Ganpat University | Institute of Computer Technology

॥ विद्यया समाजोत्कर्षः ॥

Name: Tushar Panchal

En.No: 21162101014

Sub: CS(Cloud Security)

Branch: CBA

Batch:71

----------PRACTICAL 06----------

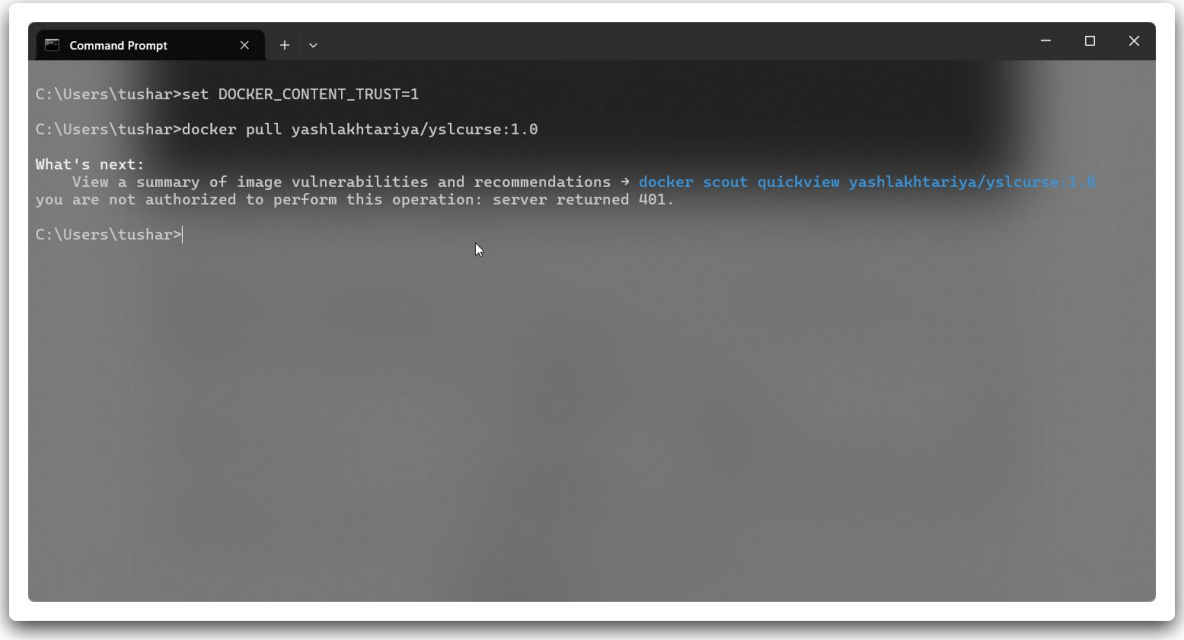
Your organization is developing a Kubernetes application that needs to comply with strict security regulations. One of the requirements is to ensure that only verified, signed container images from your organization's private container registry are deployed in the Kubernetes cluster. To enforce this, you decide to implement Kubernetes image policies to control which container images can be used within the cluster. Scenario: You are tasked with implementing a solution to meet the following security requirements:

