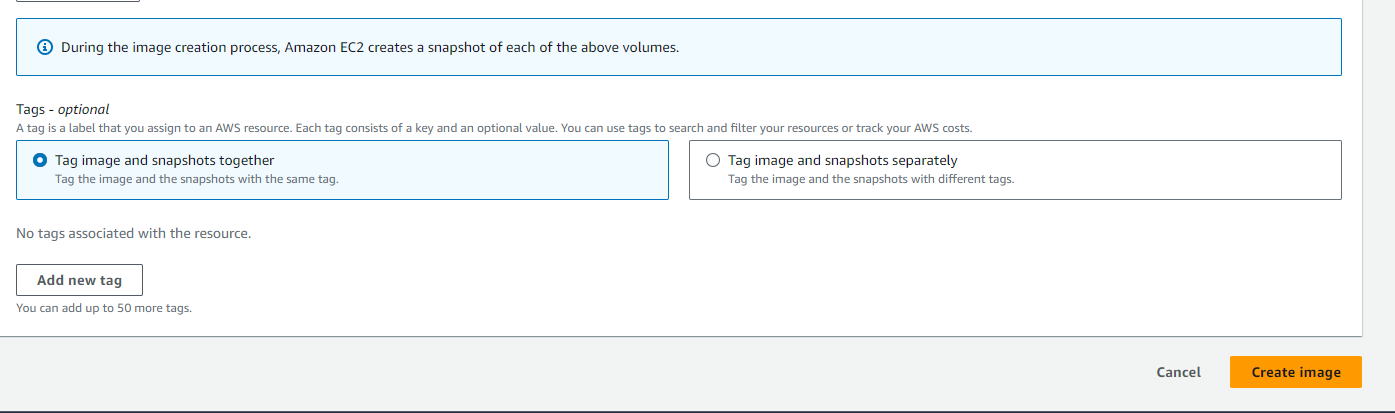
Step 17: Copy down the web server address and paste it into a new browser tab. Then select RDS and configure the fields with its corresponding values. Click submit when you accurately fill out the values.  Step 18: Once you click submit in the previous step, a message will pop-up for 10 seconds. Then, an address book will show up. Feel free to add contacts in the address book to test the database functionalities.  

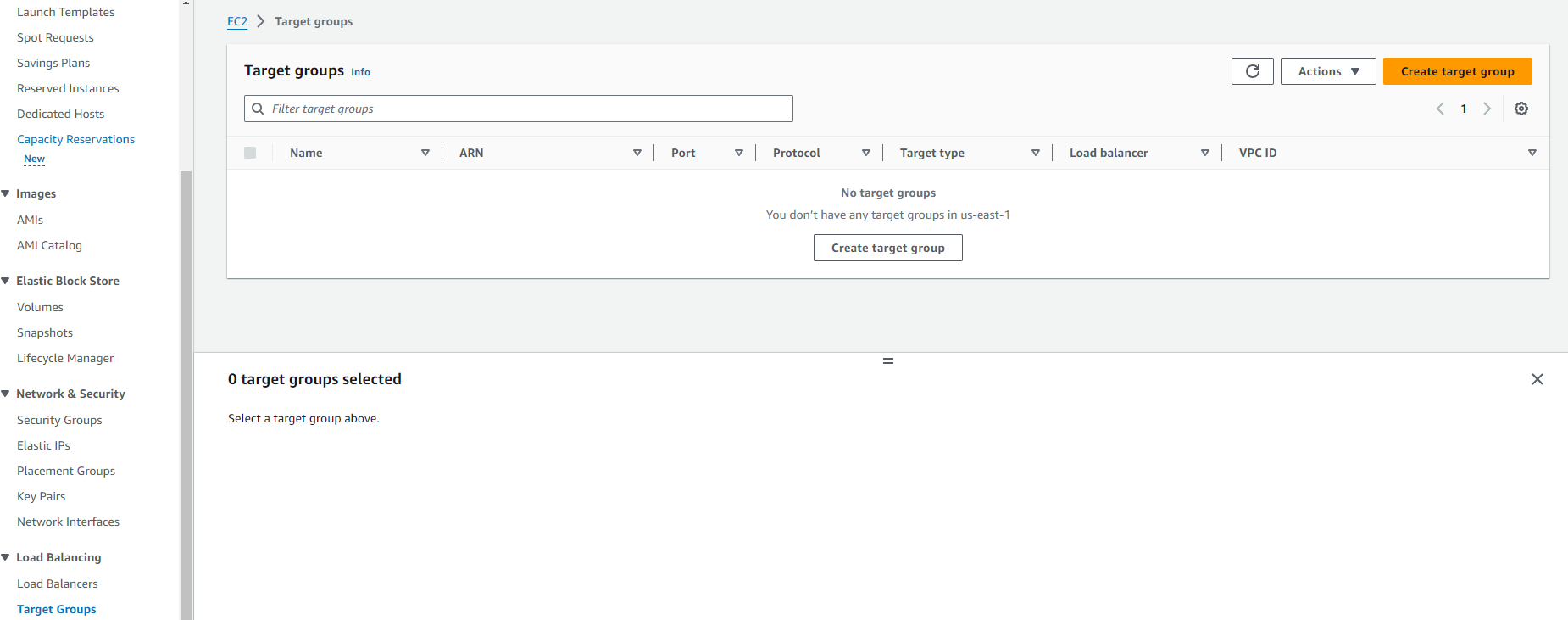
**AWS Module 10 Lab 6**

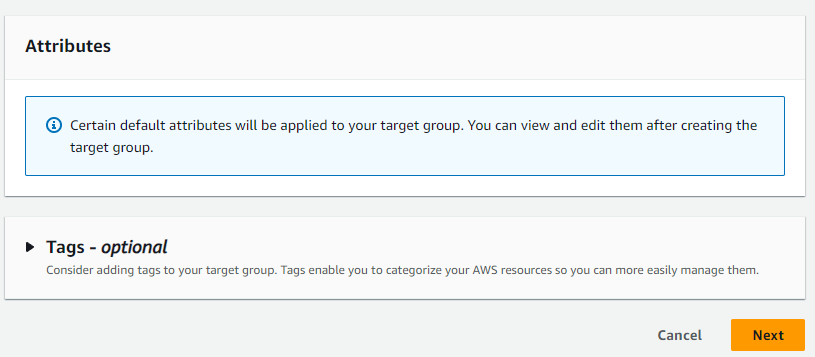
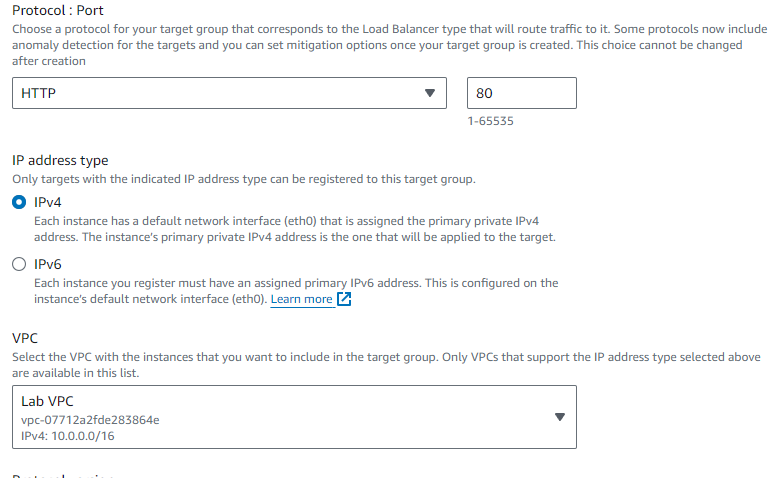
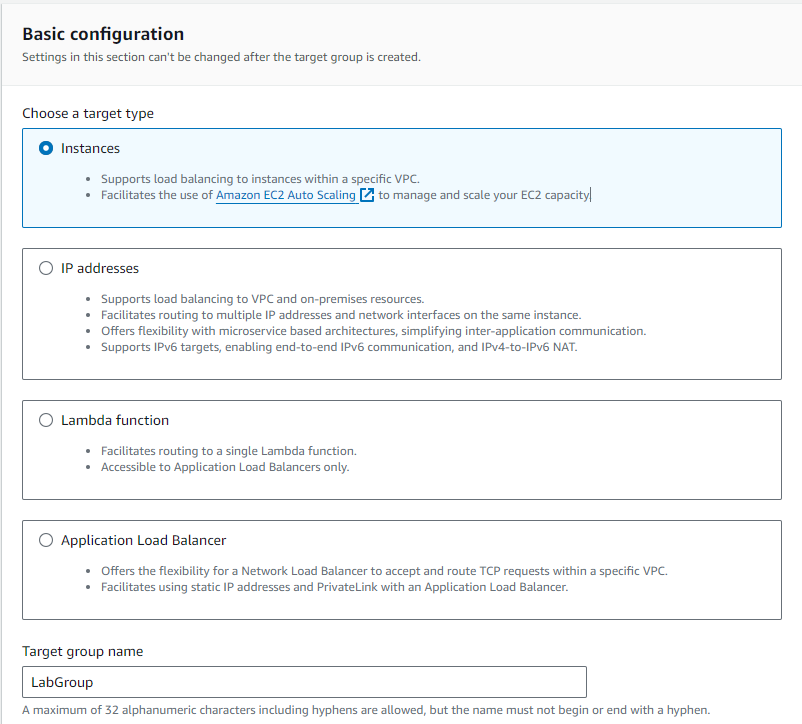
Step 1: In the services menu, search for EC2 and select the service. Then, select instances on the left navigation menu.

