**AWS Solution Architect Associate**

**What is AWS?**

AWS stands for amazon web services is a cloud provider. They provide you with servers and services that you can use on demand and scale easily.

**AWS CLI:**

Command : aws configure, aws iam list-users

|  |  |
| --- | --- |
| Commands | usage |
| aws configure | To connect your local machine to your aws account. We should give our secret credentials and default regions. |
| aws iam list-users | To see all iam users. |

**What is IAM Role?**

IAM role is an IAM entity that defines a set of permissions with credentials for making requests to AWS SERVICES, and will be used by an AWS Service. We attach a role to EC2 Instance which enables permission to talk with other aws services.

**What is an IAM Policy?**

JSON documents that define a set of permissions for making requests to AWS services, and can be used by IAM Users, User groups, IAM roles.

Simply put, list of resources to which the actions should apply.

**What is EC2 instance?**

EC2 instance is a virtual machine running on a physical host.

SSH: Secure Shell. Default port is 22.

command to connect to EC2 instance: ssh -i “pem file name” ec2-user@<public IPv4 addres>

**What is main difference between dedicated hosts and dedicated instances?**

With Dedicated Host the physical server is basically yours. It does not change, **it's always the same physical machine for as long as you are paying.**

Dedicated Instance does not work like this. Your instance runs on some dedicated hardware. It’s not lockdown to you. If you stop/start instance, you can get some other hardware somewhere else. Basically, the hardware is "yours" (you are not sharing it with others) for the time your instance is running. You stop/start it, you may get different physical machine later on (maybe older, maybe newer, maybe its specs will be a bit different), and so on. So your instance is moved around on different physical servers - whichever is not occupied by others at the time.

**What is EC2 User Data?**

EC2 User Data is used to bootstrap your EC2 instances using a bash script. This script can contain commands such as installing software/packages, download files from the Internet, or anything you want.

**What is compute power?**

The ability of a computer to perform work, often considered in terms of the number of instructions that can be carried out in a given time. It is also referred as processing power.

**What are EC2 Instance types?**

We can use different types of EC2 instances that are optimised for different use cases.

General Purpose: General purpose instances provide a balance of compute, memory and networking resources, and can be used for a variety of diverse workloads. These instances are ideal for applications that use these resources in equal proportions such as web servers and code repositories. Ex: t2.micro

Compute Optimized: Great for compute-intensive tasks that require high performance processors. Batch processing workloads, Media transcoding, High performance web servers, High performance computing (HPC),Scientific modelling & machine learning, Dedicated gaming servers.

Memory Optimized: Memory optimized instances are designed to deliver fast performance for workloads that process large data sets in memory. High performance, relational/non-relational databases, Distributed web scale cache stores, In-memory databases optimized for BI (business intelligence), Applications performing real-time processing of big unstructured data.

Storage Optimized: Storage optimized instances are designed for workloads that require high, sequential read and write access to very large data sets on local storage. They are optimized to deliver tens of thousands of low-latency, random I/O operations per second (IOPS) to applications. High frequency online transaction processing, relational and non-relational databases, Cache for in-memory databases (for example Redis), Data warehousing applications, Distributed file systems.

**What is higher bandwidth?**

Having a higher bandwidth means you will be able to achieve a higher data transfer rate which in turn leads to shorter download times. This is especially significant when downloading large files.

**What is Low latency?**

Low latency describes a computer network that is optimized to process a very high volume of data messages with minimal delay (latency). Simply we can say higher data transfer rate with minimal delay.

**What are different placement groups?**

Cluster: For critical applications.

Spread: For distributed applications.

Partition: For high performance applications.

**What is Public IPv4 and Private IPv4?**

Public IP is accessible around the globe and private IP is only accessible between instances in virtual private cloud(VPC).

**What is Elastic Network Interfaces(ENI)?**

ENI is a logical component in VPC and they are what gives EC2 instances access to the network.

**What is EC2 Hibernate?**

EC2 Hibernate is used to preserve the in-memory RAM state. When hibernate is enabled all and you instructed to stop-hibernate instance then Hibernation saves the contents from the instance memory (RAM) to your Amazon Elastic Block Store (Amazon EBS) root volume.

1. The EBS root volume is restored to its previous state
2. The RAM contents are reloaded
3. The processes that were previously running on the instance are resumed
4. Previously attached data volumes are reattached and the instance retains its instance ID
5. So that it's ready to be resumed to the desired state whenever needed.

**What is EBS?**

EBS stands for Elastic Block Store, an EBS volume is a network drive you can attach to your instances while they run. When we create an instance by default a EBS volume is created and attached to our instance.

EBS volumes allows your instances to persist data even after the termination.

They bound to specific availability zone.

**What is EBS Snapshots?**

EBS Snapshots can be considered as a backup for EBS volumes. As we know EBS volumes are zone specific so we can’t use an EBS volume in us-east-1 in us-east-2.

So what we can do is create a snapshot from EBS volume. From that snapshot we can create another EBS volume in us-east-2 zone.

**What is AMI?**

AMI stands for Amazon Machine Image. AMI are provided by AWS. But we can configure our own AMI from an EC2 instance.

1. Start an EC2 instance and customize it.
2. Stop the instance for data integrity.
3. Build an AMI – this will also create EBS Snapshots.
4. Launch instances from other AMI’s.

**What is EC2 Instance Store?**

If you need high performance hardware disk use EC2 Instance Store. You would like to have a high-performance local cache for your application hosted on an EC2 instance. You don't mind losing the cache upon the termination of your EC2 instance.

EBS volumes are network drives with good but limited performance.

**EBS Volume Types**

EBS Volumes come in 6 types

**gp2 / gp3 (SSD) or General Purpose SSD**: Cost effective low latency. In gp3 we can independently set the IOPS and throughput, in gp2 IOPS and throughput are linked to each other.

**Provisioned IOPS SSD**: Critical business applications with sustained IOPS performance or applications that need more than 16,000 IOPS.

**Hard disk drives SSD**: Cannot be boot volume, 125mb to 16tb.

**What is throughput in AWS?**

Throughput is the measure of the **amount of data transferred from/to a storage device in a second**. Typically stated in KB/MB/GB/s (e.g., if a storage device can write 1000 blocks of 128K each, throughput is 1000\*128K/s = 128MB/s).

**What is EBS Multi Attach?**

Attach the same EBS volume to multiple EC2 instances in the same AZ.

**What is EFS?**

EFS is a network file system (NFS) that allows you to mount the same file system on EC2 instances that are in different AZs.

