University Practical Exam

1.You are working as an intern at a company that develops and maintains web applications. The company is currently using Docker to containerize its NodeJs applications, your task is to understand the basics of secure containerization. Clone the application https://github.com/singhdeepu/NodeApp

To your local system. This NodeApp has vulnerabilities that you need to find and fix.

a.build the image.

b.Run the hadolint and upload the containerized image to the container registry in IBM cloud. Make sure to include name and enrolment number in the namespace.

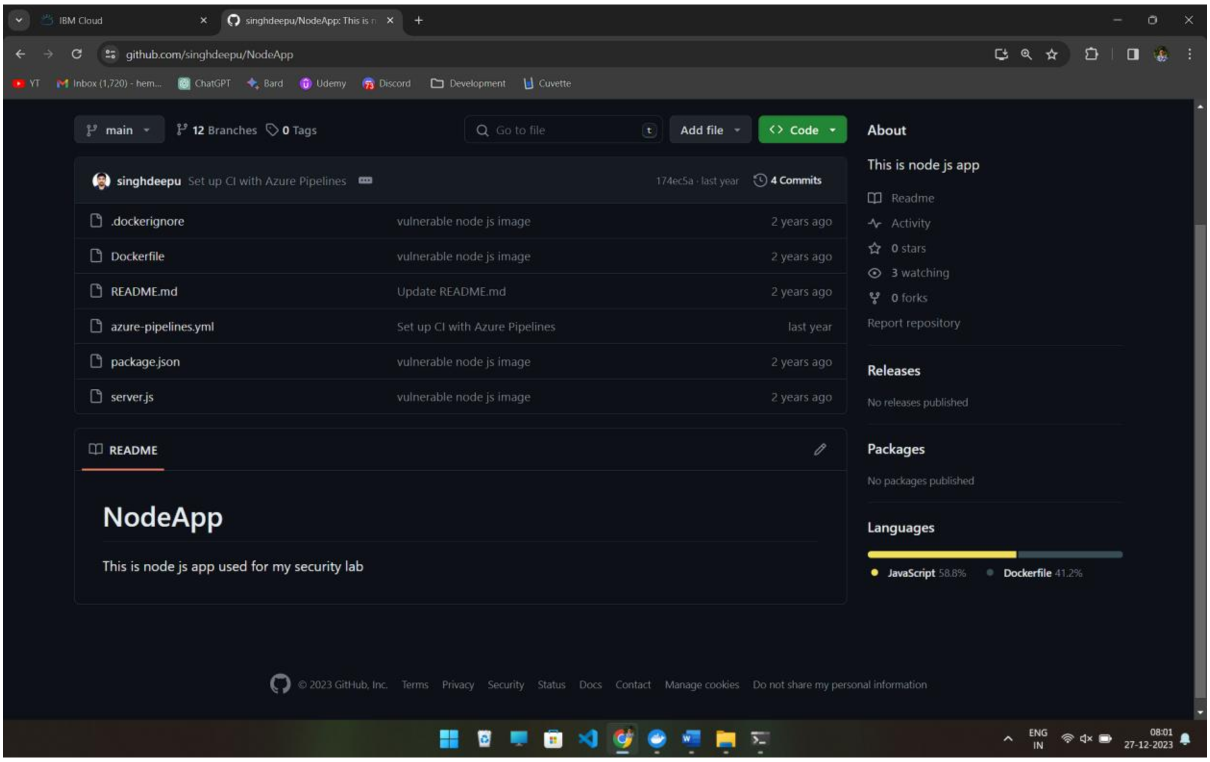
c. Check the vulnerabilities on IBM cloud.

d. Update the docker fil-

https://gist.github.com/singhdeepu/3b442a32358f84a07e35b594903f9a6b

e. Verify the fix with IBM cloud vulnerability advisor.

- Clone the application from the github using git bash commands.
  git clone <URL>



