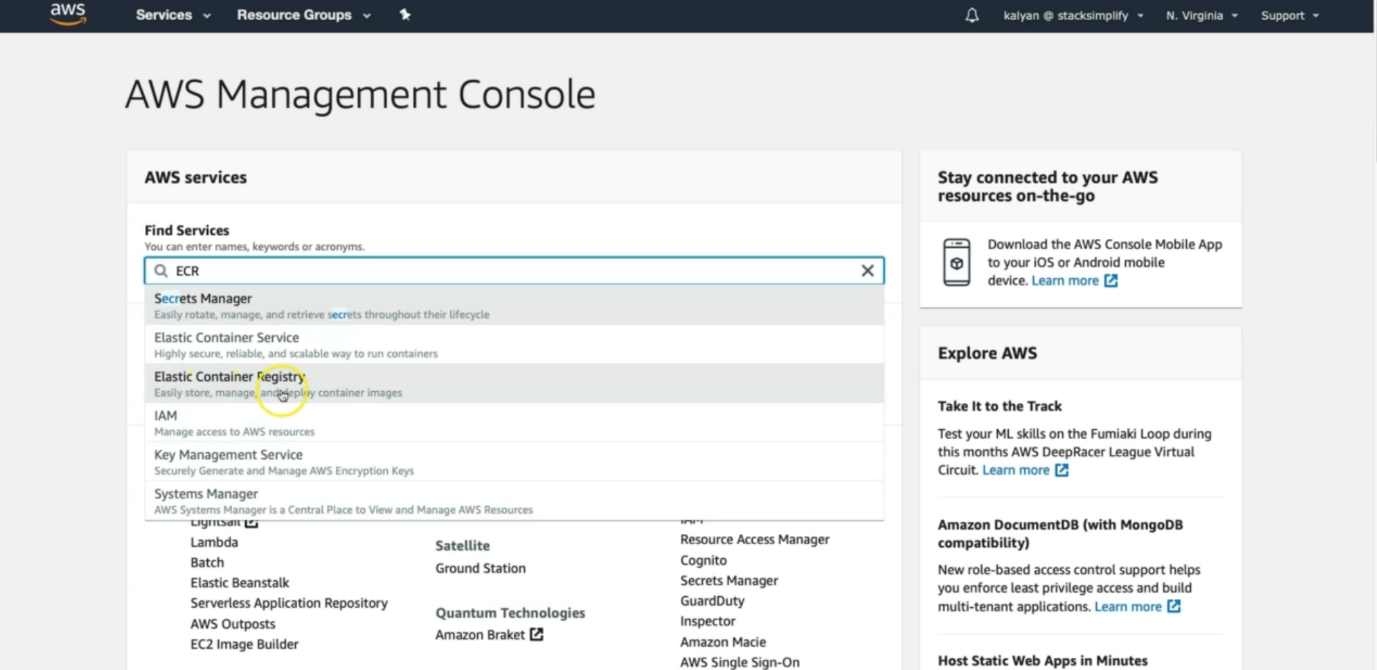
**03. Create ECR Repository on AWS, Build Docker Image Locally & Push to ECR**

--- <https://github.com/stacksimplify/aws-eks-kubernetes-masterclass/tree/master/10-ECR-Elastic-Container-Registry-and-EKS>