Amazon Elastic File System (Amazon EFS) provides a simple, scalable, fully managed elastic NFS file system for use with AWS Cloud services and on-premises resources. It is built to scale on demand to petabytes without disrupting applications, growing and shrinking automatically as you add and remove files, eliminating the need to provision and manage capacity to accommodate growth. Amazon EFS is designed to provide massively parallel shared access to thousands of Amazon EC2 instances, enabling your applications to achieve high levels of aggregate throughput and IOPS with consistent low latencies.

**What is difference between AWS regions and AWS availability zones?**

AWS Regions are large and widely dispersed into separate geographic locations. Availability Zones are distinct locations within an AWS Region that are engineered to be isolated from failures in other Availability Zones.

**What is difference between Edge Locations and Availability zones in AWS?**

Both the Edge Locations (EL) and Availability Zones (AZ) are AWS Data Centre’s, but EL are primarily used for caching of the data to provide better user experience with low latency, the AZ are used for hosting servers, websites, applications, software’s, Big Data processing, analytics and a wide variety of use cases.

**What is an Availability Zone?**

Each Availability Zone is a one or more discrete data centres with redundant power, networking and connectivity. They are separated from each other so that they are isolated from disasters. They are connected with high bandwidth, ultra-low latency networking.

**Quiz questions I made mistakes**

* When using an Application Load Balancer to distribute traffic to your EC2 instances, the IP address you'll receive requests from will be the ALB's private IP addresses. To get the client's IP address, ALB adds an additional header called

"X-Forwarded-For" contains the client's IP address.

"X-Forwarded-Port" contains the port number

"X-Forwarded-Porto" contains the protocol

* When you enable ELB Health Checks, your ELB won't send traffic to unhealthy (crashed) EC2 instances.

1. Application Load Balancers support HTTP, HTTPS and WebSocket
2. Classic load balancer Supports TCP (Layer 4), HTTP & HTTPS (Layer 7)
3. NETWORK load balancer Supports TCP and UDP traffics.

* Network Load Balancer has one static IP address per AZ and you can attach an Elastic IP address to it. Application Load Balancers and Classic Load Balancers have a static DNS name.

**What is Automated Provisioning?**

Automated provisioning is a key DevOps capability that delivers computing capacity on-demand without manual intervention.

**What is OS Patching?**

Linux Host Patching is a feature in Enterprise Manager Grid Control that helps in keeping the machines in an enterprise updated with security fixes and critical bug fixes, especially in a data centre or a server farm. ... Allow non-compliant packages to be patched.

**AWS DEVELOPER ASSOCIATE** :

Configuing aws CLI in local:

aws configure

give access key id, give secret access key

give region

aws iam list-users

**EC2 Fundamentals**:

1. Create a EC2 instance and download the pem keypair file.
2. Open terminal in the pem file directory.
3. Run commands in EC2 instance connect section.
4. To installl any we have to be root user. So enter sudo -i
5. Jdk 18 installation:

<https://computingforgeeks.com/how-to-install-java-18-on-centos-fedora/>

**Install an executable jar in EC2 instance:**

1. Upload jar to S3 bucket then go to Actions & get pre-assigned url.
2. Now go to EC2 instance as a root user( sudo -i). wget -O name-of-jar “presigned url”
3. java -jar JAR-NAME
4. Add security inbound rule, Type -> Custom TCP, Port range(8080 -> your application running port), source -> 0.0.0.0/0.
5. Take the public URL in the EC2 instance and hit via postman

**To install mysql in EC2 server**

1. yum install -y mariadb-server
2. systemctl enable mariadb
3. systemctl start mariadb
4. mysql\_secure\_installation
5. mysql -uroot -p

**Deploying the React Application:**

1. Create AWS EC2 instance.
2. Install [node](https://docs.aws.amazon.com/sdk-for-javascript/v2/developer-guide/setting-up-node-on-ec2-instance.html) in ec2 instance.
3. Install git (sudo yum install git -y) and use node version 16
4. Clone the project to the instance
5. Run npm install to download the modules.
6. Run npm run build and after run “serve -s build”

**Observation**: After creating an instance from an AMI(which contains both jdk 18 and mysql) and posting some data via rest api(aws\_student\_0.0.1-student.jar) to mysql database, then even is you stop the instance and restart it the data is being saved.

**Elastic Load Balancing and Auto Scaling Groups:**

Load balancers act as front wall which takes the traffic and distributes the traffic to the target groups(target groups are nothing but group of instances).

We talk about Application Load balancers which are mostly used for HTTP/HTTPS.

1. Click on Create a application load balancer(ALB).
2. Give a name to the ALB.
3. Scheme -> Internet-facing and IP address -> IPV4.
4. Mappings -> select all AG’s
5. Select/create a security group which has inbound rules CUSTOM TCP, port range(8080 -> your application running port), source IPV4
6. Listerns -> HTTP – 8080, Select a target group.
7. Create a target group -> Choose a traget type as an instance.
8. Give a name to the target group
9. Protocol HTTP 8080 and Next
10. Register target instances -> click all the instances that apply(attach the security group to the instances with inbound CUTSOM TCP -> 8080, source IPV4 OR if you don’t want anyone to direct access the instance but access via onlt ALB then update source IPV4 to security which is attached to your load balancer).
11. Ports for the selected instances -> 8080
12. Click on Include as pending below.
13. And click on create a target group.
14. Finally select the target group in ALB page
15. Click on Create load balancer.

**ROUTE 53**

**What is DNS?**

A Domain name system which translates the human friendly hostnames into the machine IP addresses.

**Record Types:**

A -> Maps a ec2 insatnce IPV4 to a domain name. Ex: “test.thevishaldev.com”

AAAA -> Maps a ec2 insatnce IPV6 to a domain name. Ex: “test.thevishaldev.com”

CNAME -> Maps a hostname(load balancer) to another hostname(test.thevishaldev.com)

ALIAS -> Alias is used to point hostname to a AWS Resource. To use ALIAS we should select RecordType as “A”, click alias on, choose the aws resource, region.

Note: The important feature in ALIAS WE CAN point aws resource to the root hostname(“thevishaldev.com”)

**Routing Policy:**

**Simple**: Typically route traffice to a single resource. Its typically route traffic to a IPV4 or DNA name we provide at initial stage. If you provide multiple IPV4 addresses then client will pick random address and use one address.

**Weighted**: We can set how much percentage of time our DNS server should point to a instances. Instance 1 (70%) and Instance 2 (30%).

**Latency**: If one user is India and another user is in America. And one server are deployed in India and another server is deployed in america. User 1 who is in India will be redirected to India Server and user 2 who is in america will be redirected to America server. So users feels low latency.