1. Allow only signed images from your private registry (registry.example.com).

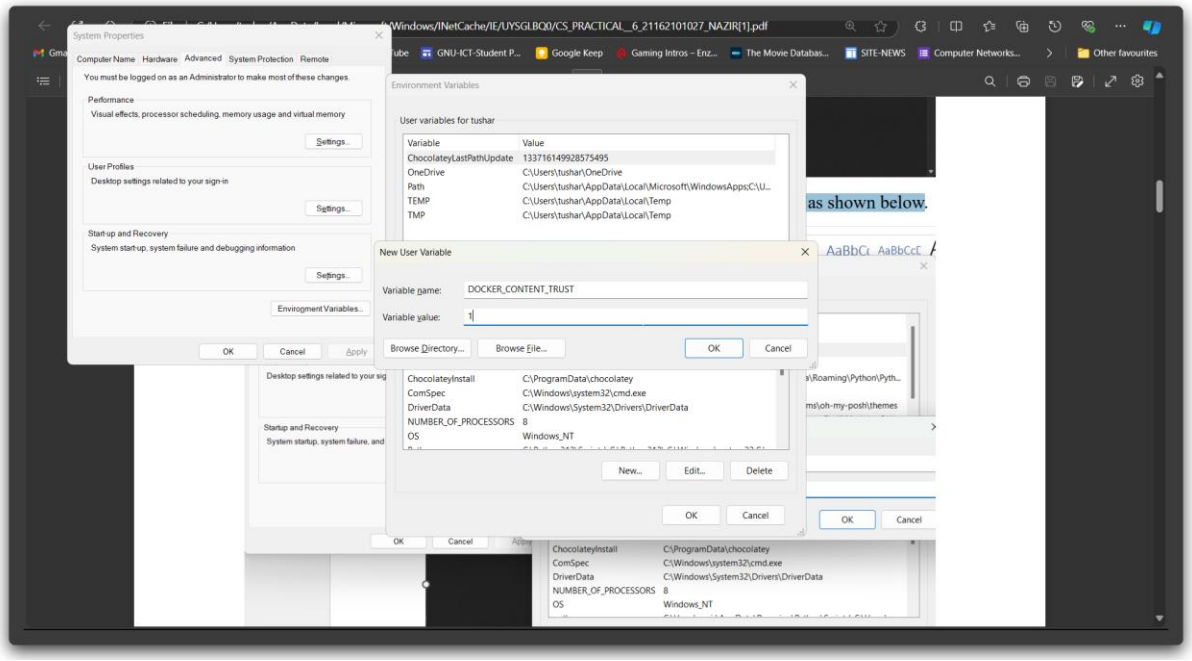
2. Block any unsigned or unknown images from being pulled into the cluster.

3. Ensure that only images from specific trusted repositories (e.g., registry.example.com/trusted-apps/*) are permitted to run.

Try locally, pulling any docker image unsigned and untrusted



If your image is pulled after running this command then set the env variable as shown below