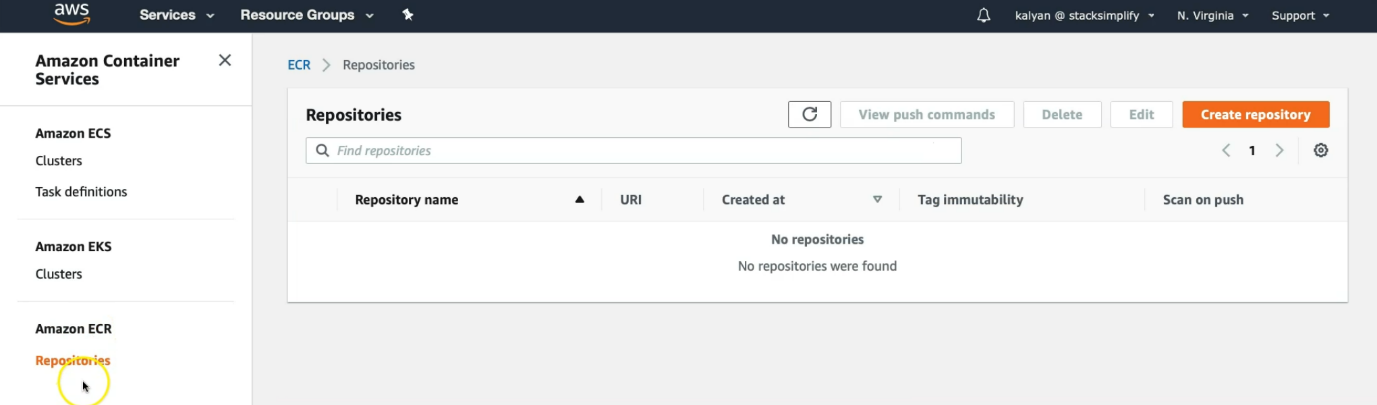
**Create ECR Repository**

--- Create simple ECR repository via AWS Console

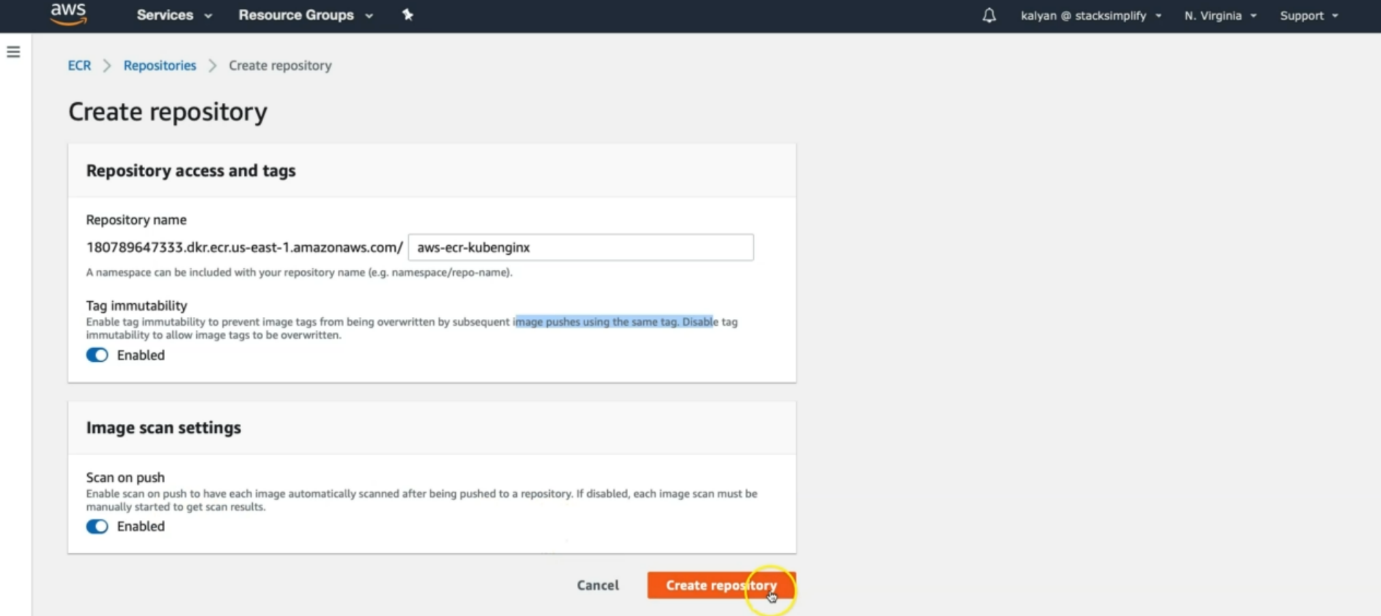
1. Repository Name: aws-ecr-kubenginx
2. Tag Immutability: Enable
3. Scan on Push: Enable