**Geolocation**: Germany people should route to instance that deployed in germany which gives german version of application and india people should route to a instance deployed in india which gives india’s version of application.

**VPC**

**VPC**: Virtual private cloud is a private network to deploy your resources (regional resource).

**Subnets**: Subnets allow you to partition your network inside your VPC (Availability Zone resource)

A public subnet is a subnet that is accessible from the internet.

A private subnet is a subnet that is not accessible from the internet.

To define access to the internet and between subnets, we use Route Tables.

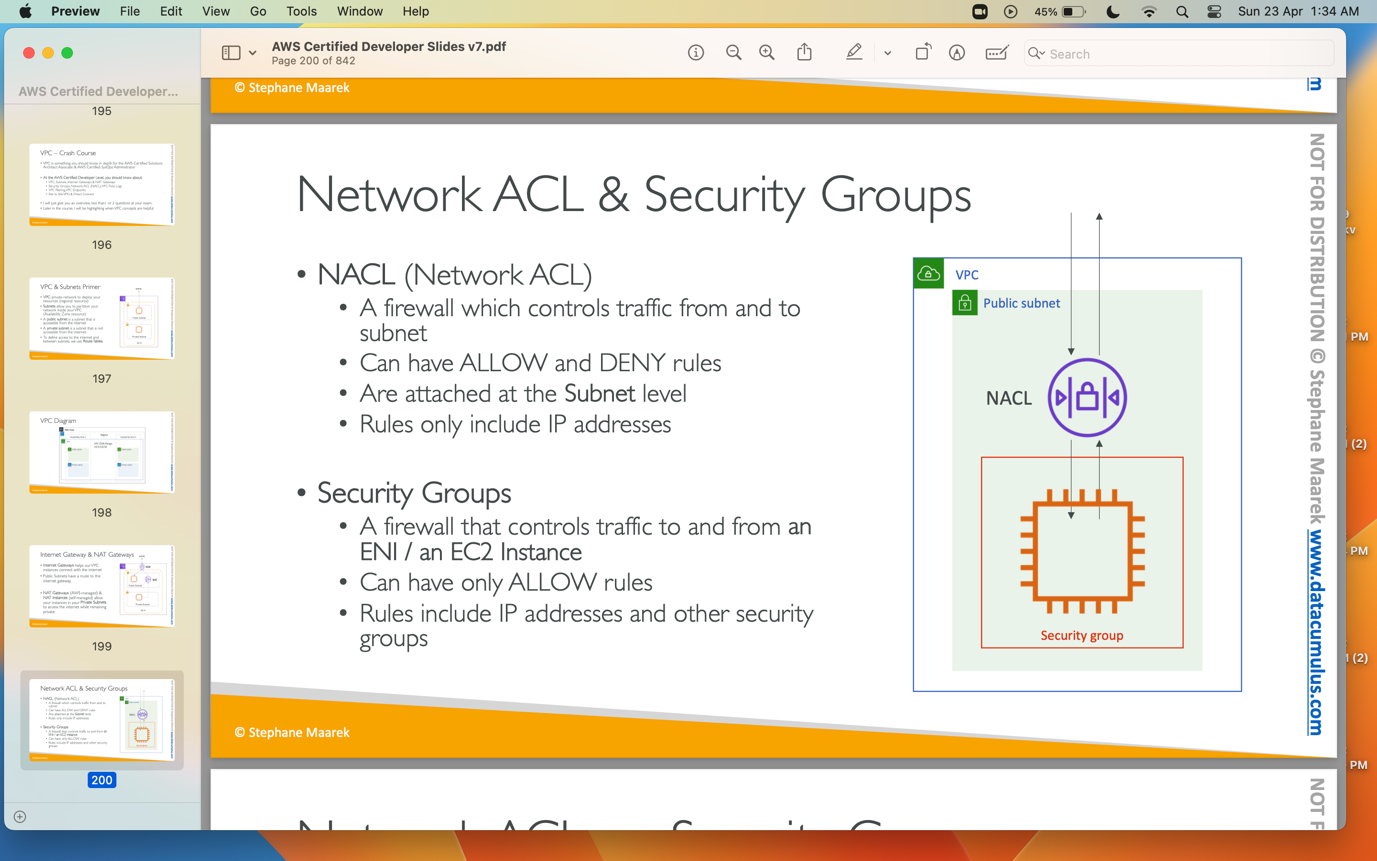
**Internet Gateway and NAT Gateways**

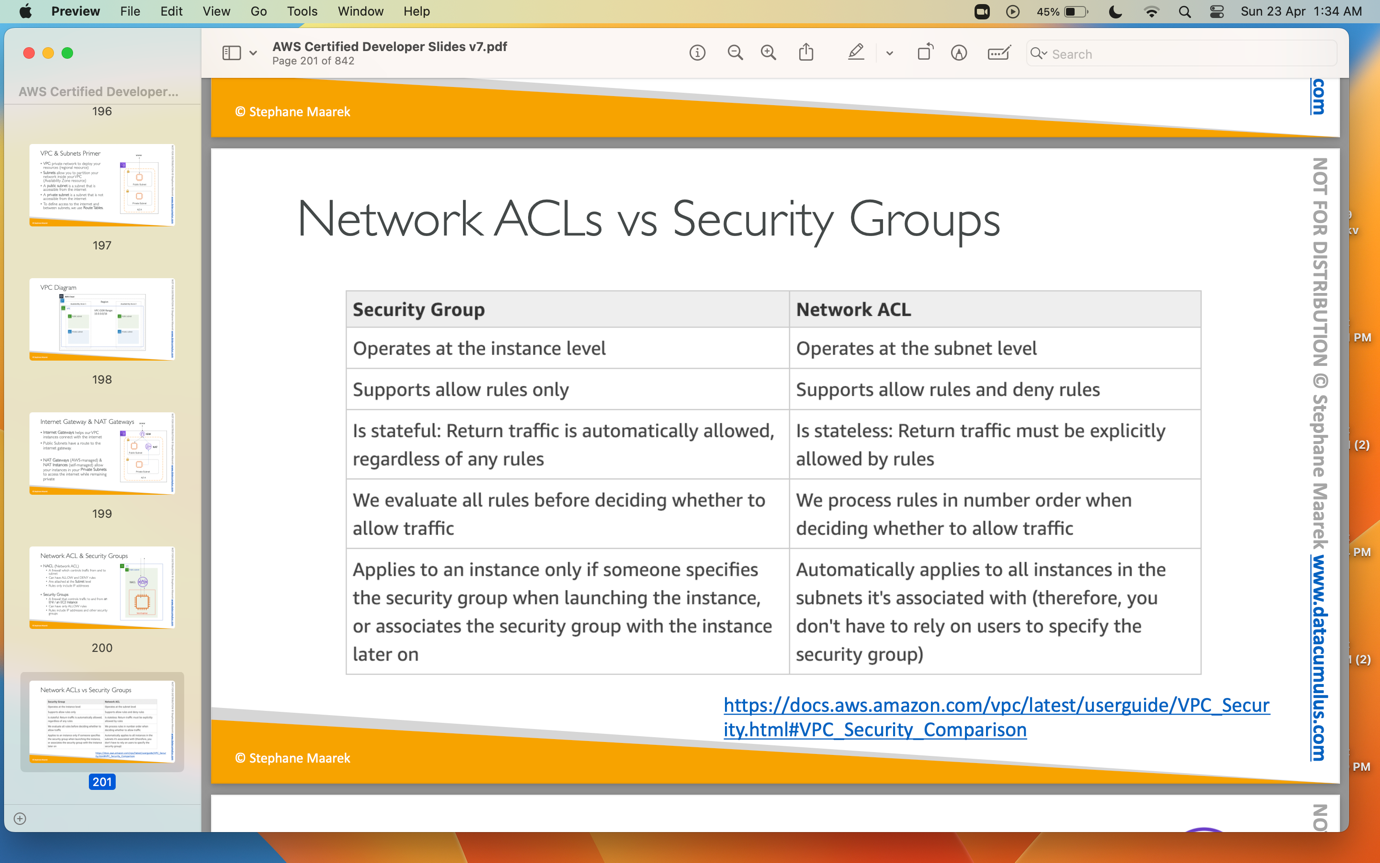
Internet gateway helps our VPC instances connect with the internet, public subnets have a route to the internet gateway.