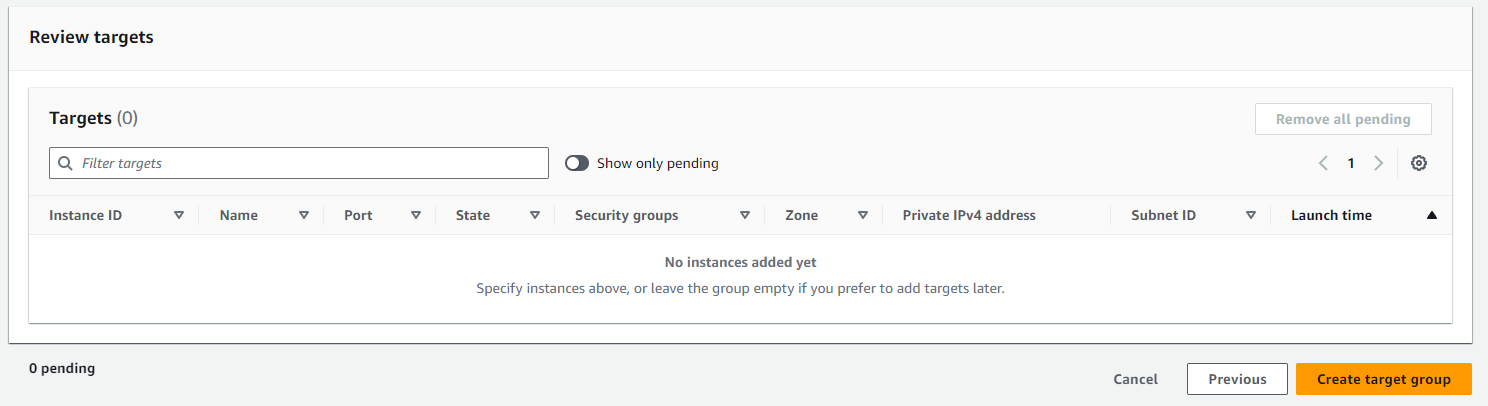
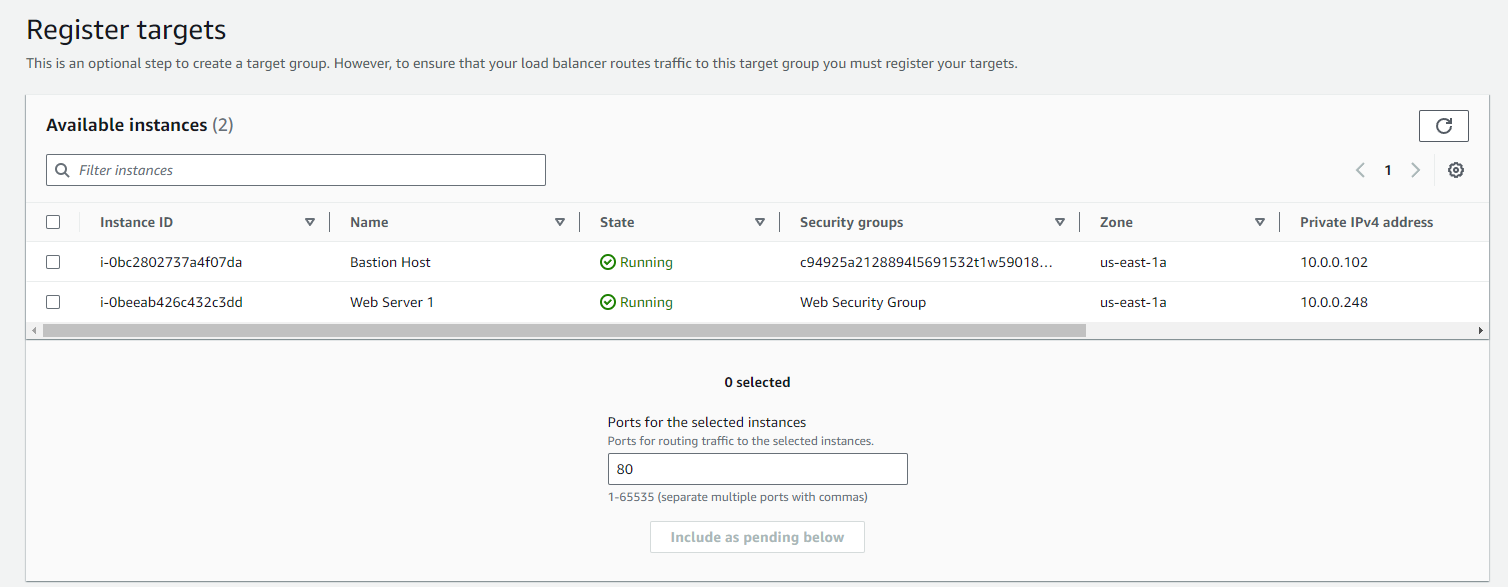
Step 2: Ensure that 2/2 checks have been passed for the Web Server 1 instance. 

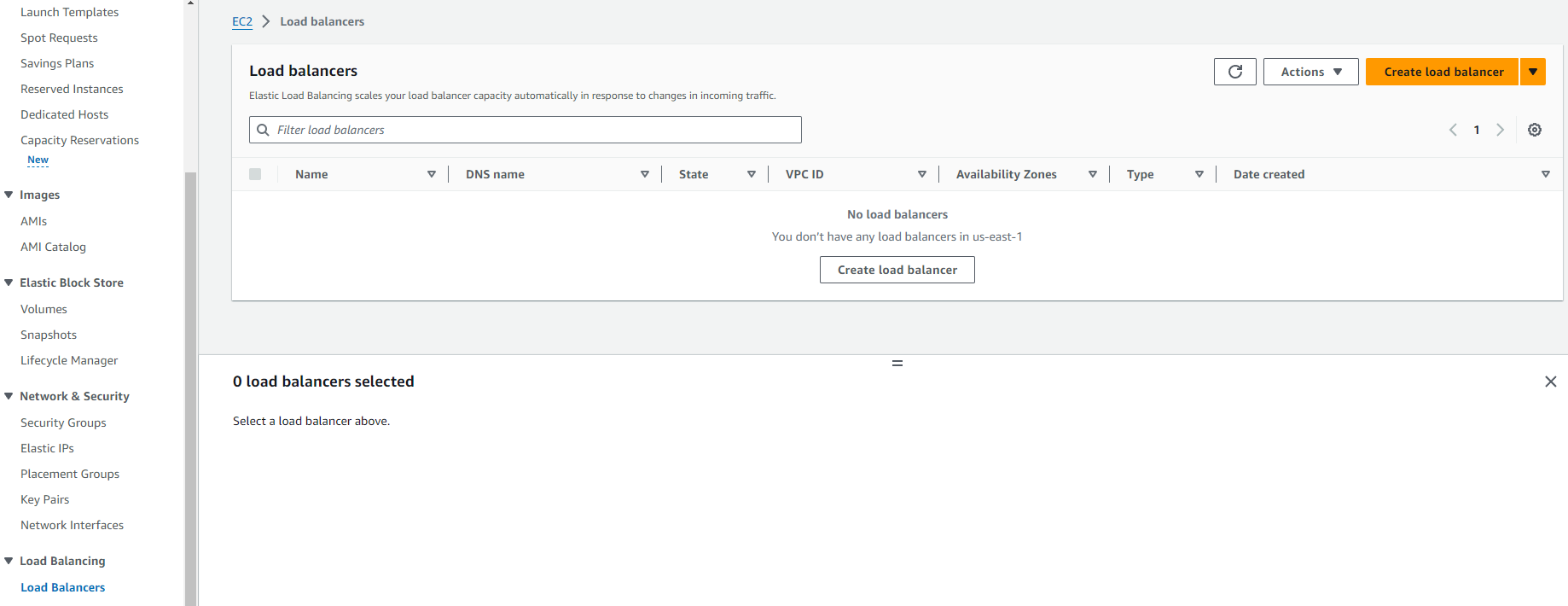
Step 3: Select the Web Server 1 instance and head to the action’s menu. From there, hover over the *Images and Templates* dropdown and select *Create Image.*