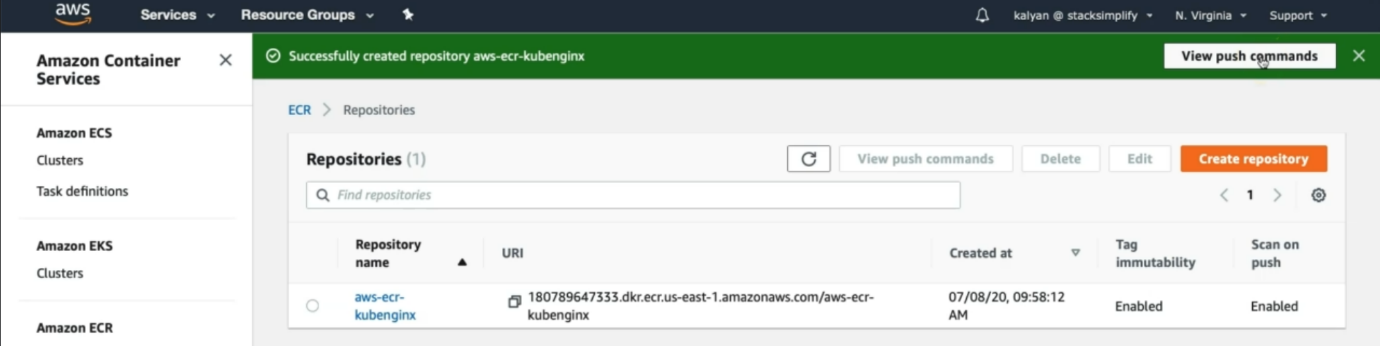
--- click on Elastic container registry.



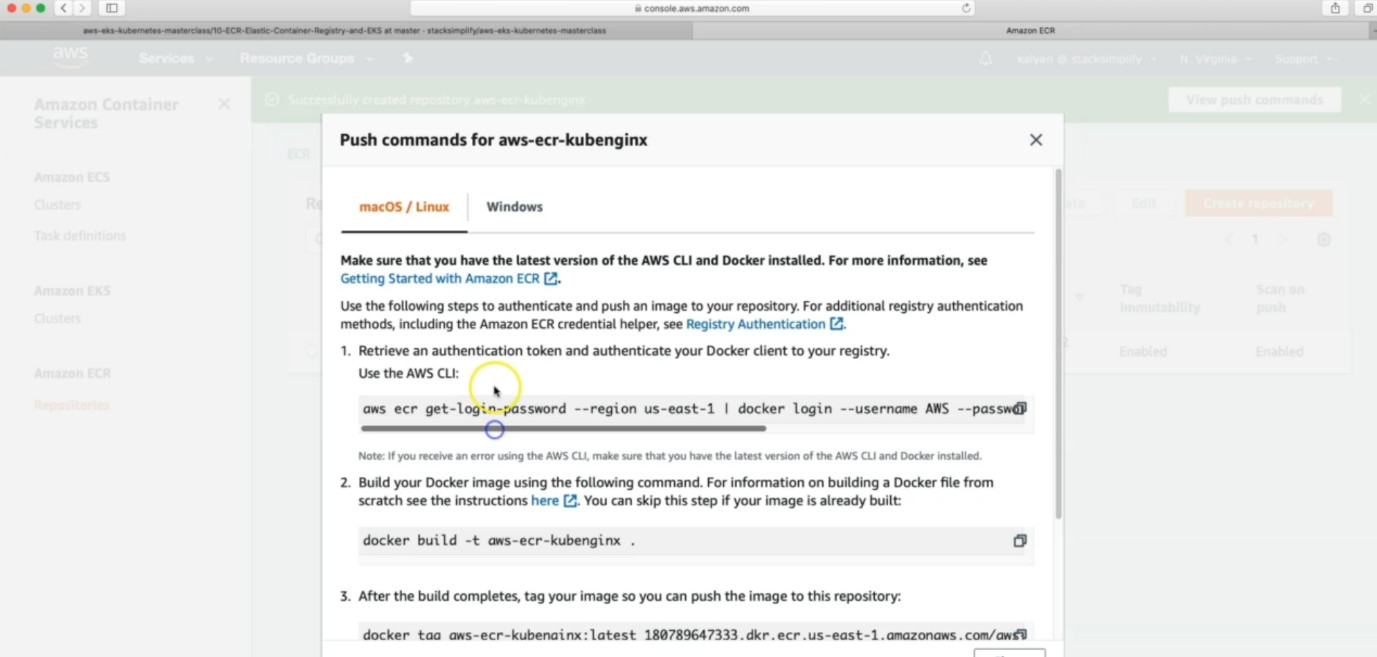
--- click on create repository.