NAT Gateways which is AWS managed allows the instances in the private subnets to connect with internet. NAT gateway routes via internet gateway to allow access to the private subnet instances.

**What is Network ACL and Security groups**

When we create a ec2 instance, in a default vpc, a default Network ACL is there, we never configured it, so it took the default configuration.





**VPC Flow Logs**: It is used to capture information about IP traffice going into your interfaces.

**VPC Peering**: Connect two VPC, privatly using AWS’s network.

**VPC Endpoint**: VPC endpoints allow instances in private subnets to connect aws services.

**VPC Endpoint Gateway**: S3 & Dynamo DB.

**VPC Endpoint Interface**: the rest AWS services.

Site to Site VPN: Connect on permsises datacenter to VPC

Direct Connect: Physical connect between on premises datacenter to VPC.

**Amazon S3**

Amazon S3 is very important service in AWS. It Is widely used service.

We have to create a bucket in S3 in order to use it. Bucket name should be unique in all aws service account level.

We upload any file into the s3 bucket, it is treated as object.

The key is the FULL path:

• s3://my-bucket/my\_file.txt

• s3://my-bucket/my\_folder1/another\_folder/my\_file.txt

The key is composed of prefix + object name

• s3://my-bucket/my\_folder1/another\_folder/my\_file.txt

Making an S3 BUCKET public so we can access all files/objects inside the S3 bucket:

1. Click on the permisions tab in s3 bucket and edit the block public access and uncheck the block public access checkbox.
2. Create a S3 bucket Policy using policy generator, policy should be like below.

{

"Version": "2012-10-17",

"Id": "Policy1682247897459",

"Statement": [

{

"Sid": "Stmt1682247894513",

"Effect": "Allow",

"Principal": "\*",

"Action": "s3:GetObject",

"Resource": "arn:aws:s3:::firstbucket392/\*"

}

]

}

**S3 Bucket versioning:**

By Default S3 bucket is not enabled. Go to the properties tab and edit the bucket versioning to enable. Then try to delete a file. After deleting a file come to your bucket and toggle the show versions. You will see that your deleted file not deleted, it is still in the bucket and a versioning is added to that. Now delete the versioned file, your file which was deleted will be restored.

AWS CLI DRY RUN

We can use --dry-run command to test the command line api’s wheather we have permissions or not. If you wanna check wheather you can start an ec2 instance via command line but you don’t wanna really start ec2 instances, in that case we can add –dry-run in your command which prevent them to really start the ec2 instances.

AWS CLI STS DECODE

To debug an error message we can use this command in cli “aws decode-authorization-message –encoded-message <error value>”. This will give the detailed explaination of error message.

AWS SDK

If you want to perform actions on AWS directly from your applications code we need aws sdk. SDK stands for software development kit.

**AWS ECS**

Amazon Elastic Container Service (ECS) is a fully-managed container orchestration service that helps you run, deploy, and scale containerized applications on AWS. An ECS cluster is a logical grouping of Amazon EC2 instances or AWS Fargate tasks that run containerized applications.

**Task**: A set of one or more containers that run together on a single container instance.

**Task Definition**: A blueprint for a task that specifies the Docker image, CPU and memory requirements, networking, storage, and other configuration settings for each container in the task.

**Container Instance**: An Amazon EC2 instance that has the ECS agent installed and is registered to an ECS cluster. Container instances can host one or more containers.

**Cluster**: A logical grouping of container instances that can be managed together as a single unit.

**Task Placement:** The process of selecting the best container instance to run a task on, based on factors such as resource requirements, task definitions, and available capacity.

**ECS Agent:** A daemon that runs on each container instance and communicates with the ECS service to manage tasks and report status.

**Fargate:** A serverless launch type for ECS that allows you to run containers without managing the underlying infrastructure.

**Service**: A long-running task that ensures that a specified number of tasks are running and automatically replaces any that fail or become unhealthy.