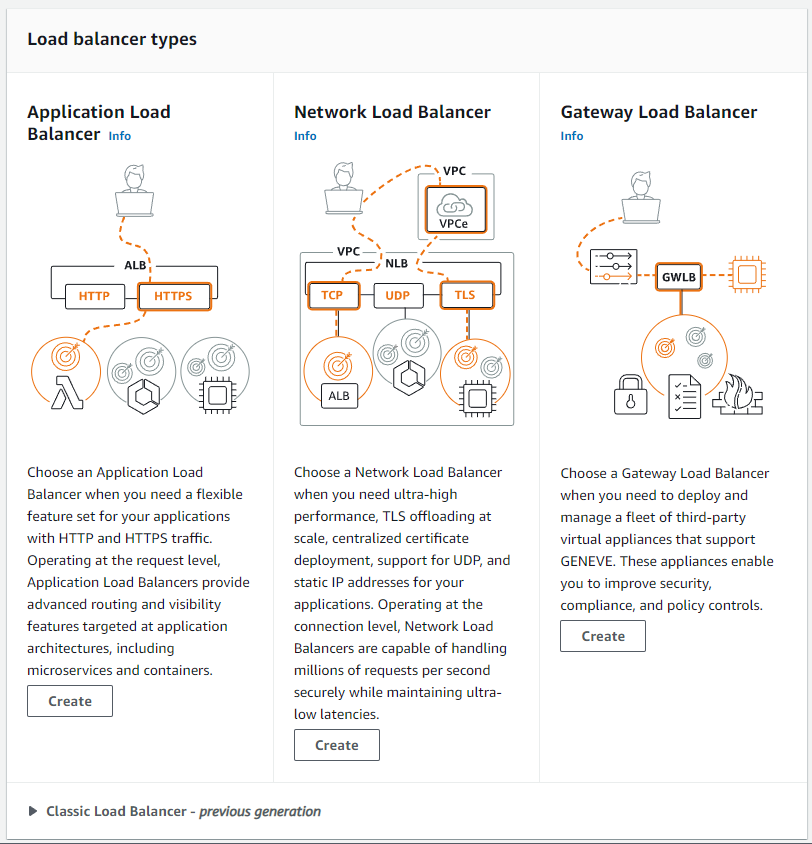
Step 4: Configure the annotate settings. Then press create by scrolling in the WebView. 

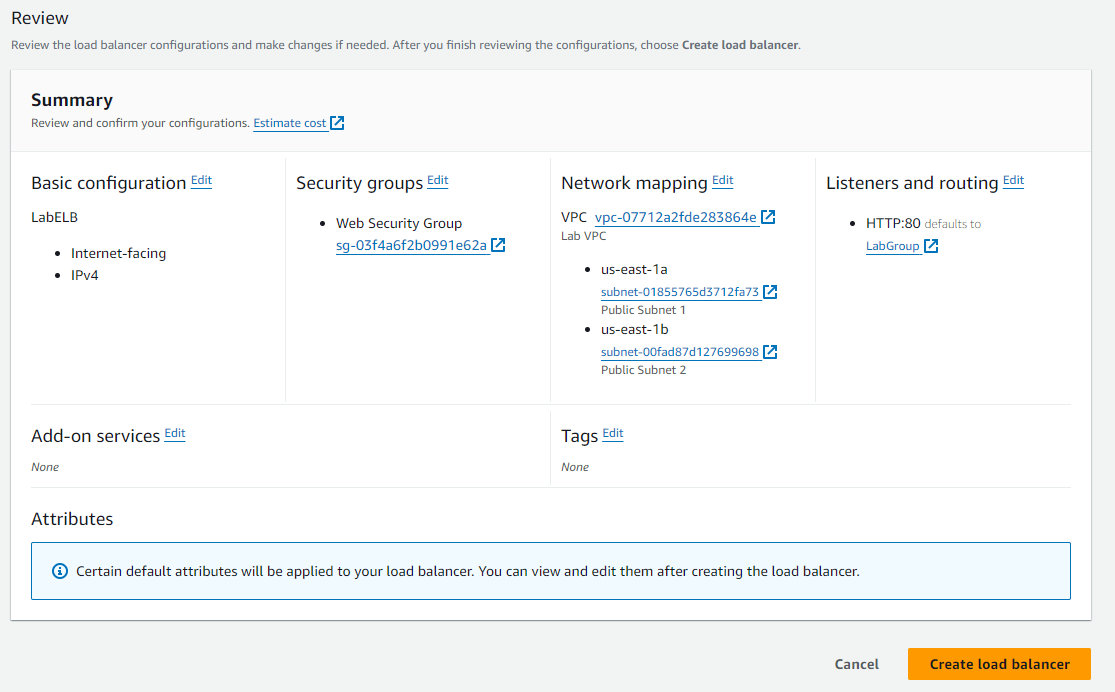
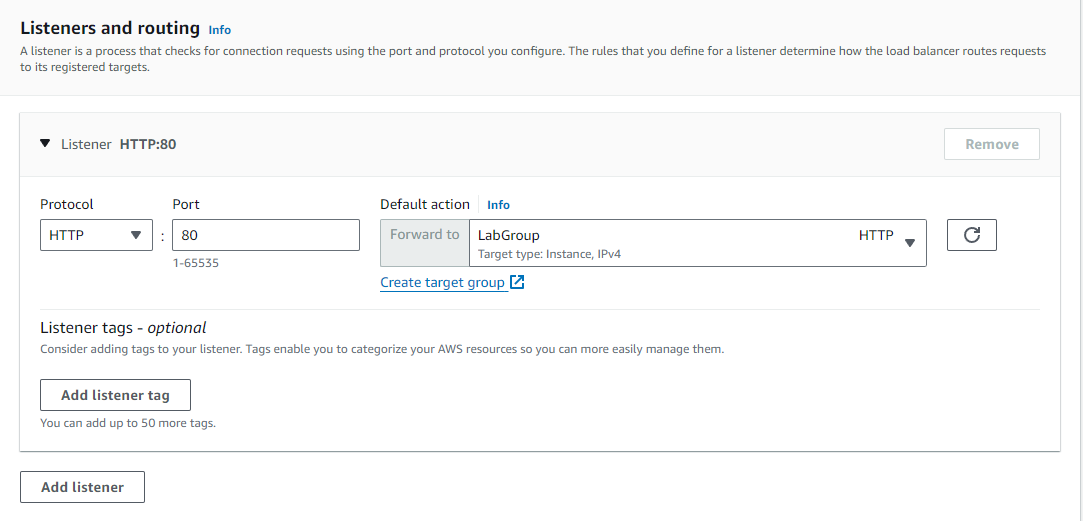
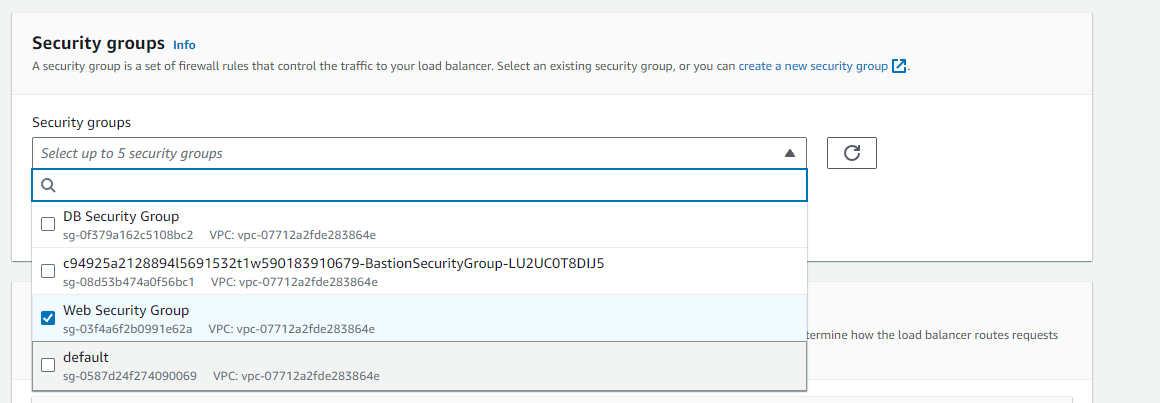
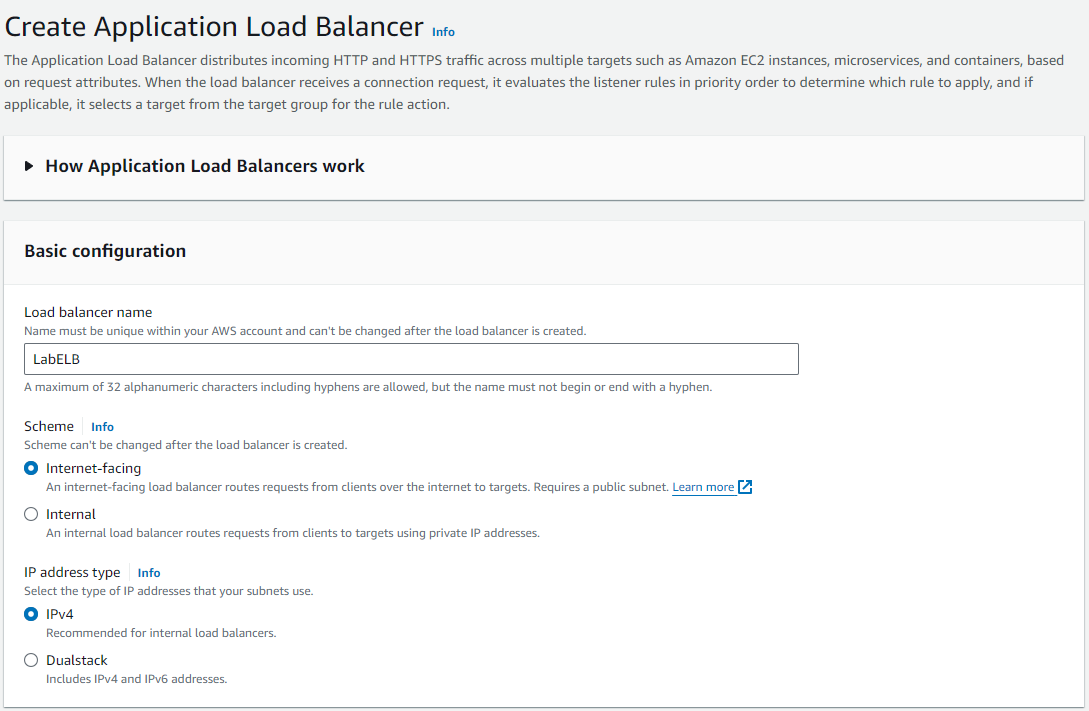
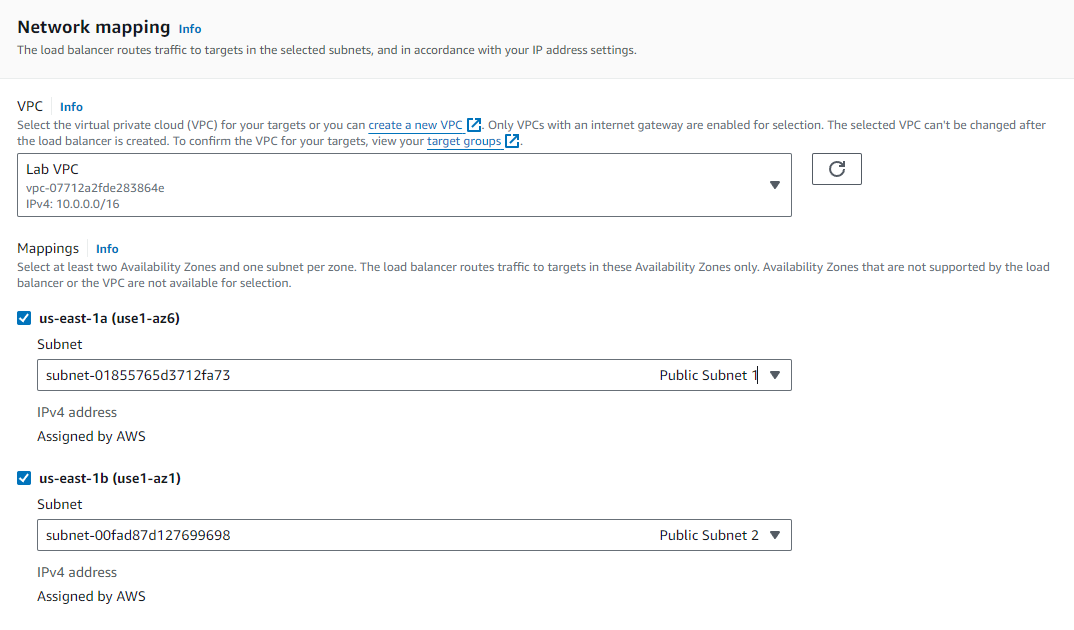
Step 5: Choose *Target Groups* from the navigation panel slated on the left. Then select, the *create target group* button highlghted in orange.