--- repository name is the repository url.



--- our repository is created. If you want then you can see the push commands.



--- these are the commands; we can use them to push images to ECR.

--- Explore ECR console.

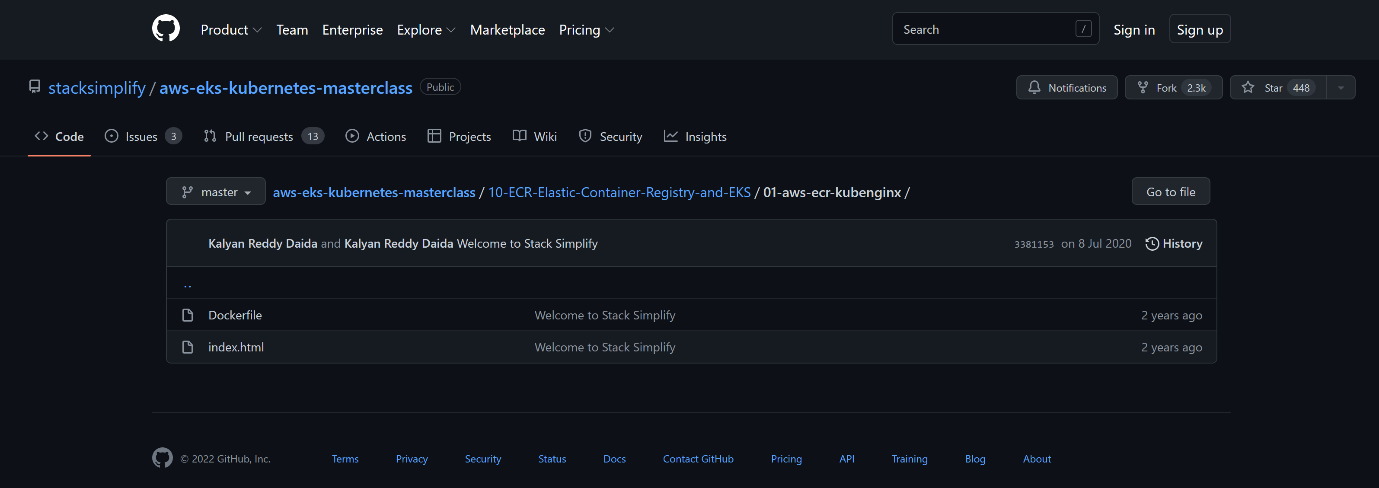
**#Create ECR Repository using AWS CLI**

--- **aws ecr create-repository --repository-name aws-ecr-kubenginx --region us-east-1**

--- **aws ecr create-repository --repository-name <your-repo-name> --region <your-region>**

**Create Docker Image locally**

--- Navigate to folder **10-ECR-Elastic-Container-Registry\01-aws-ecr-kubenginx** from course github content download.



--- under the above folder, you will find these 2 folders.

