Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 6

Scenario : 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.**

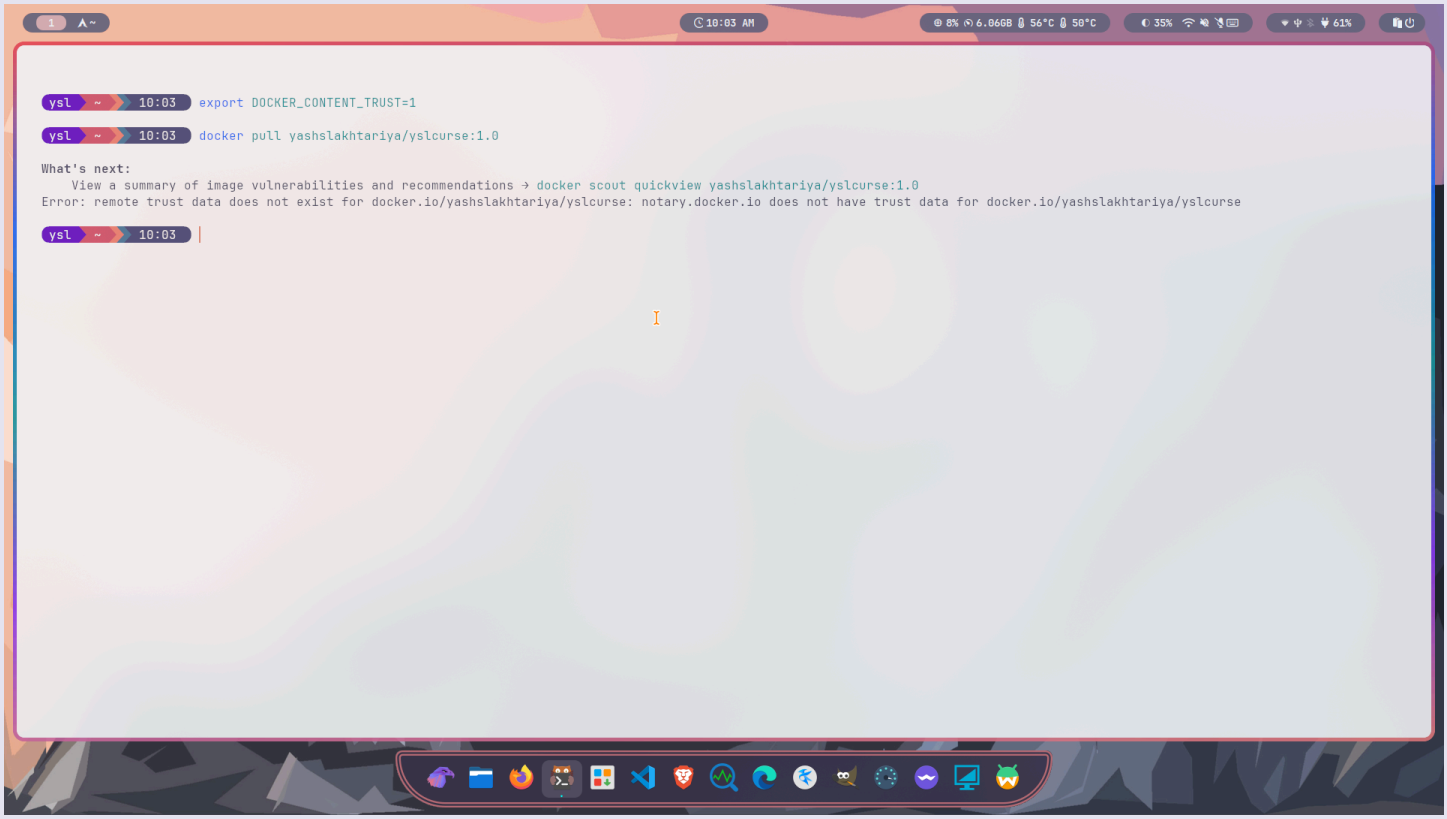
Steps and Screenshots :

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 6

1. Try locally, pulling any docker image unsigned and untrusted