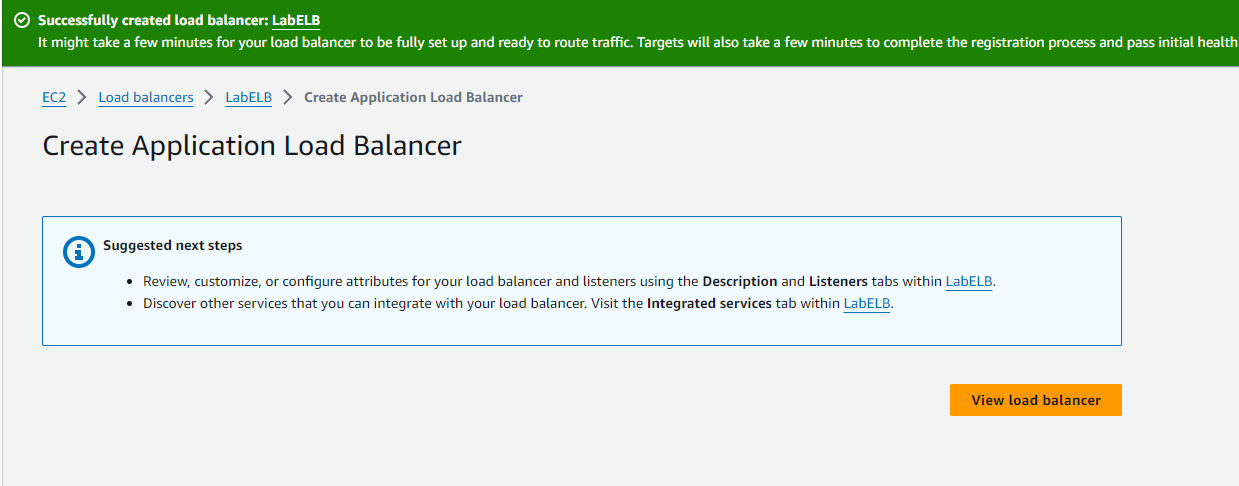
Step 6: Select *Instances* as the target type. Create a group name and select Lab VPC from the provided dropdown menu of options. Then select the *Next* button to move to the next page. 

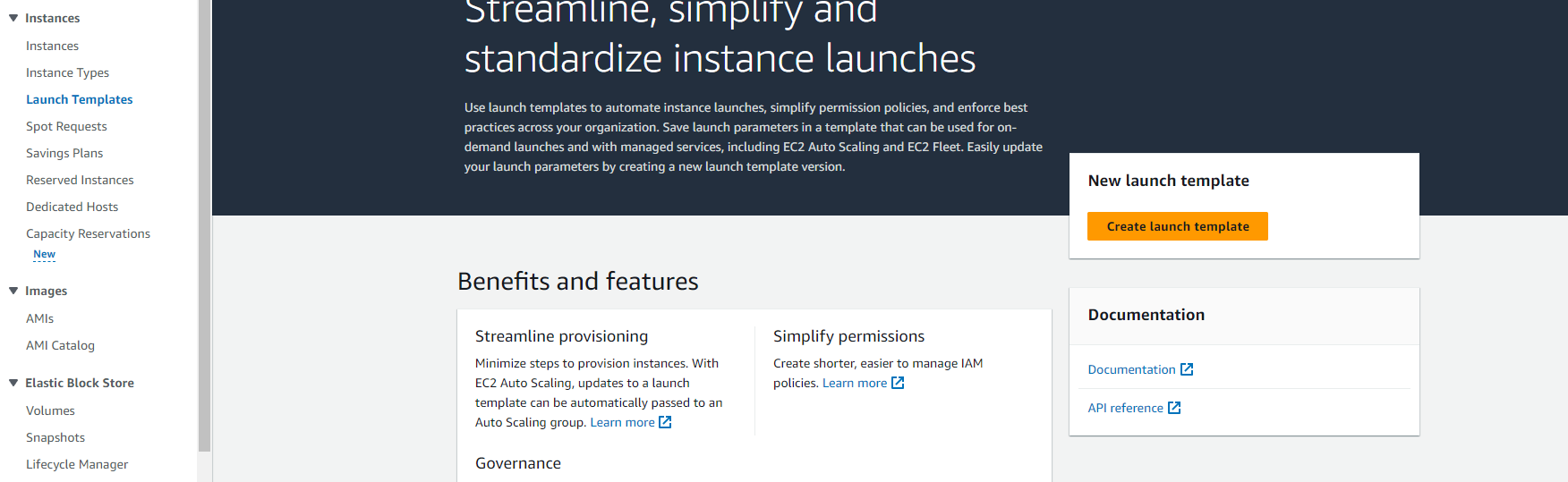
Step 7: Review the target settings. Once you are finished examining, create the target group. 