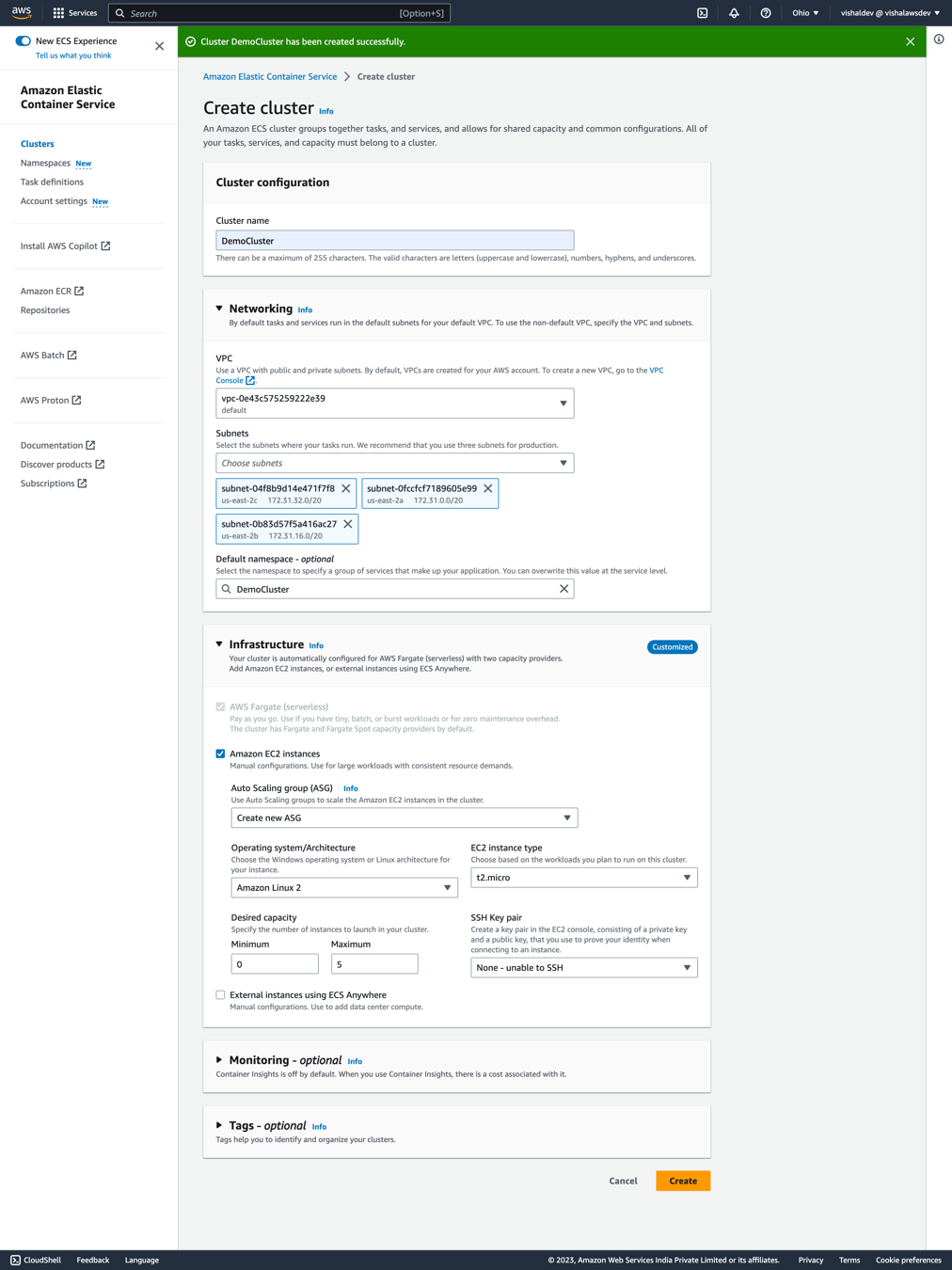
Launch Docker containers on AWS = Launch ECS Tasks on ECS Clusters.

**Difference between EC2 Launch type and Fragate Launch type**

If you need more control over the underlying infrastructure, then EC2 launch type may be a better choice. However, if you prefer a simpler and more automated approach, then Fargate launch type may be a better fit.

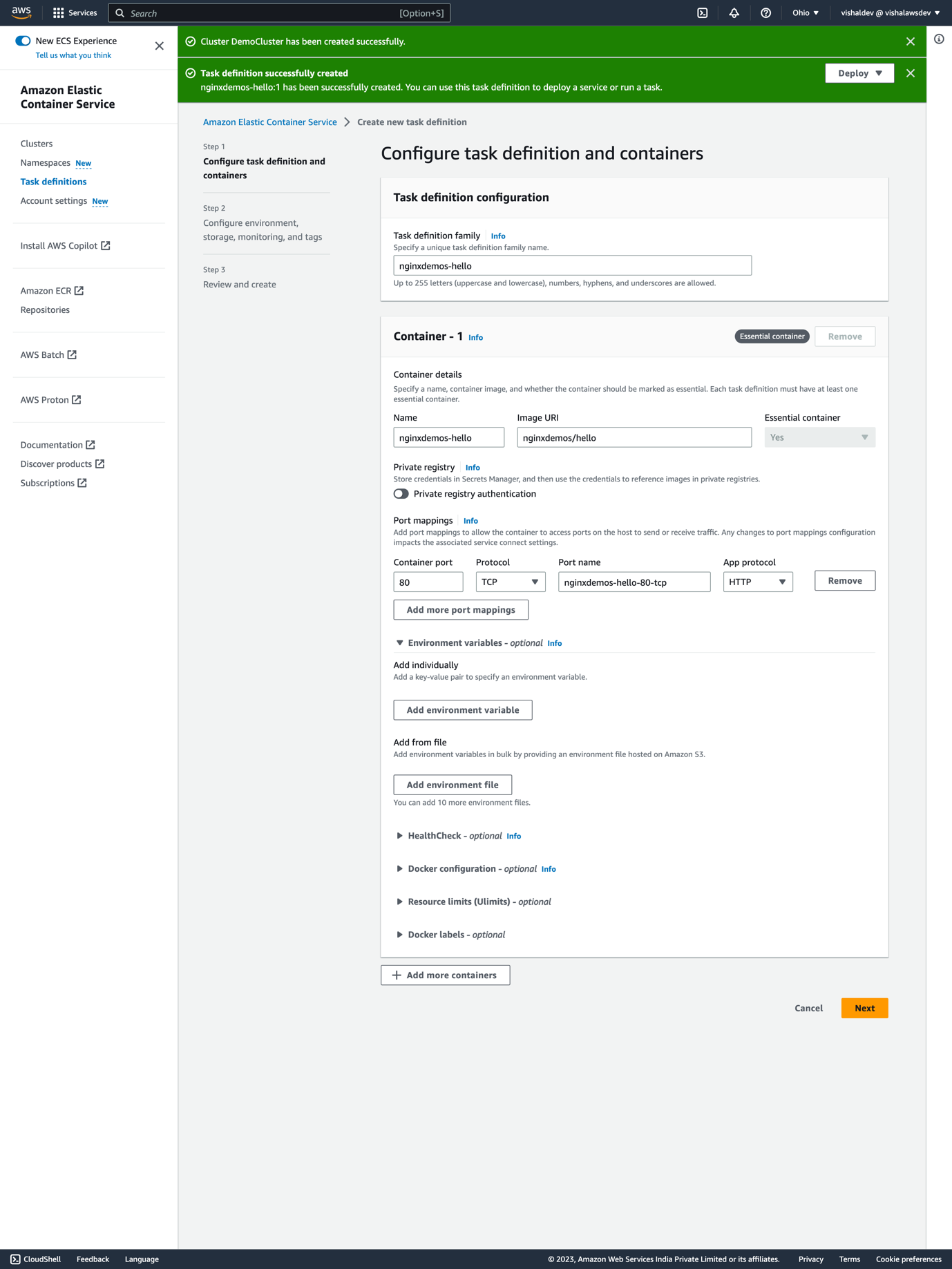
Creating a ECS Cluster:

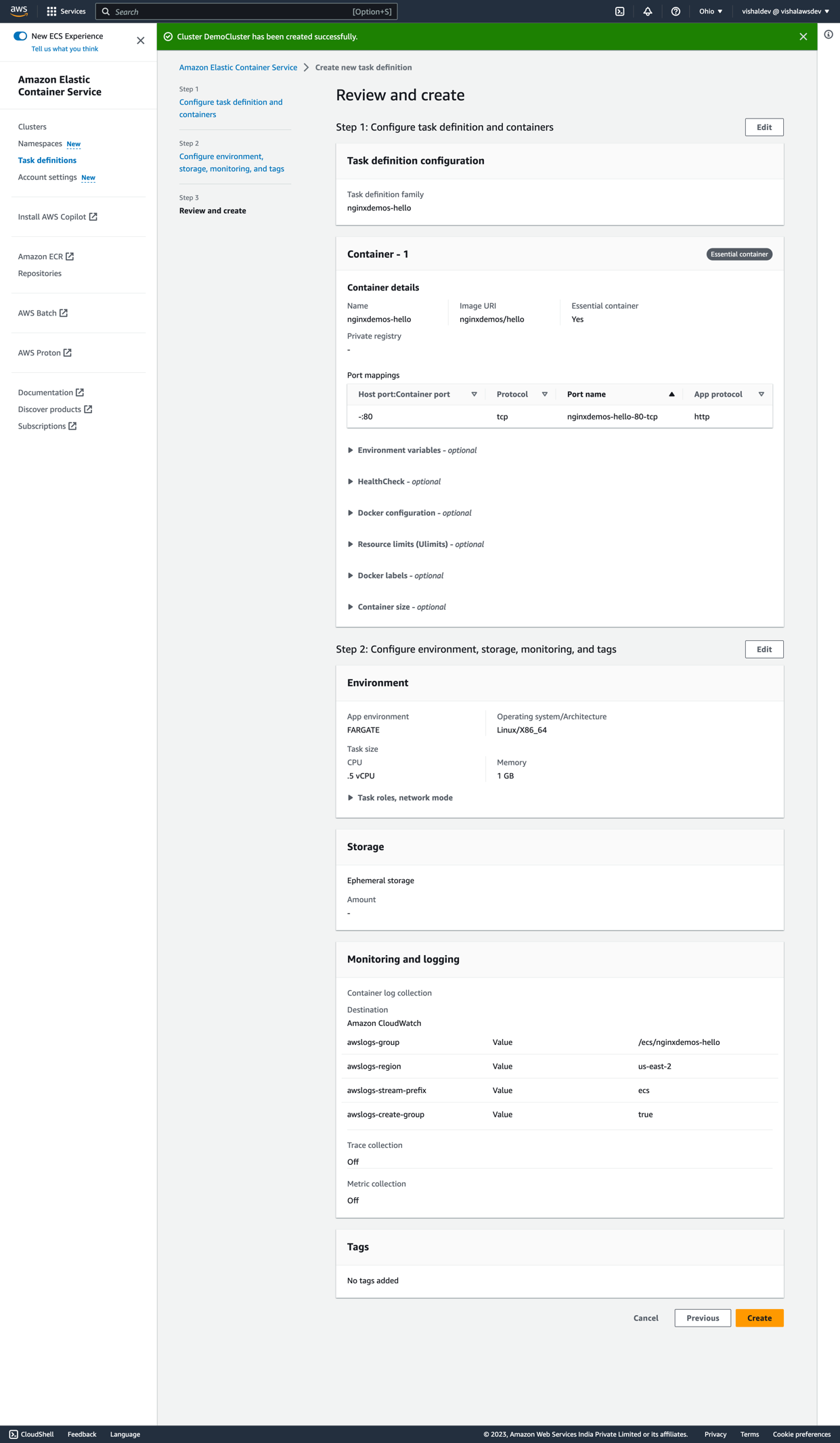
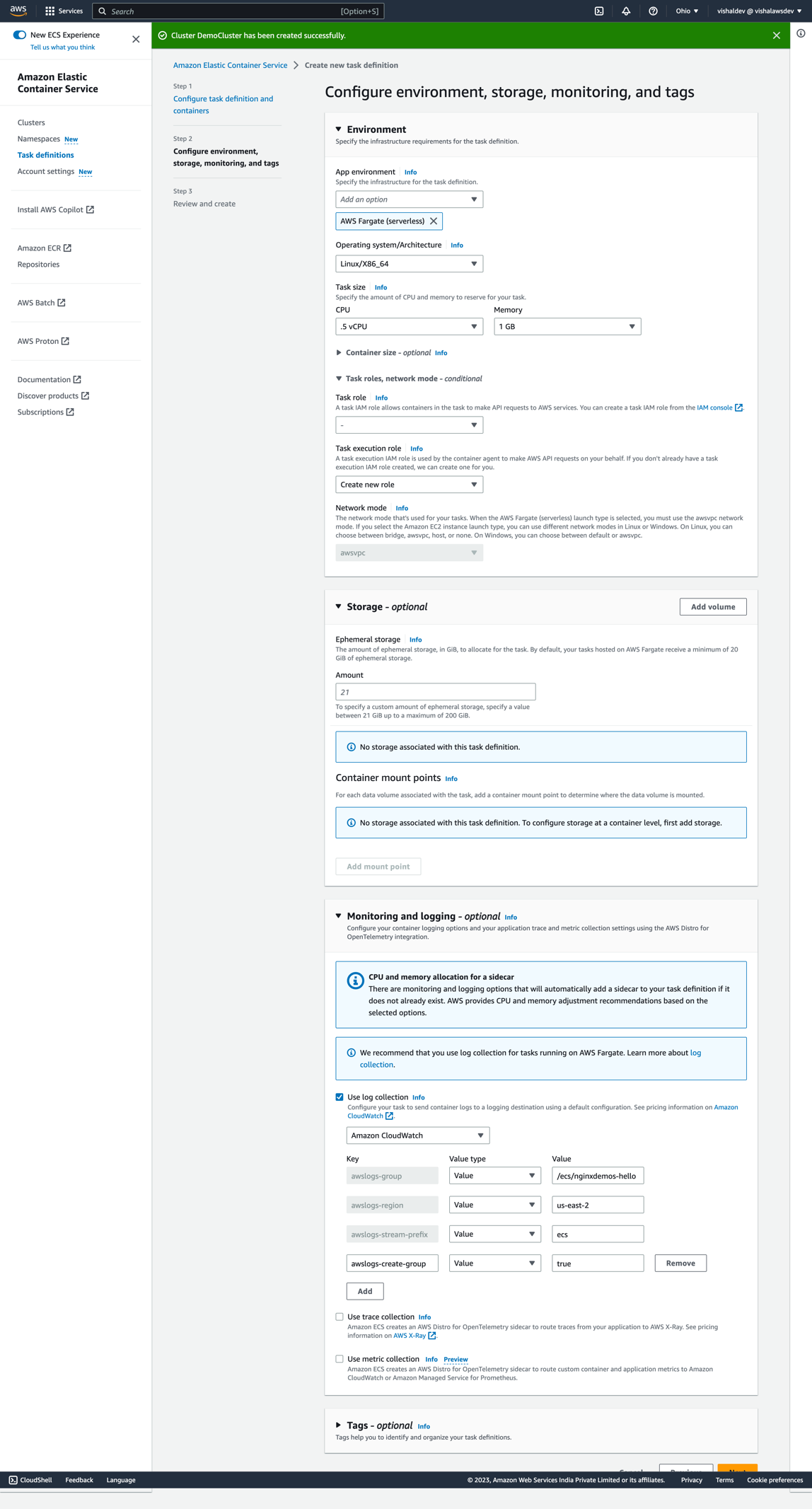
1. Navigate to AWS ECS page.
2. Click on cluster and click on create cluster.