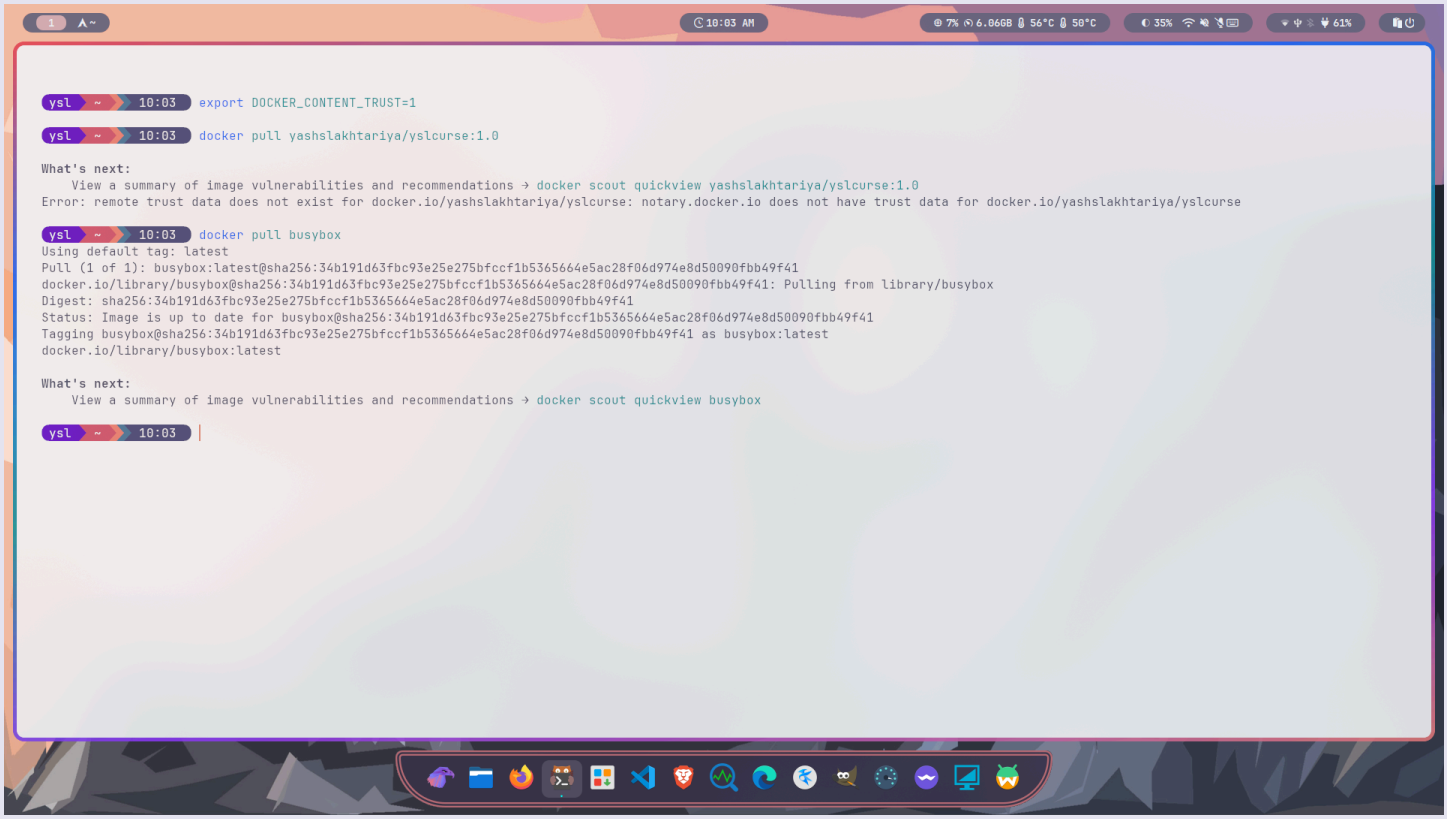
Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 6

2. Export Docker Trust environmental variable and try again, it will fail



Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 6

3. Try pulling busybox, which is signed and trusted image from docker hub