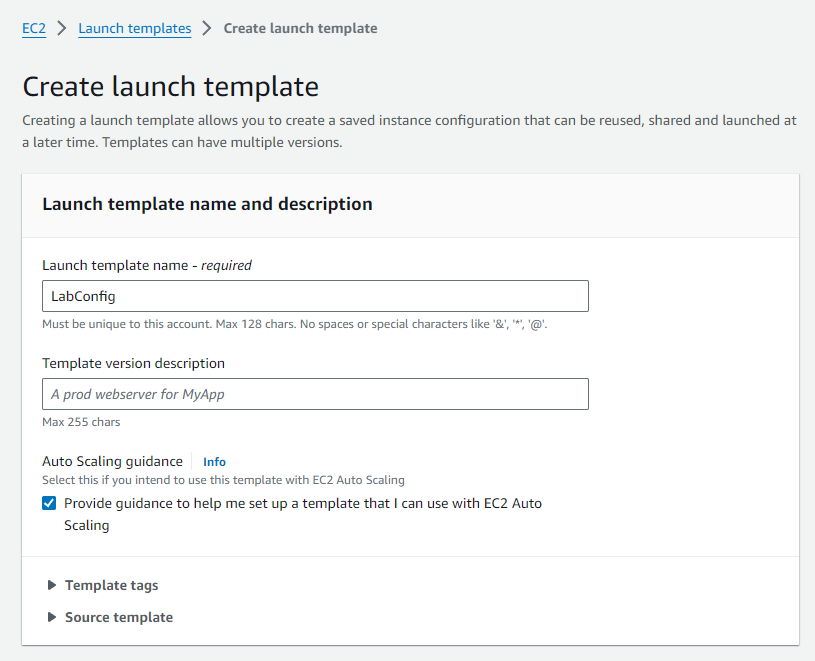
Step 8: Select the button that says Create load balancer from the the load balancers section. 

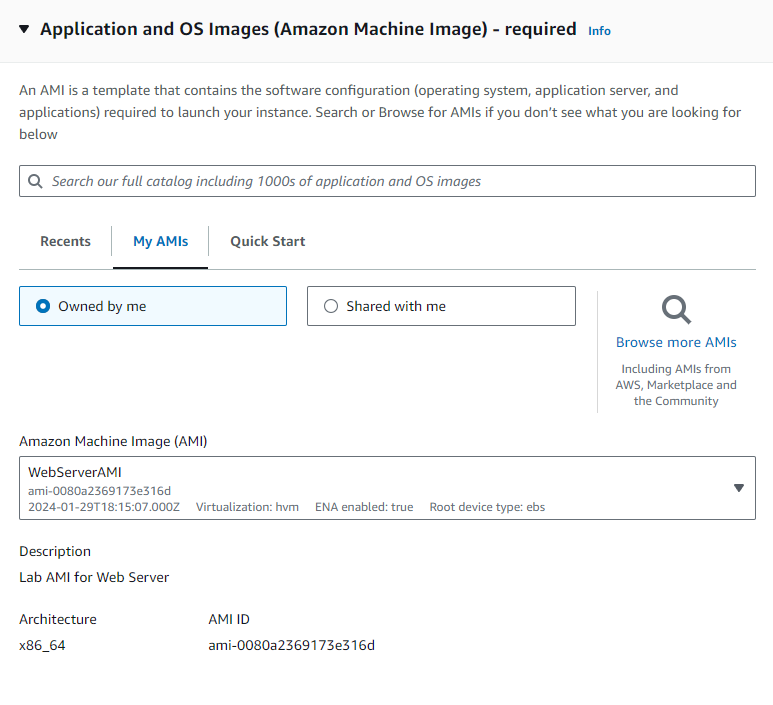
Step 9: Select *Create* under the Application Load Balancer.

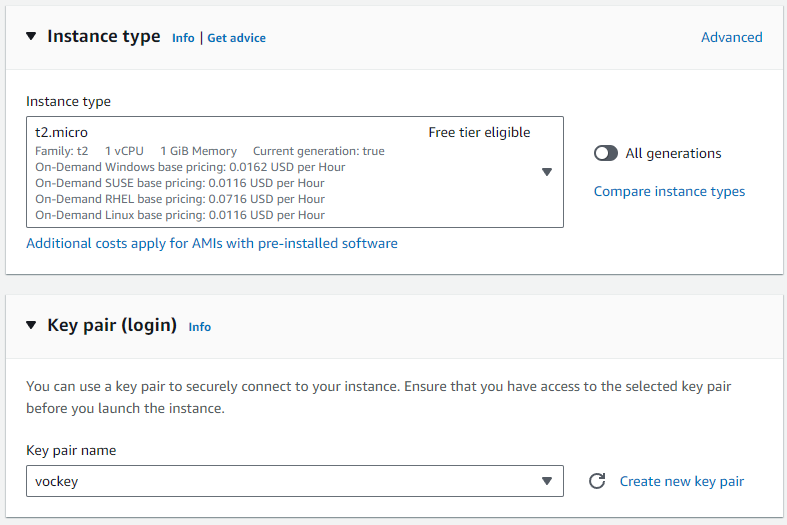
Step 10: Set the name of the load balancer, then scroll down to network mapping. Select Lab VPC. Then choose both available zones and select the public subnet for each one. Then, under the Security Groups section, un-select default, but select the Web Security Group. Change the default action to forward to the LabGroup. Then select *Create load balancer.* 

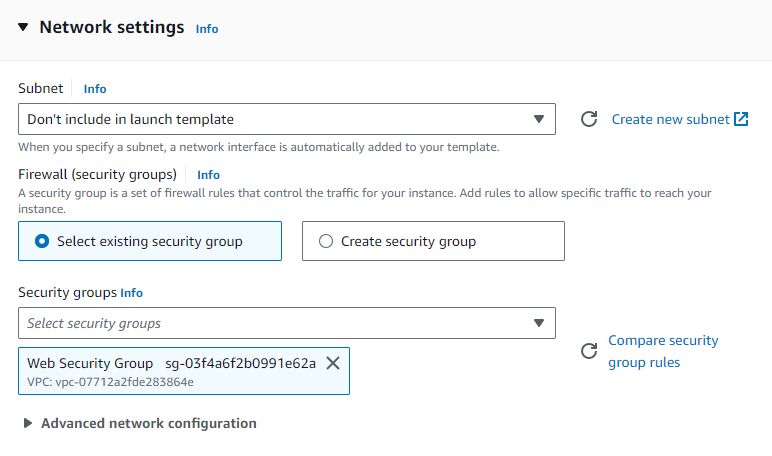
Step 11: Select view load balancer to check it out.

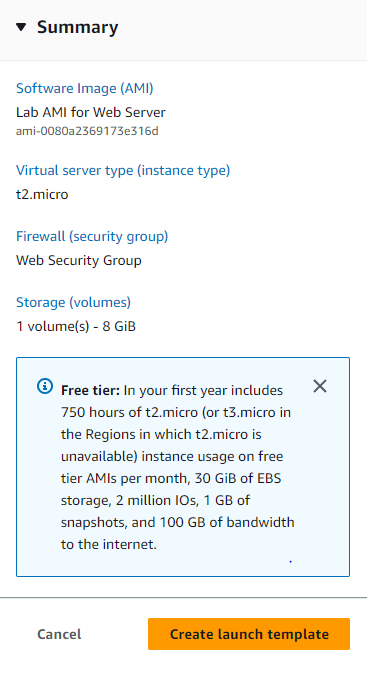
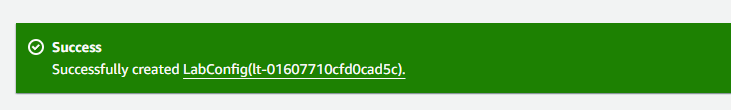
Step 12: Head to the launch templates section on the left in the navigation menu, then create launch template.

Step 13: Give it a launch template name and select the auto scaling guidance tickbox.