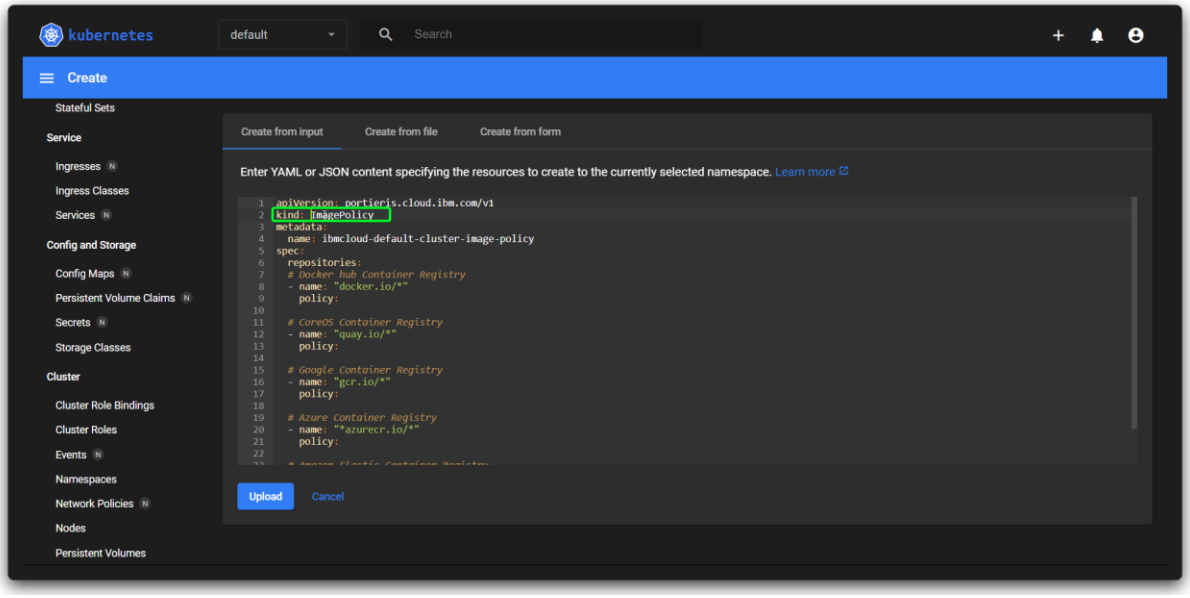


Here you can set the variable as shown above

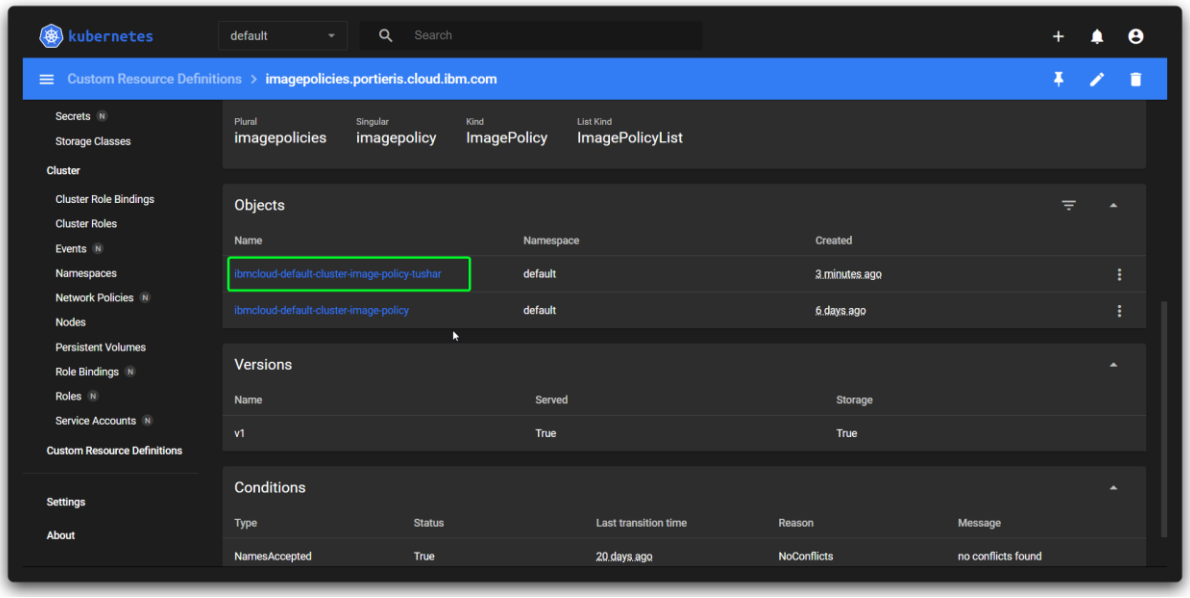
```
Command Prompt
Microsoft Windows [Version 10.0.22631.4169]
(c) Microsoft Corporation. All rights reserved.

C:\Users\tushar>set DOCKER_CONTENT_TRUST=1

C:\Users\tushar>docker pull yashlakhtariya/yslcurse:1.0

What's next:
    View a summary of image vulnerabilities and recommendations → docker scout quickview yashlakhtariya/yslcurse:1.0
you are not authorized to perform this operation: server returned 401.

C:\Users\tushar>docker pull amazon/aws-for-fluent-bit
Using default tag: latest

What's next:
    View a summary of image vulnerabilities and recommendations → docker scout quickview amazon/aws-for-fluent-bit
Error: remote trust data does not exist for docker.io/amazon/aws-for-fluent-bit: notary.docker.io does not have trust data for docker.io/amazon/aws-for-flue
nt-bit

C:\Users\tushar>docker pull busybox
Using default tag: latest
Pull (1 of 1): busybox:latest@sha256:c230832bd3b0be59a6c47ed64294f9ce71e91b327957920b6929a0caa8353140
docker.io/library/busybox@sha256:c230832bd3b0be59a6c47ed64294f9ce71e91b327957920b6929a0caa8353140: Pulling from library/busybox
2fce1e0cdfc5: Pull complete
Digest: sha256:c230832bd3b0be59a6c47ed64294f9ce71e91b327957920b6929a0caa8353140
Status: Downloaded newer image for busybox@sha256:c230832bd3b0be59a6c47ed64294f9ce71e91b327957920b6929a0caa8353140
Tagging busybox@sha256:c230832bd3b0be59a6c47ed64294f9ce71e91b327957920b6929a0caa8353140 as busybox:latest
docker.io/library/busybox:latest

What's next:
    View a summary of image vulnerabilities and recommendations → docker scout quickview busybox

C:\Users\tushar>
```

Now you can see the image is not pulling only the trusted image is pulling.