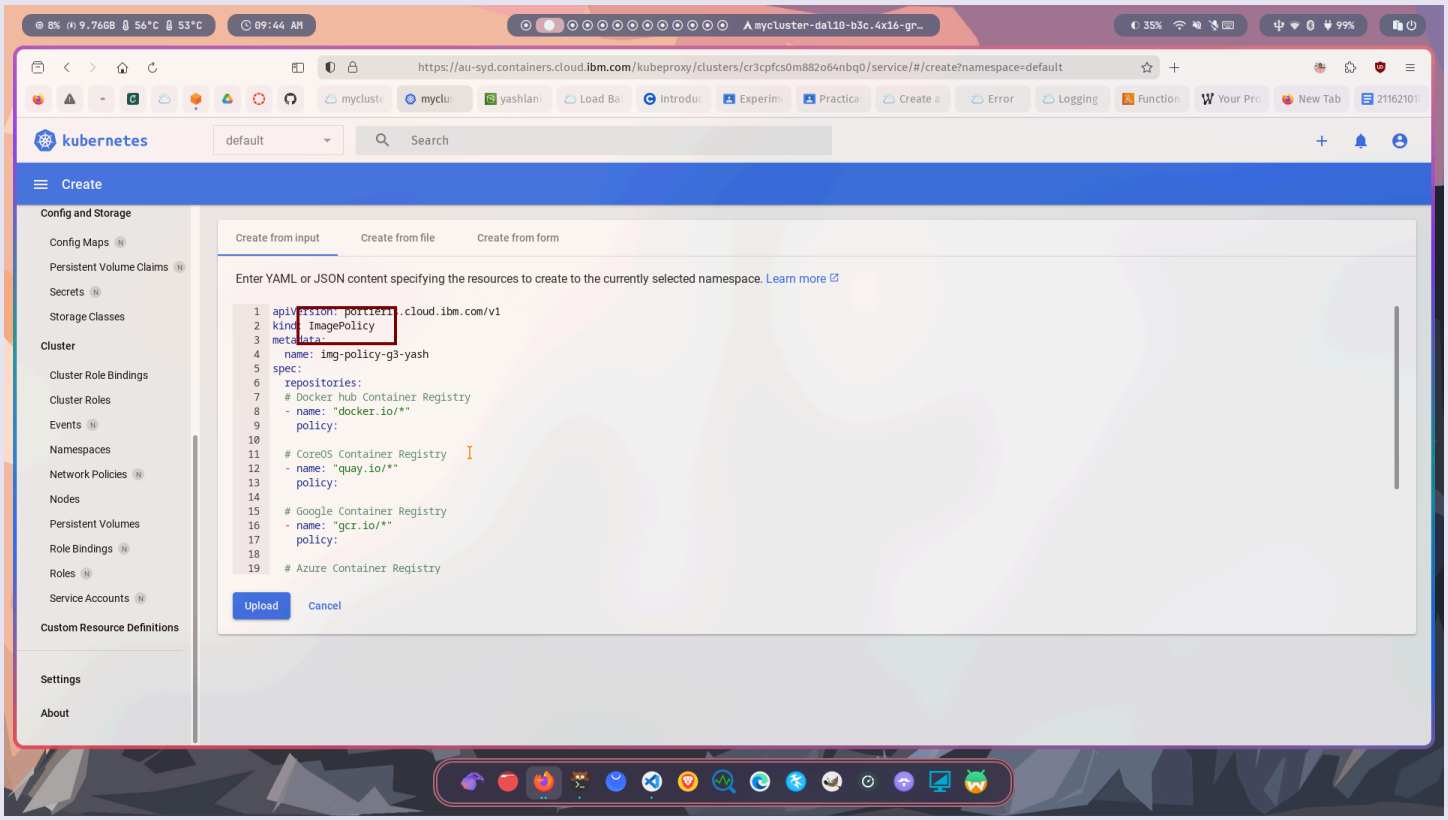
Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 6

4. Now, try on IBM Cloud Kubernetes Cluster using image policy :

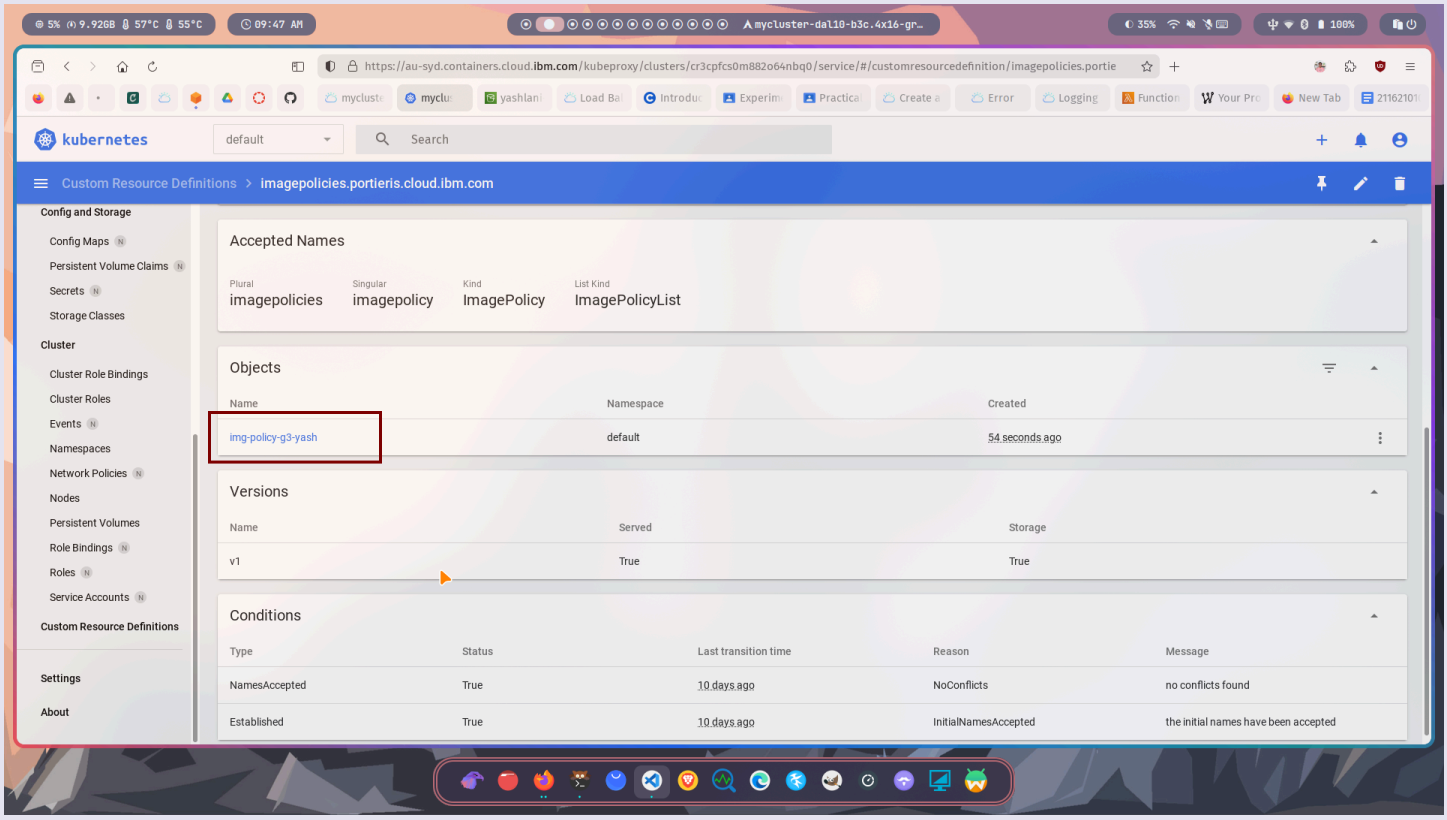
```
apiVersion: portieris.cloud.ibm.com/v1
kind: ImagePolicy
metadata:
 name: img-policy-g3-yash
spec:
 repositories:
 # Docker hub Container Registry
 - name: "docker.io/*"
   policy:

 # 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 - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 6

```
- name: "*amazonaws.com/*"
  policy:
```

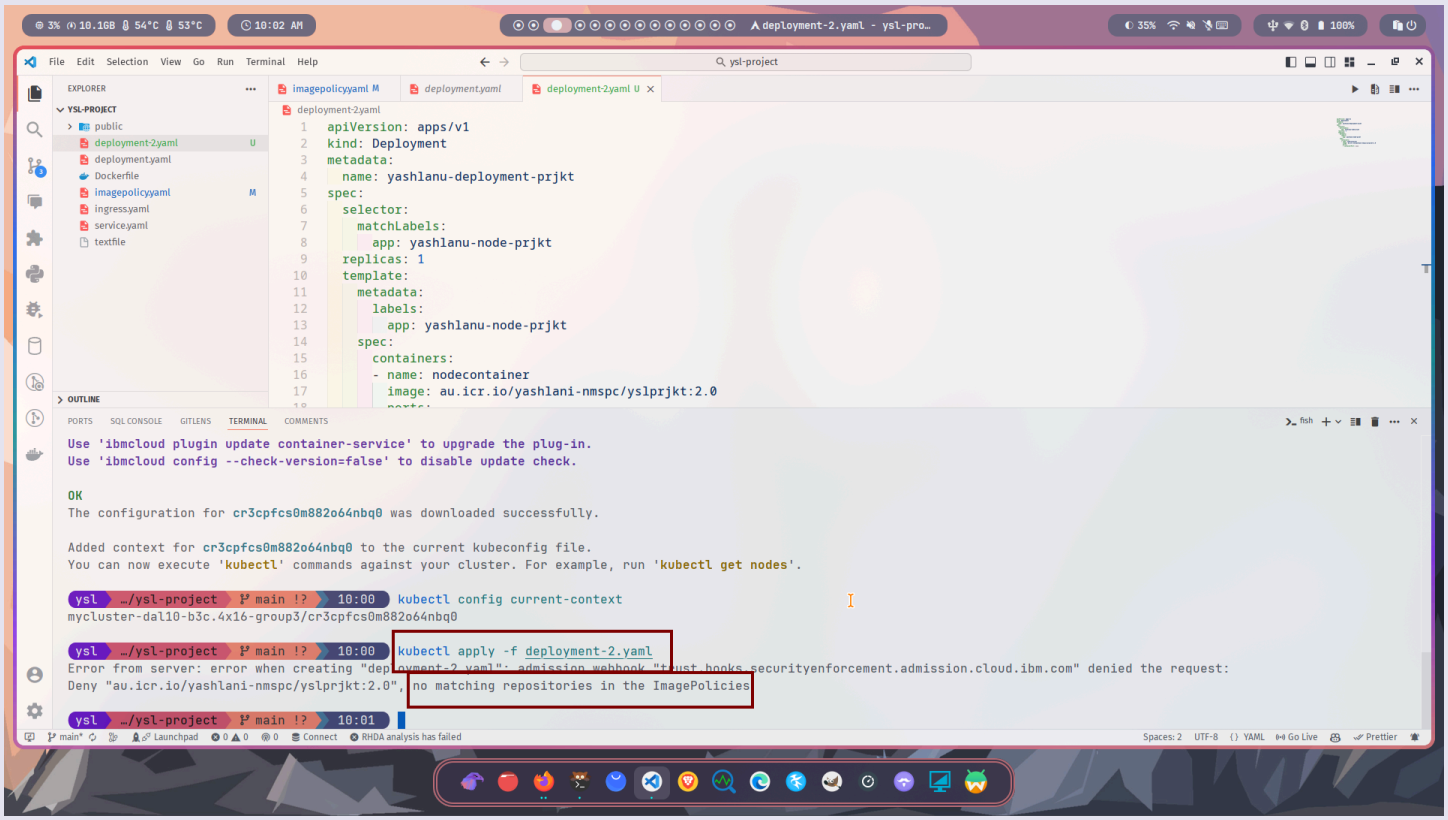


Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 6



Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 6

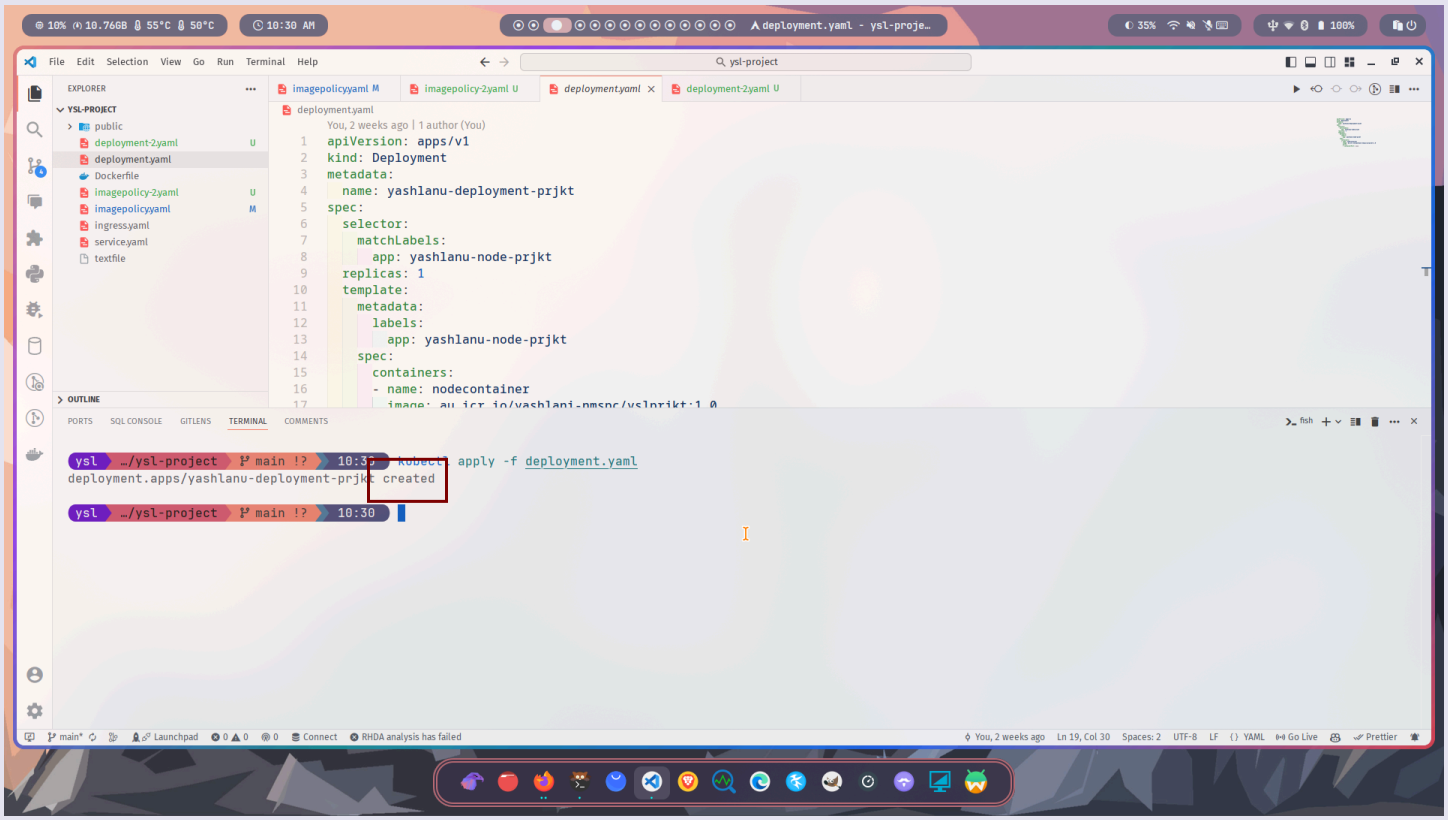
5. Create deployment from previous image from IBM Container Registry, which is not allowed in above image policy, it will return error



Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 6

6. Now, try allowing icr registry

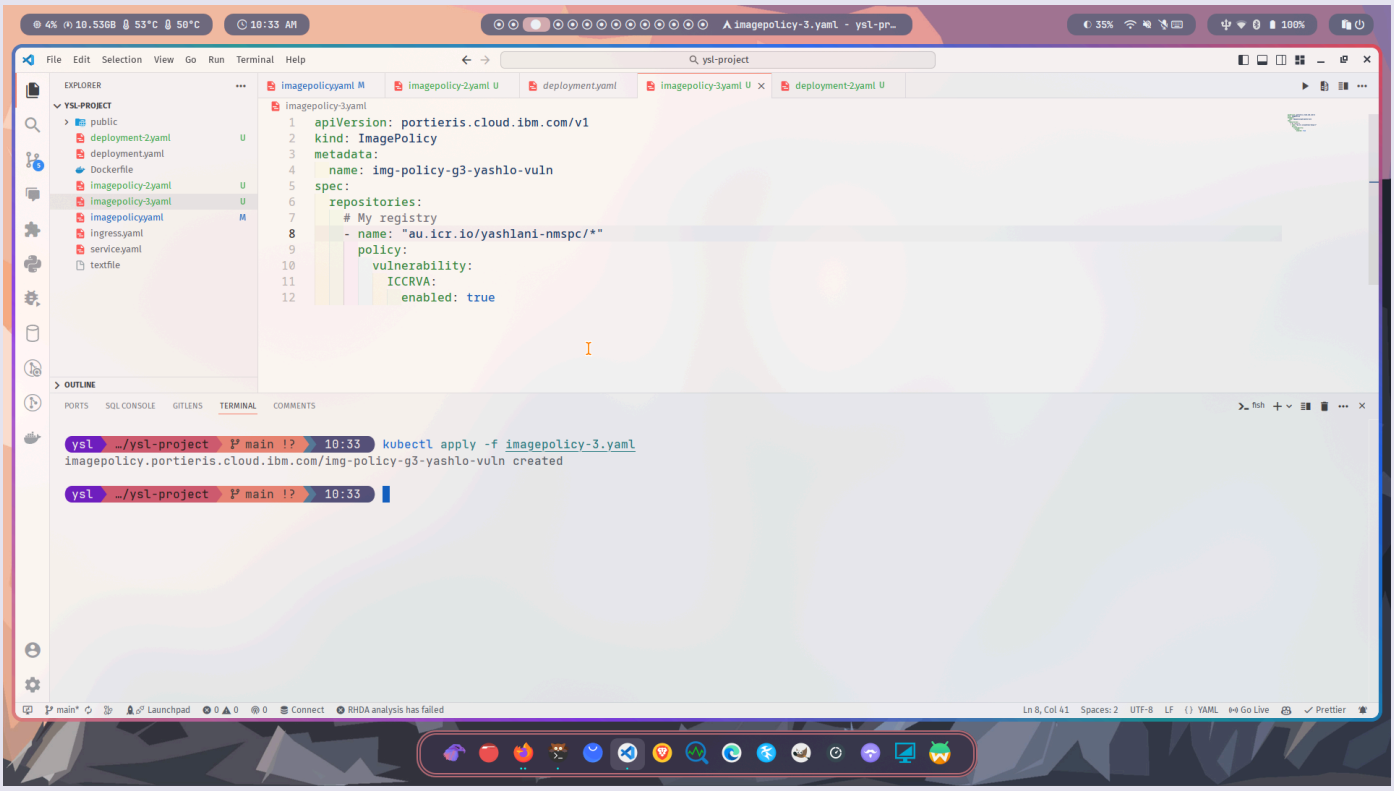
```
apiVersion: portieris.cloud.ibm.com/v1
kind: ImagePolicy
metadata:
 name: img-policy-g3-yashlo-only
spec:
 repositories:
 # My registry
 - name: "au.icr.io/*"
   policy:
```



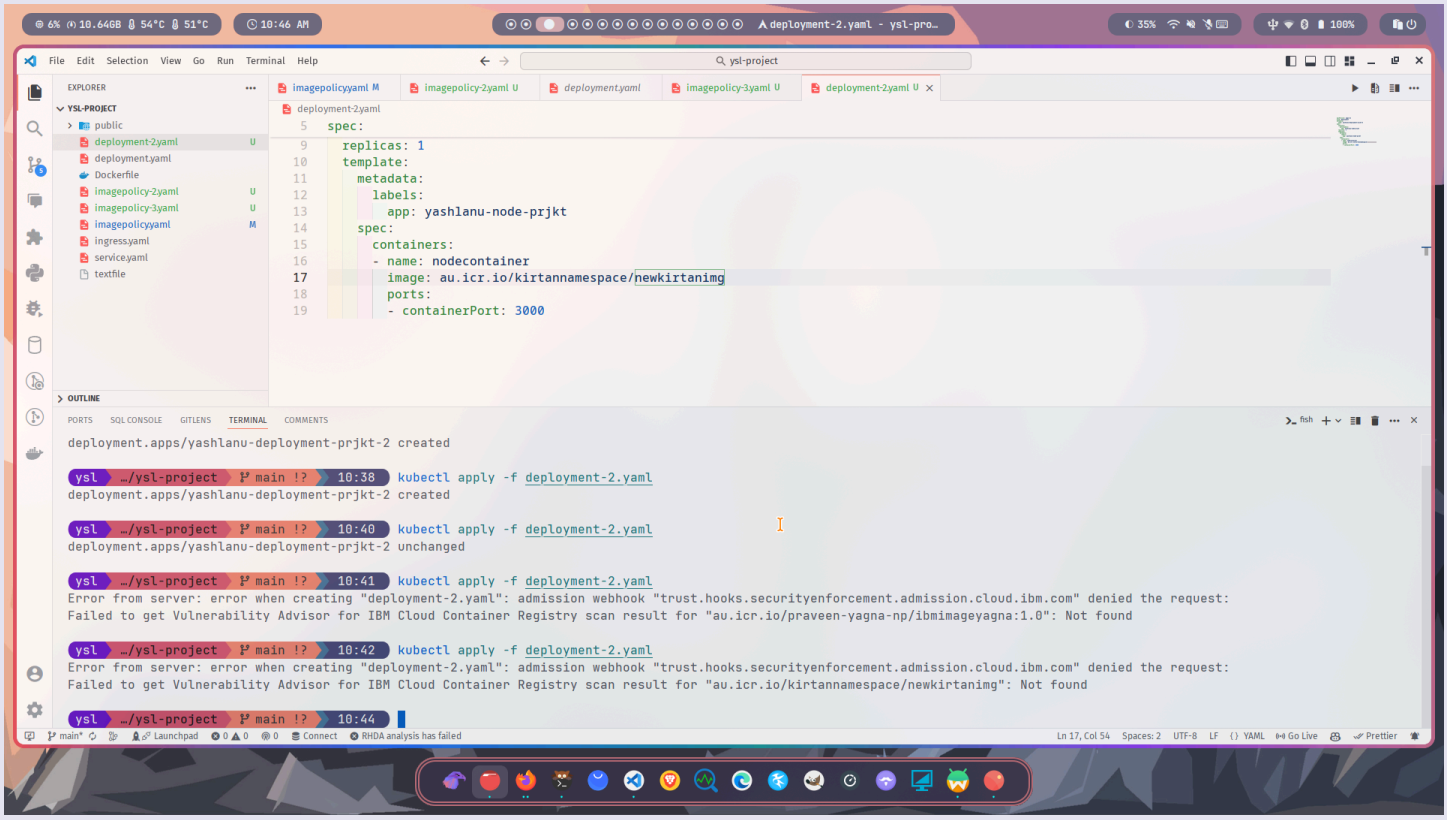
Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 6

7. Now, try enabling vulnerability checker via policy

```yaml
apiVersion: portieris.cloud.ibm.com/v1
kind: ImagePolicy
metadata:
 name: img-policy-g3-yashlo-vuln
spec:
 repositories:
   # My registry
   - name: "au.icr.io/*"
     policy:
       vulnerability:
         ICCRVA:
           enabled: true
```



Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 6



(As seen, trying different images gives error due to issues in vulnerability checker, but otherwise it should give error of unsafe image)