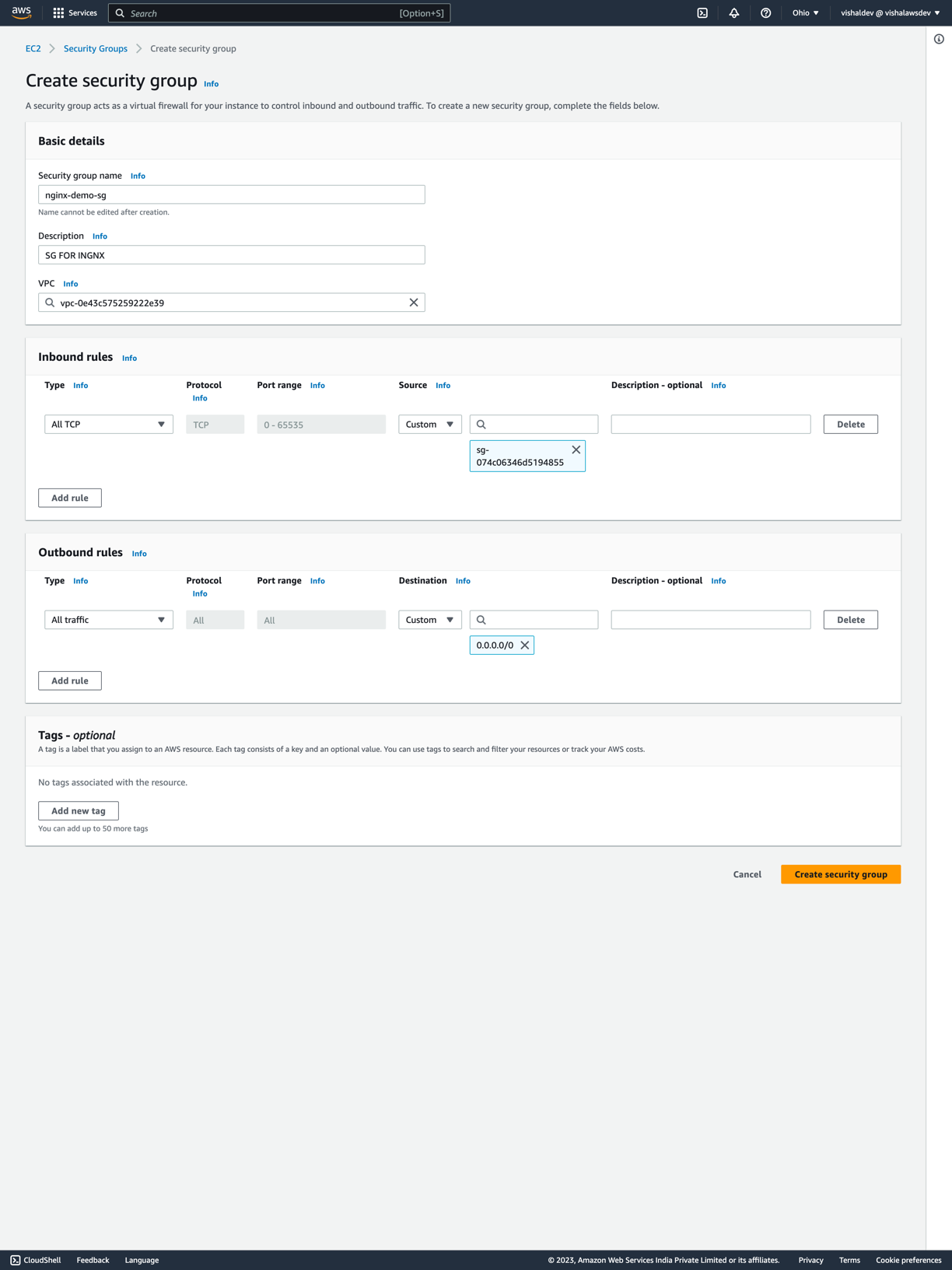
As we selected Amaxon EC2 instances, a new Auto scaling group will be created for you.