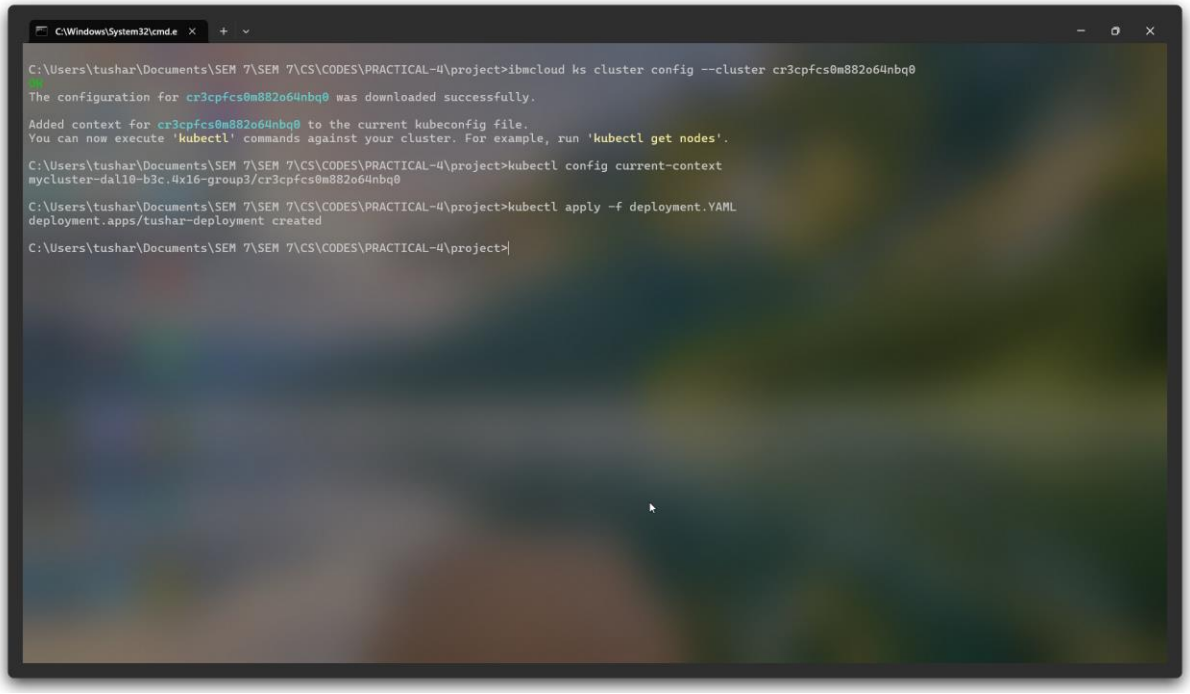
```yaml
spec:
  repositories:
  - name: "docker.io/*"

  # CoreOS Container Registry
  - name: "quay.io/*"
    policy:

  # Google Container Registry
  - name: "gcr.io/*"
    policy:

  # Azure Container Registry
  - name: "*azurecr.io/*"
    policy:

  # Amazon Elastic Container Registry
  - name: "*amazonaws.com/*"
    policy:
```