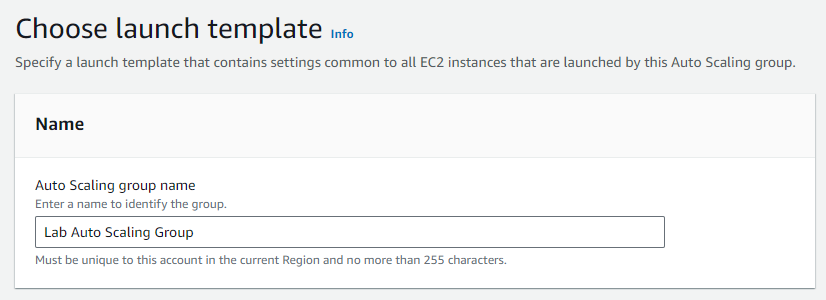
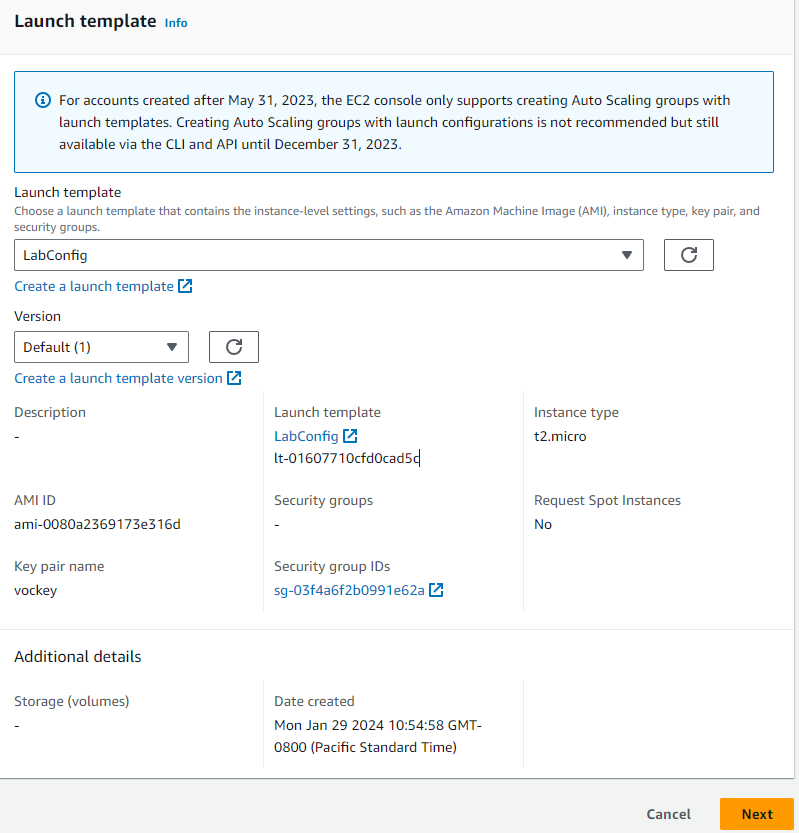
Step 14: Select My AMIs, then select WebServerAMI. 

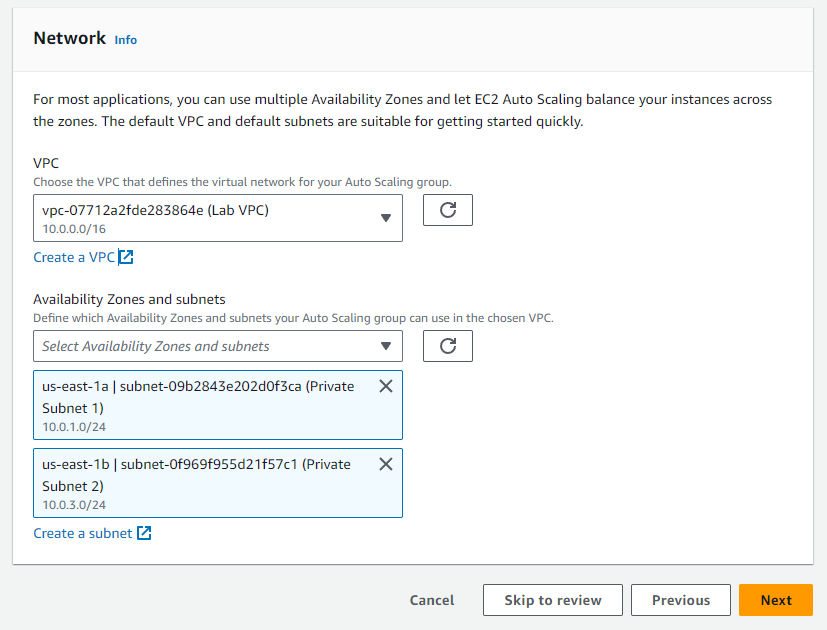
Step 15: Select t2.micro as the instance type and set the key pair login to vockey.

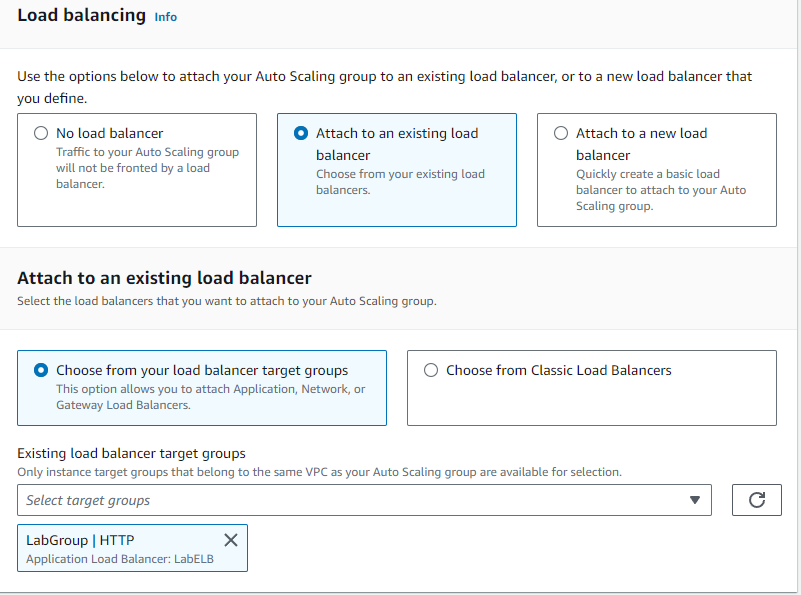
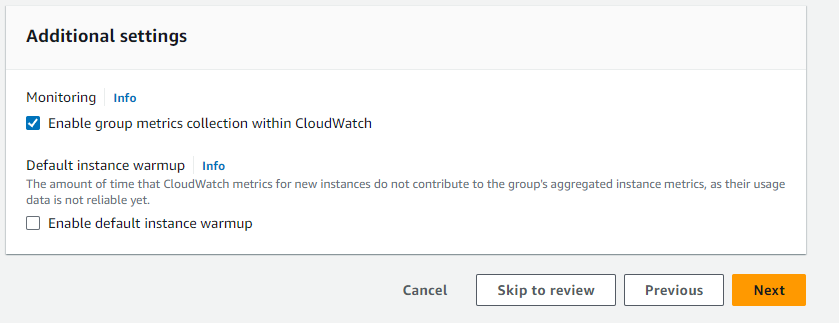
Step 16: Under Network Settings, select the Web Security group.

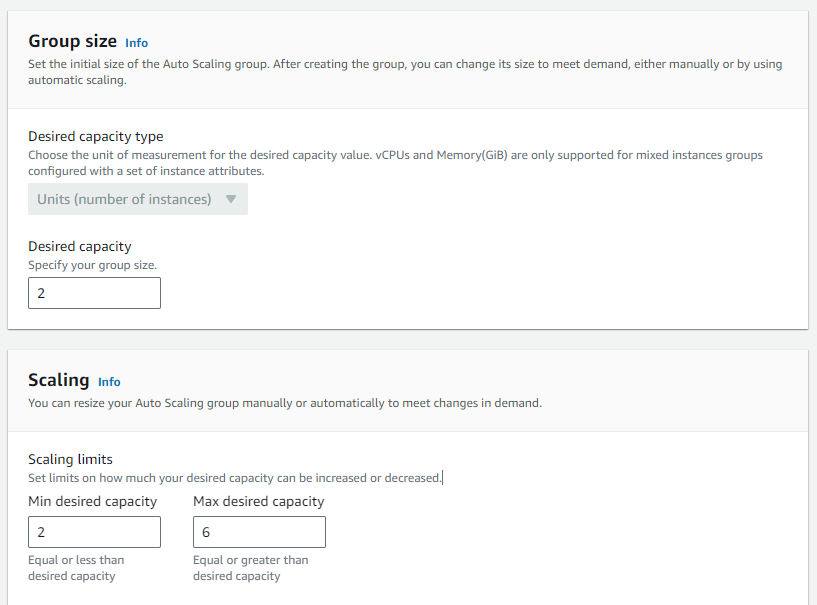
Step 17: Enable Detailed CloudWatch Monitoring after expanding the Advanced Details Section. Then, click create launch template.   
Step 18: Choose the LabConfig link from the success banner.

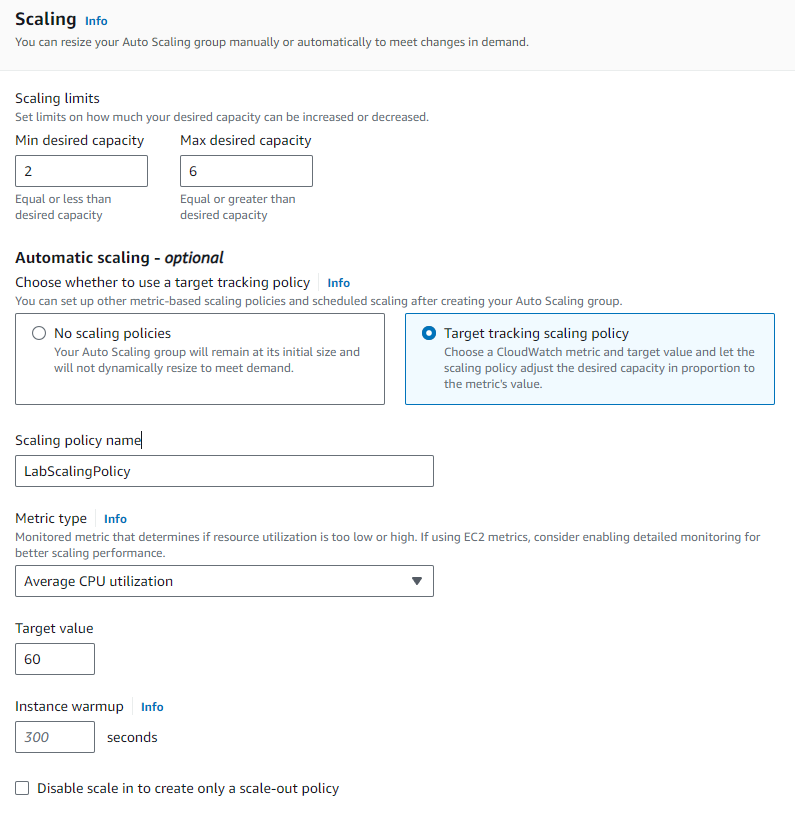
Step 19: Select the launch template. Then head to the actions menu, and select create auto scaling group. 