--- **Dockerfile**

FROM nginx

COPY index.html /usr/share/nginx/html

--- **index.html**

<!DOCTYPE html>

<html>

<body style="background-color:rgb(217, 250, 210);">

<h1>Welcome to Stack Simplify</h1>

<h3>AWS EKS Master Class - Integration with ECR Registry</h3>

</body>

</html>

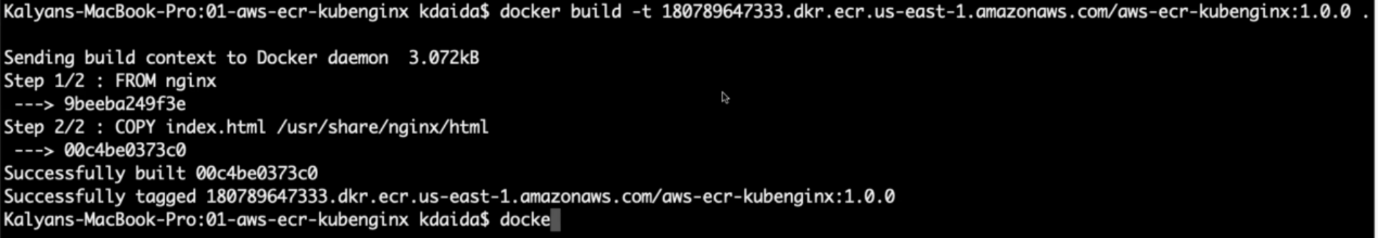
--- Create docker image locally

--- Run it locally and test

**# Build Docker Image**

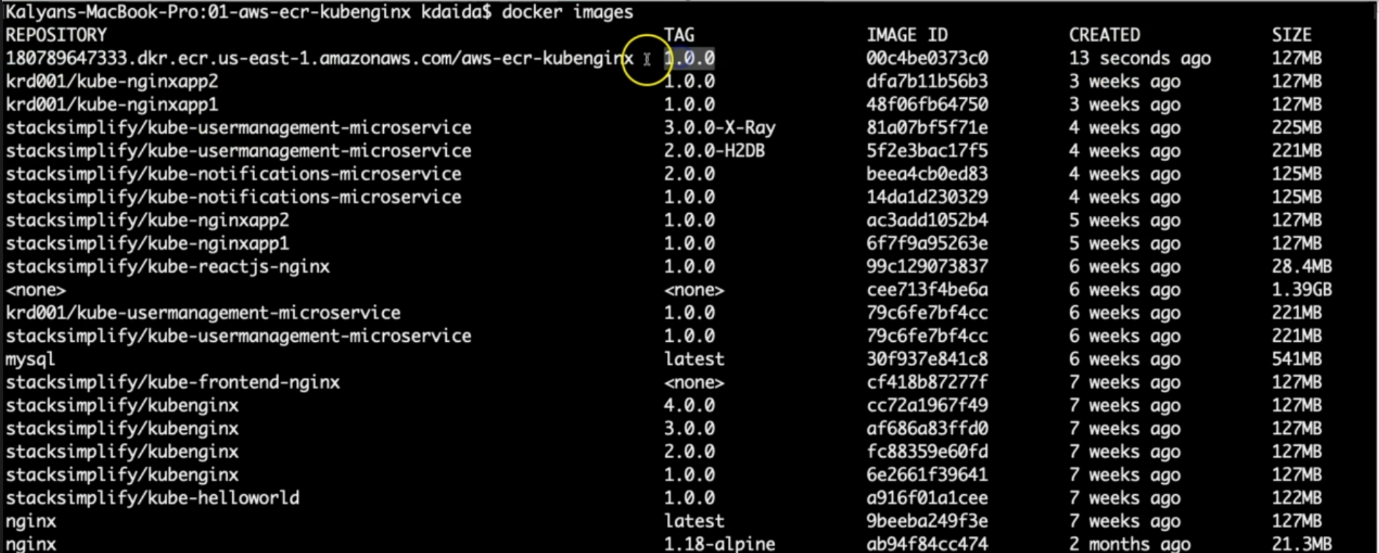
--- **docker build -t <ECR-REPOSITORY-URI>:<TAG> .**

--- **docker build -t 180789647333.dkr.ecr.us-east-1.amazonaws.com/aws-ecr-kubenginx:1.0.0 .**



--- successfully built the image.

--- **docker images**

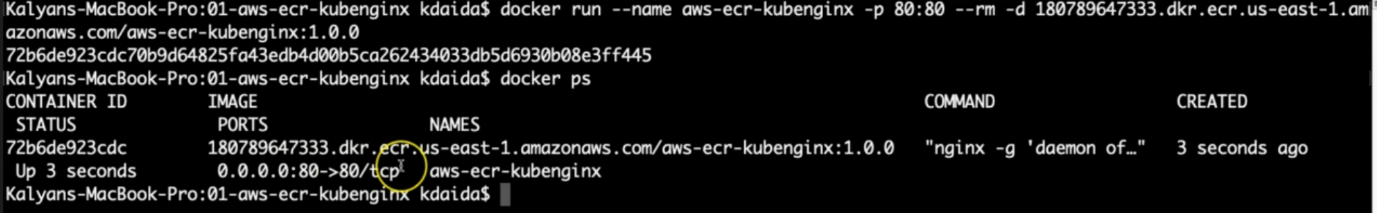


--- the 1st image is the one, I created.