Open Cluster Image Policy

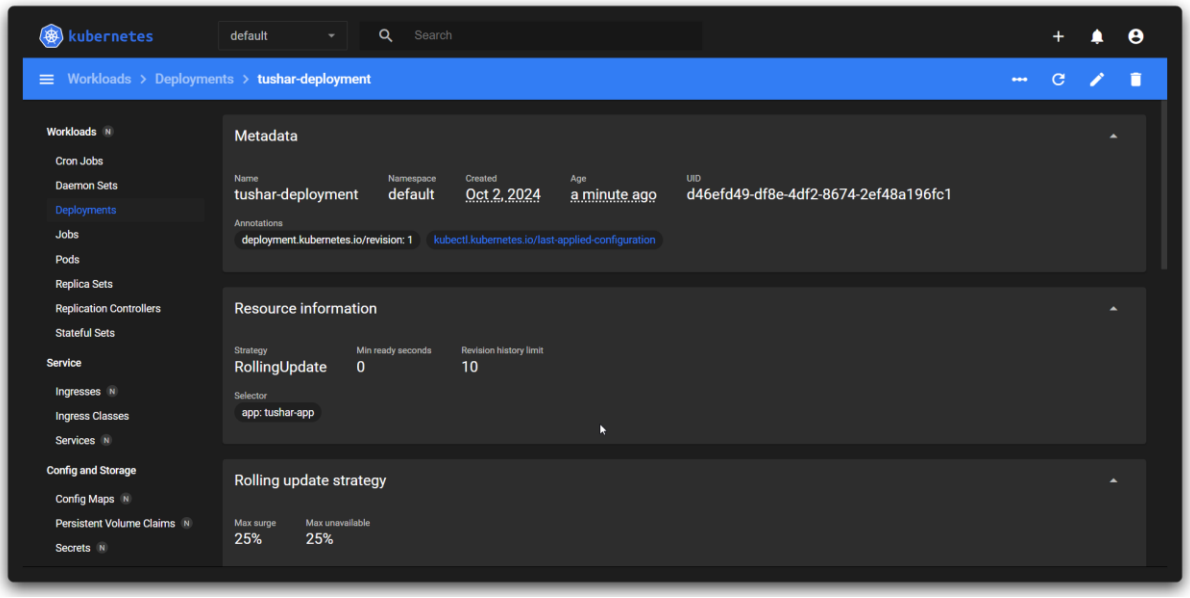


You don't have to add this as your group member has done

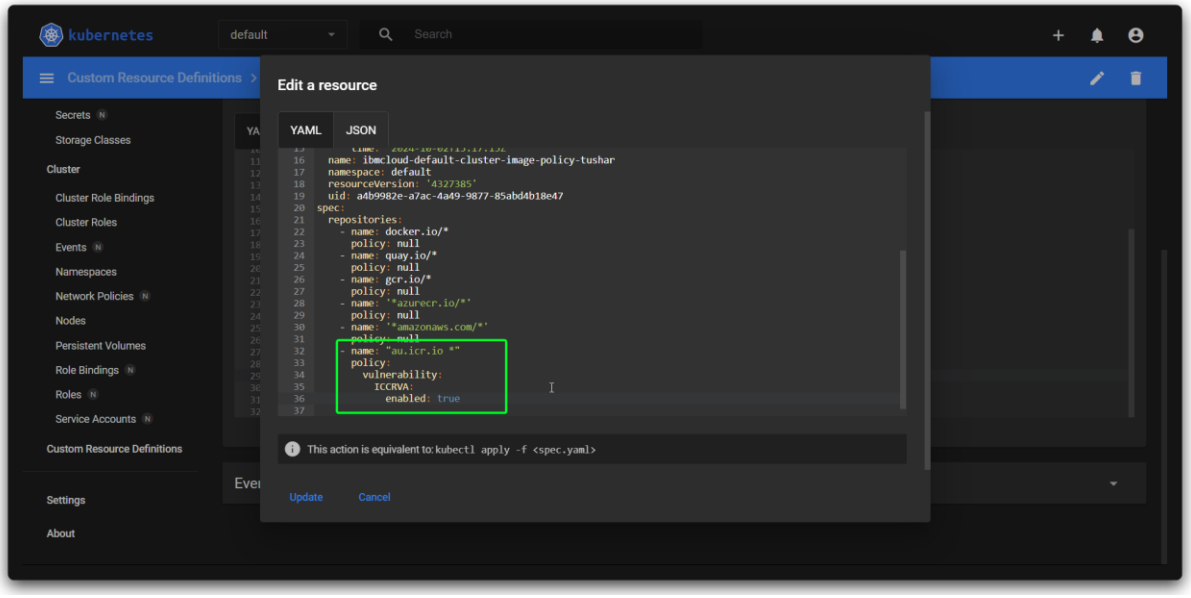


After setting the image policy now you can see you don't have access to deploy the YAML file as the image is not signed

Now as you can see the deployment file is created successfully



Now try to enable vulnerability checker via policy



Now you add the these three lines in vulnerability and check for the image with issue it can deploy.