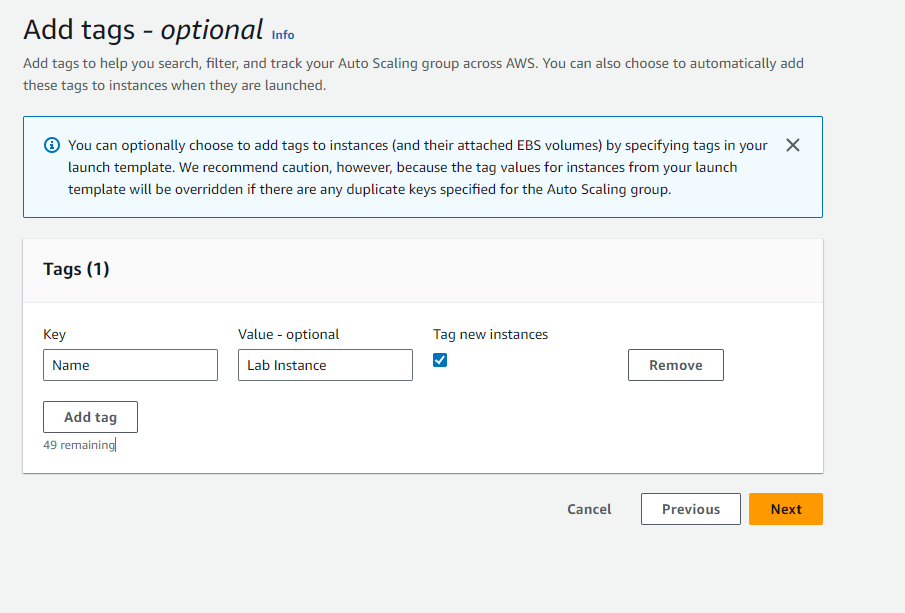
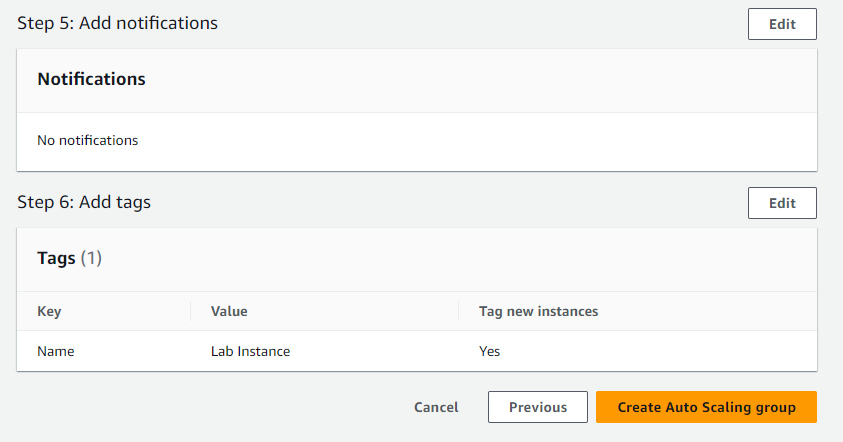
Step 20: Configure a group name for the auto scaling. Verify that the LabConfig template is selected. Then choose next. 

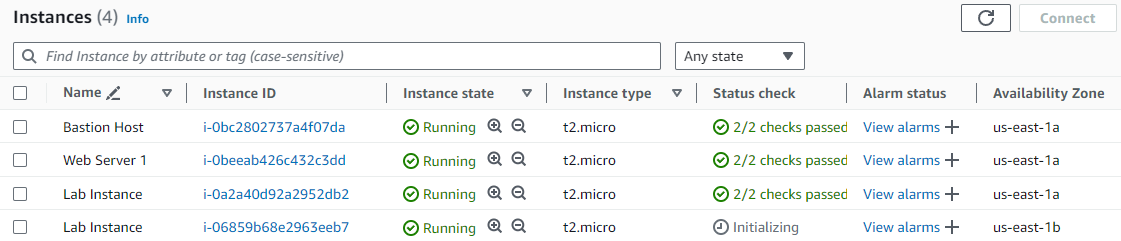
Step 21: Select Lab VPC and choose both private subnets vailable.

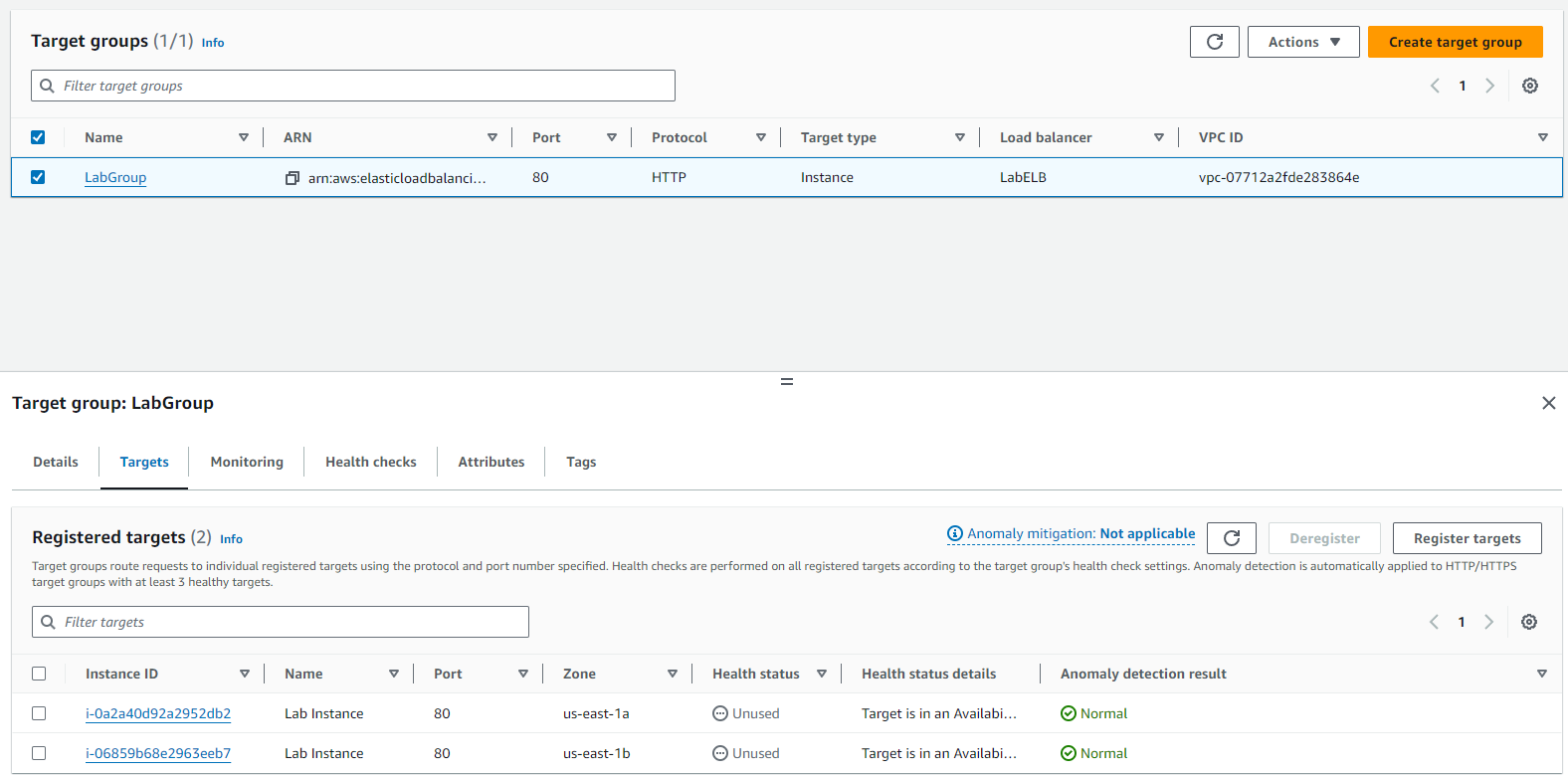
Step 22: Choose *Attatch to a new load balancer.* Enable the annotated checkbox and select next.  