**# Run Docker Image locally & Test**

--- **docker run --name <name-of-container> -p 80:80 --rm -d <ECR-REPOSITORY-URI>:<TAG>**

--- **docker run --name aws-ecr-kubenginx -p 80:80 --rm -d 180789647333.dkr.ecr.us-east-1.amazonaws.com/aws-ecr-kubenginx:1.0.0**



**# Access Application locally**

--- **http://localhost**

**# Stop Docker Container**

--- **docker ps**

--- **docker stop aws-ecr-kubenginx**

--- **docker ps -a -q**

**Push Docker Image to AWS ECR**

--- Firstly, login to ECR Repository

--- Push the docker image to ECR

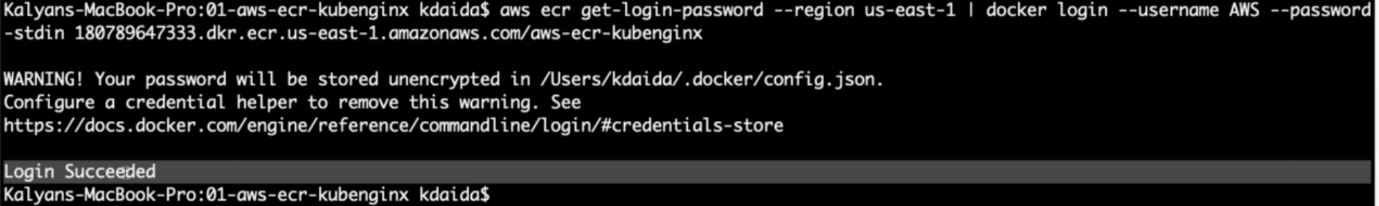
--- AWS CLI Version 2.x

**# Get Login Password**

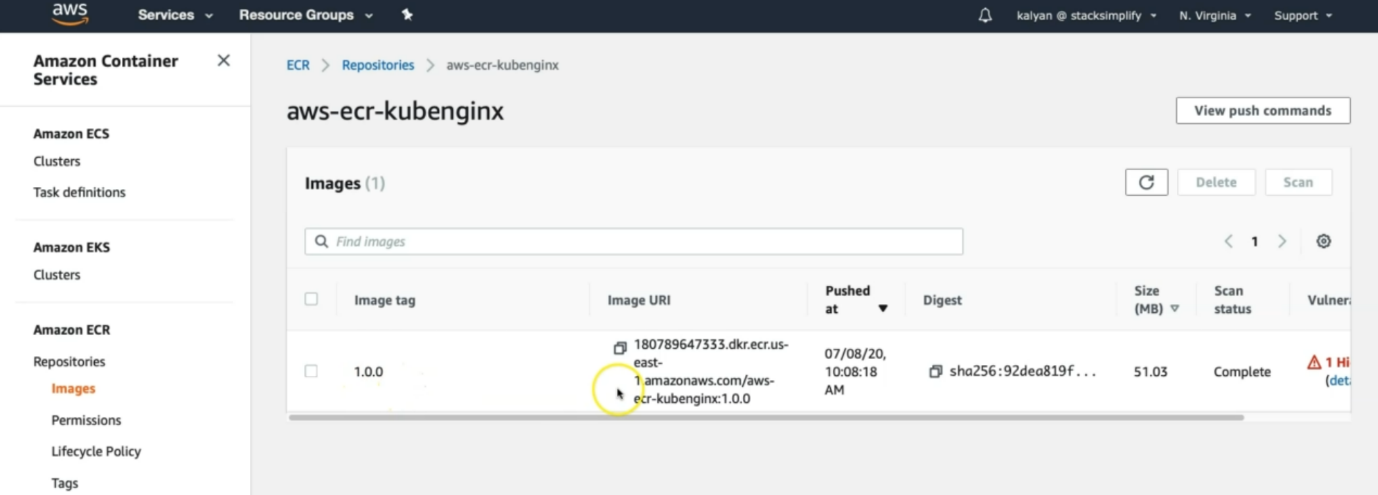
--- **aws ecr get-login-password --region <your-region> | docker login --username AWS --password-stdin <ECR-REPOSITORY-URI>**

--- **aws ecr get-login-password --region us-east-1 | docker login --username AWS --password-stdin 180789647333.dkr.ecr.us-east-1.amazonaws.com/aws-ecr-kubenginx**

--- **note** – take the password and pipe it to the next command.