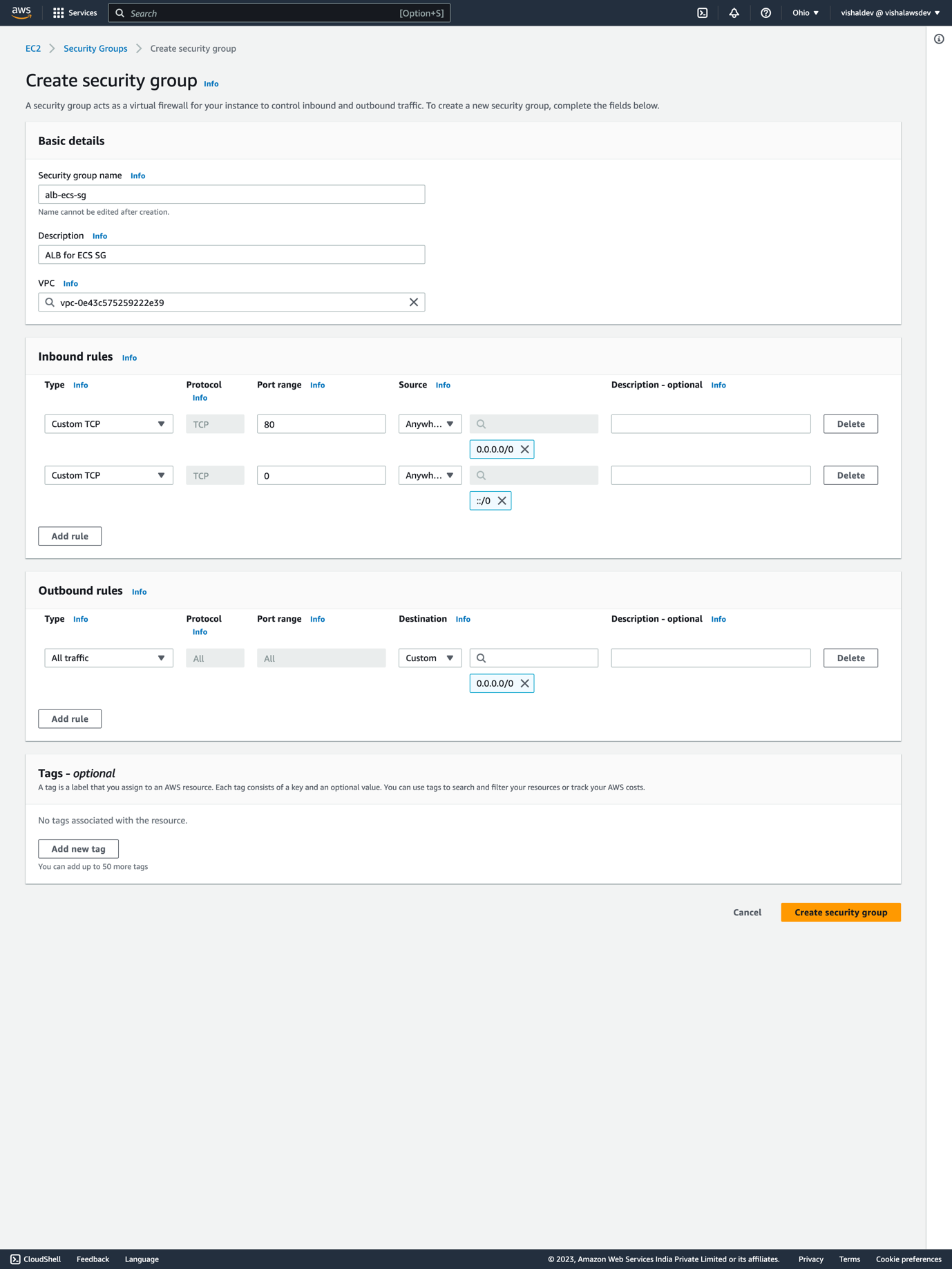
Creating a task definition



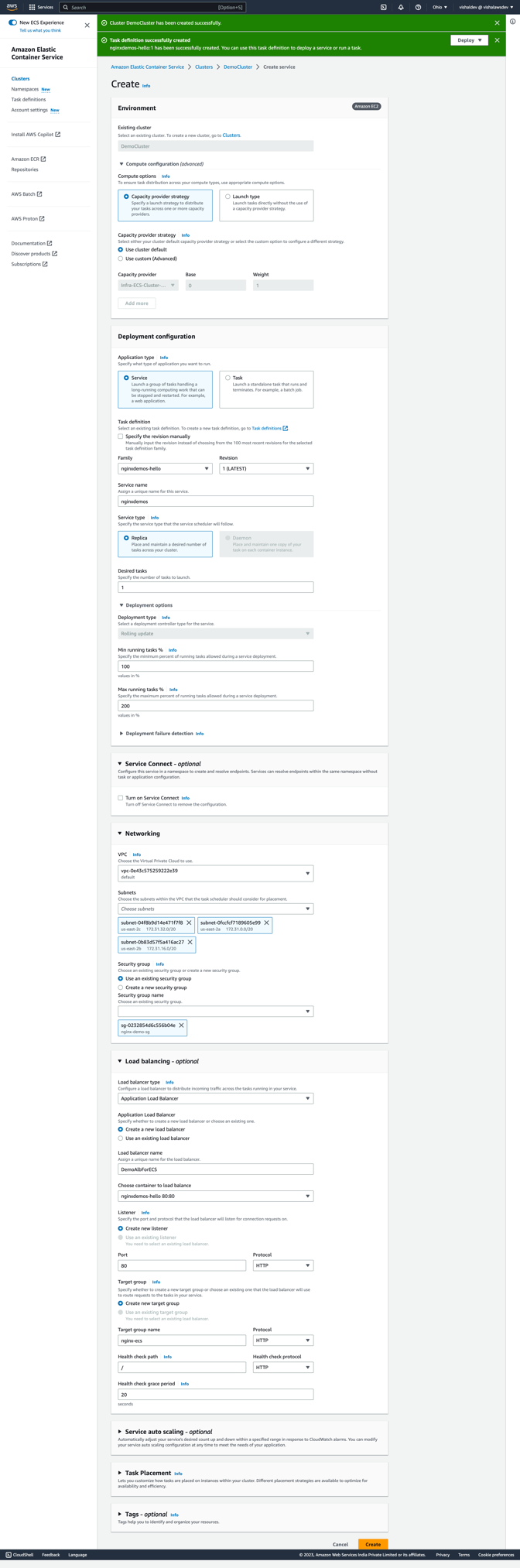


Creating a two security groups





Creating a service



When you create the service, under the Compute Configuration section the Capacity Provider option is automatically selected to be the EC2 instance we created before. However, in this demo Stephane is using Fargate as the capacity provider. Therefore, you need to change Capacity Provider Strategy to be Use custom (Advanced) instead of Use cluster default. This then allows you to select Fargate as the capacity provider. I think he missed this step out of the video.

**Note**: in the above picture we selected wrongly the nginx-demo-sg, change to alb-demo-sg, we cant do that by going into the alb and edit the inbound rules.

After creating the service, go to the alb and copy the DNS and try hitting the url it should work.



**What is regions and Availability Zones?**

us-east-1 -> reagion

us-east-1a , us-east-2a, us-east-3a -> Availability zones.

**Aws Elastic Bean Stalk**