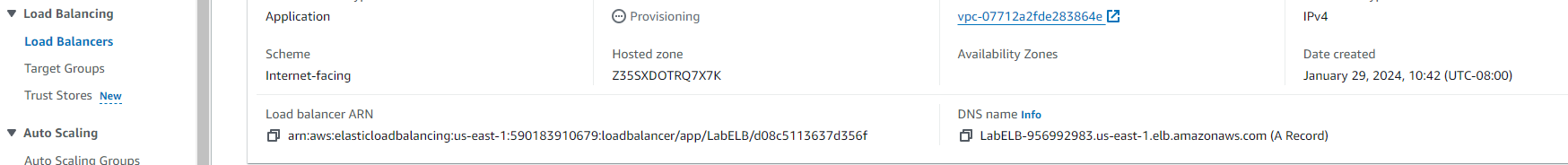
Step 23: Select the desired capacity to be 2. Configure the minimum desired capacity to be 2 and maxium to be 6.

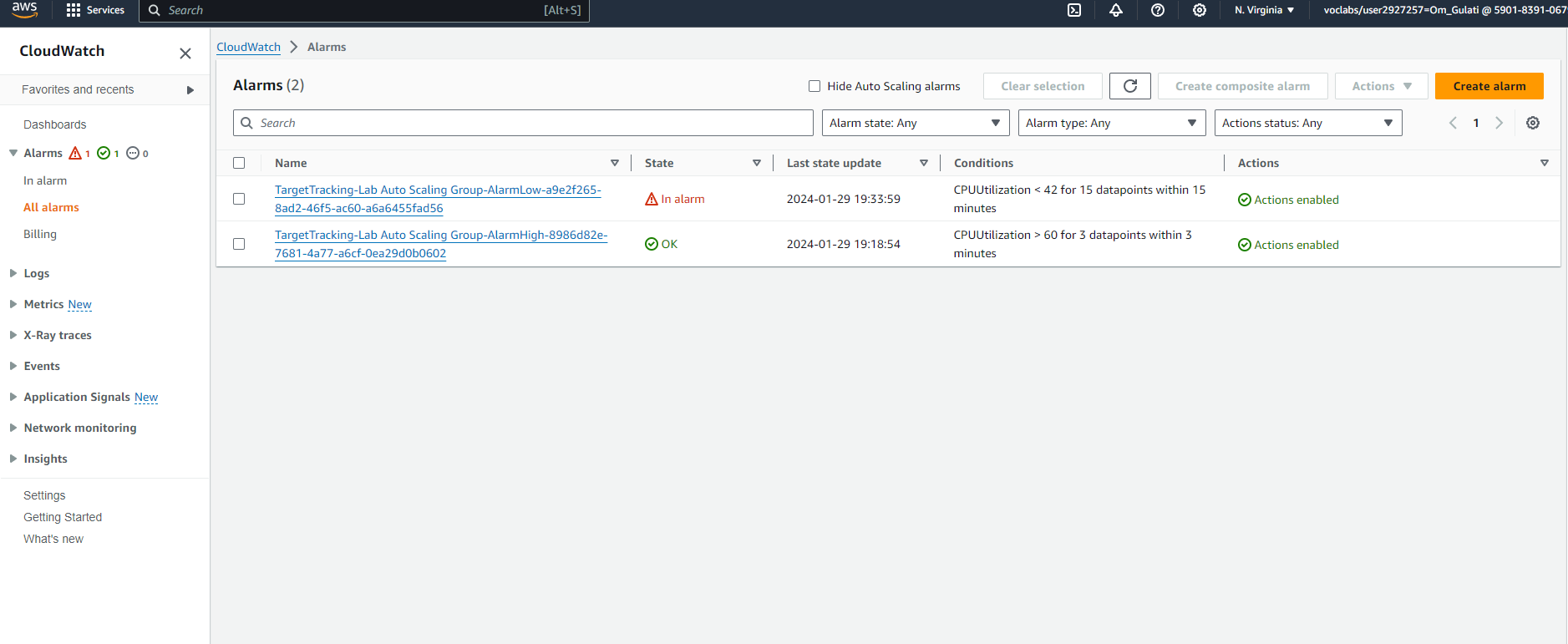
Step 24: Configure the following scaling values by choosing *Target tracking scaling policy.* Then choose next.

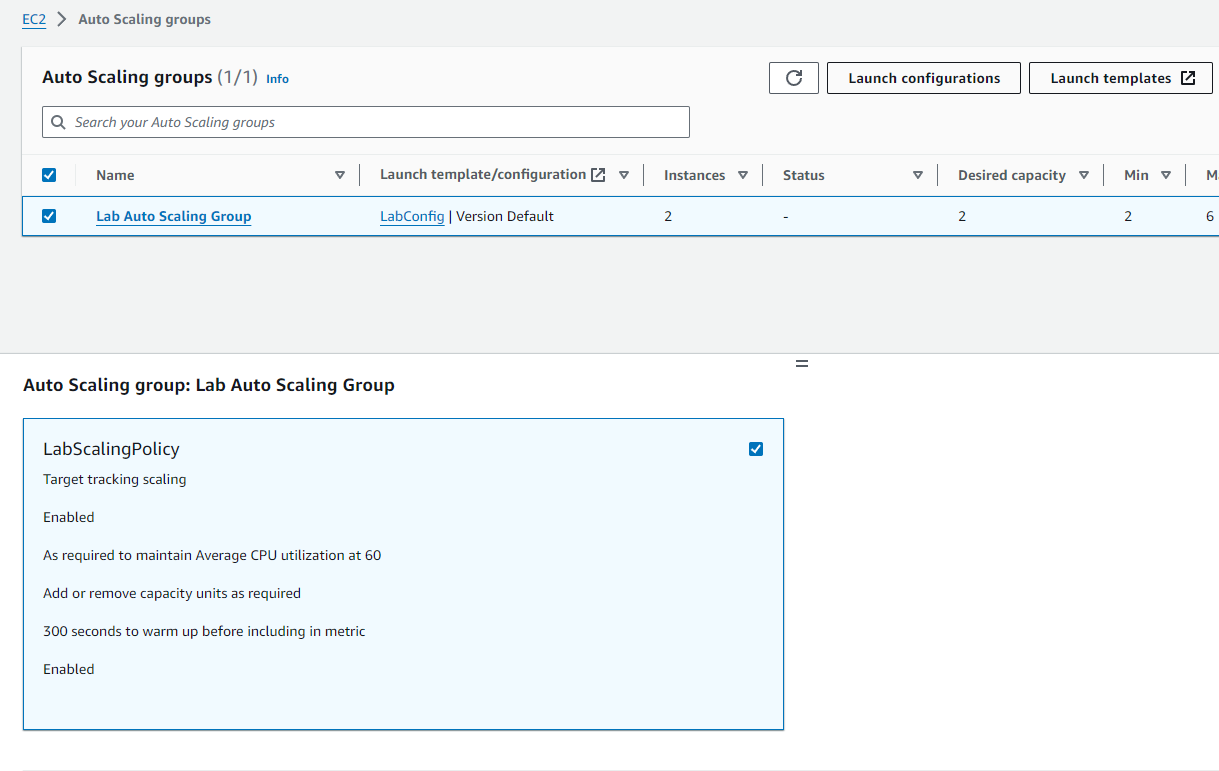
Step 25: Create a tag and set a key and optional value. Then choose next. Then, review the details and create the auto scaling group on the next page. 

Step 26: In the left navigation panel, select instances and ensure two new lab instances have been created.

Step 27: Head to target groups throught the left navigation panel. Select lab group. Then, select targets. 

Step 28: Under the load balancer sub section, select the available LB and copy the DNS name from the details section. Paste it into a new tab.

Step 29: Head to the Cloud Watch service through the services menu. Select all alarms and examine.

Step 30: Select *Lab Auto Scaling* in the EC2 dashboard under the Auto Scaling section. Then select LabScalingPolicy. Head to the actions menu and click edit. Change the target value to 50 instead of 60.

Step 31: Congratulations you have completed all 6 AWS labs!

Problems:

I faced a recurring issue in my lab sessions where, after clicking "end lab" as per the instructions, the labs wouldn't open for extended periods, sometimes days, slowing down progress on my writeup. Also, in Lab 4, I encountered an unexplained problem attaching an instance to my volume, which was resolved by reloading the page.

Another challenge occurred in Lab 6 when I couldn't paste my DNS name, leading to the attached image. This was traced back to a misconfiguration in an earlier step.

Despite these challenges, the overall lab experience remained relatively smooth, thanks to clear and easily understandable instructions.

Conclusion:

These labs helped me learn how to create databases, which is valuable for my software-related goals. Additionally, using SSH with Putty was interesting and served as a reminder that Putty is versatile, going beyond console access to networking devices. It's a powerful tool in various situations.