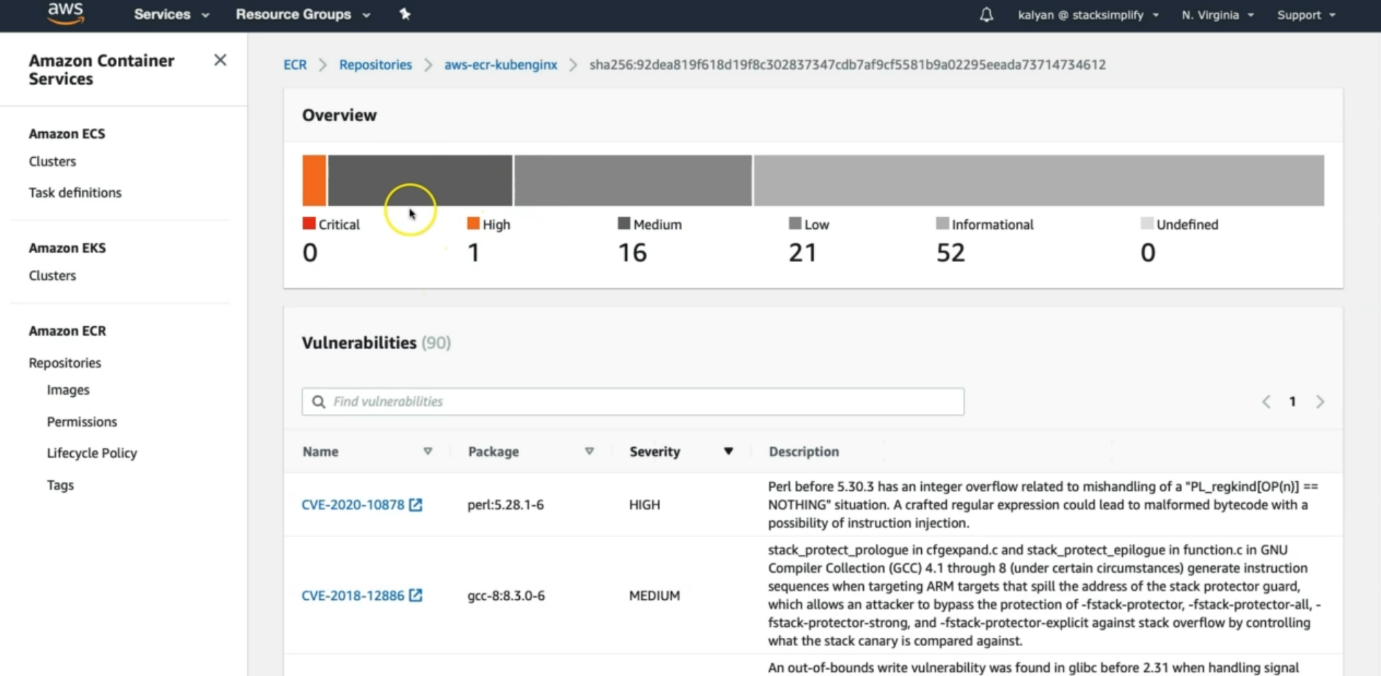


--- our password is saved in above mentioned path and we successfully login.



--- our image is pushed to aws ECR.

--- click on vulnerabilities.



--- these are the scanning results. During the push we opted for this option.

**# Push the Docker Image**

--- **docker push <ECR-REPOSITORY-URI>:<TAG>**

--- **docker push 180789647333.dkr.ecr.us-east-1.amazonaws.com/aws-ecr-kubenginx:1.0.0**

--- Verify the newly pushed docker image on AWS ECR.

--- Verify the vulnerability scan results.