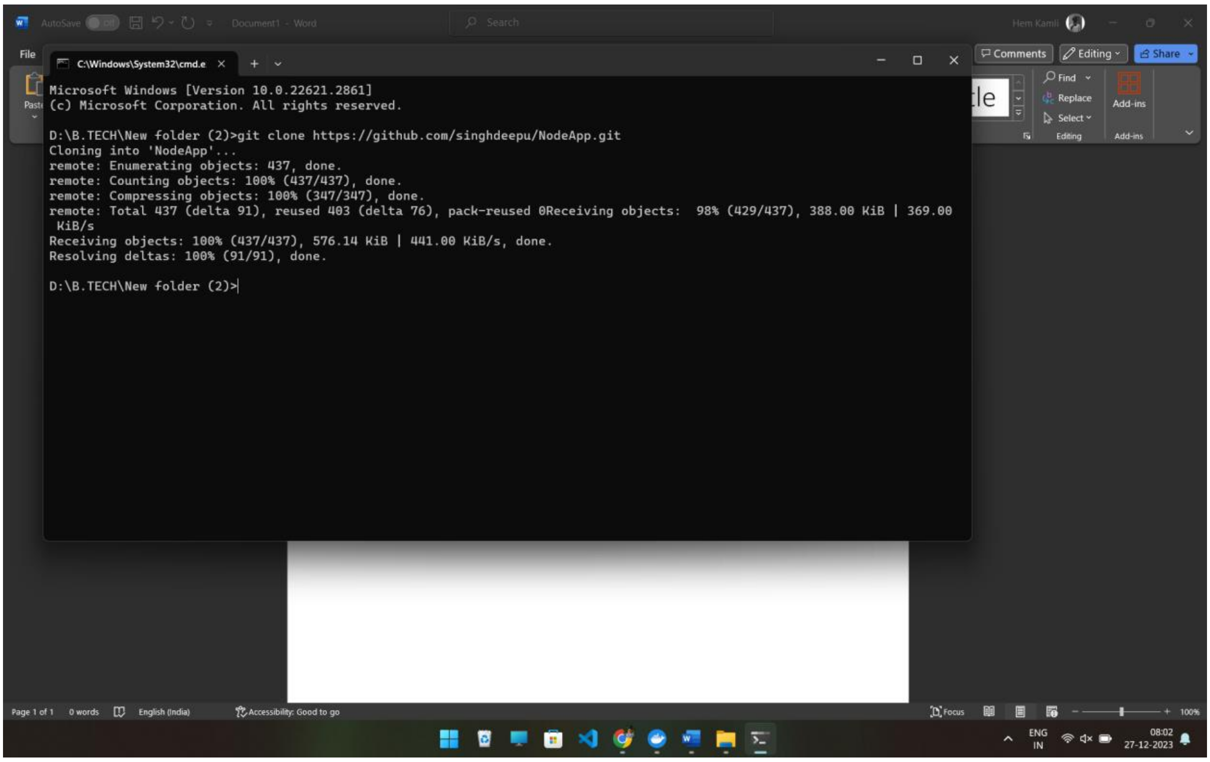
This is node js app

singhdeepu Set up CI with Azure Pipelines 174ec5a · last year 4 Commits

.dockerignore	vulnerable node js image	2 years ago
Dockerfile	vulnerable node js image	2 years ago
README.md	Update README.md	2 years ago
azure-pipelines.yml	Set up CI with Azure Pipelines	last year
package.json	vulnerable node js image	2 years ago
server.js	vulnerable node js image	2 years ago

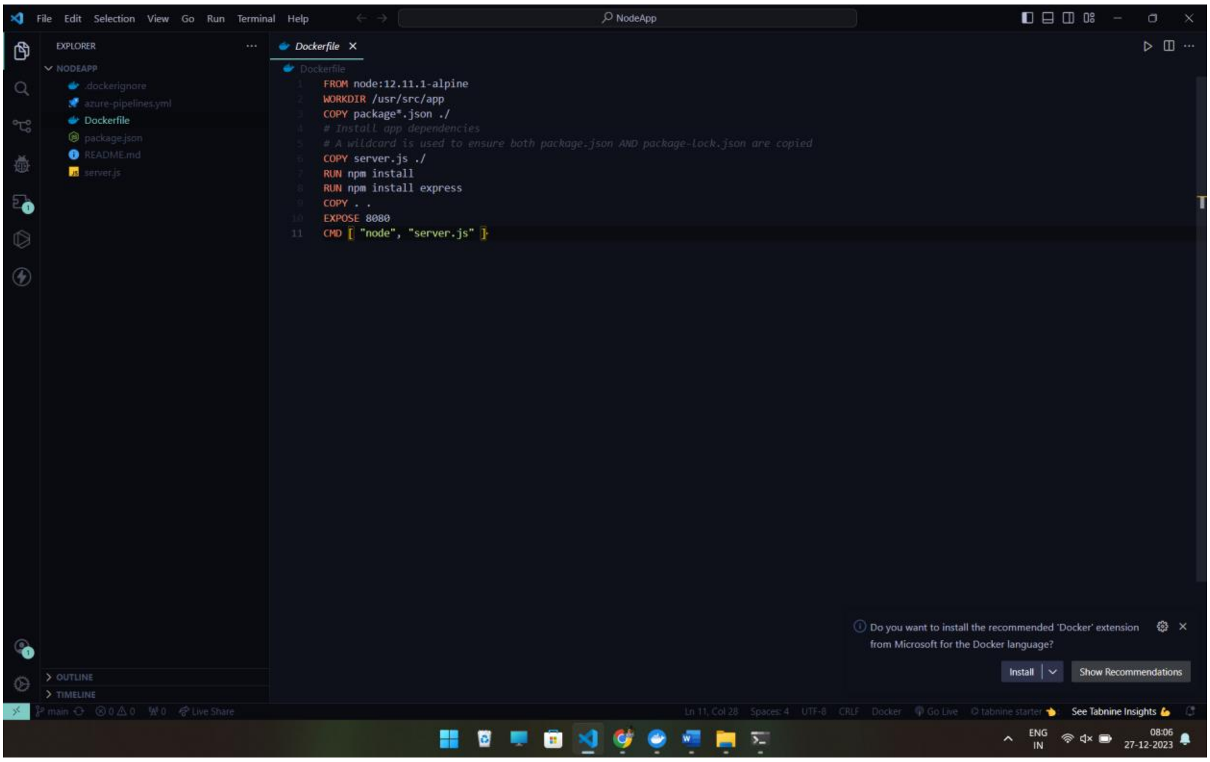
NodeApp

This is node js app used for my security lab



```
Microsoft Windows [Version 10.0.22621.2861]
(c) Microsoft Corporation. All rights reserved.

D:\B.TECH\New folder (2)>git clone https://github.com/singhdeepu/NodeApp.git
Cloning into 'NodeApp'...
remote: Enumerating objects: 437, done.
remote: Counting objects: 100% (437/437), done.
remote: Compressing objects: 100% (347/347), done.
remote: Total 437 (delta 91), reused 403 (delta 76), pack-reused 0Receiving objects:  98% (429/437), 388.00 KiB | 369.00
 KiB/s
Receiving objects: 100% (437/437), 576.14 KiB | 441.00 KiB/s, done.
Resolving deltas: 100% (91/91), done.

D:\B.TECH\New folder (2)>
```

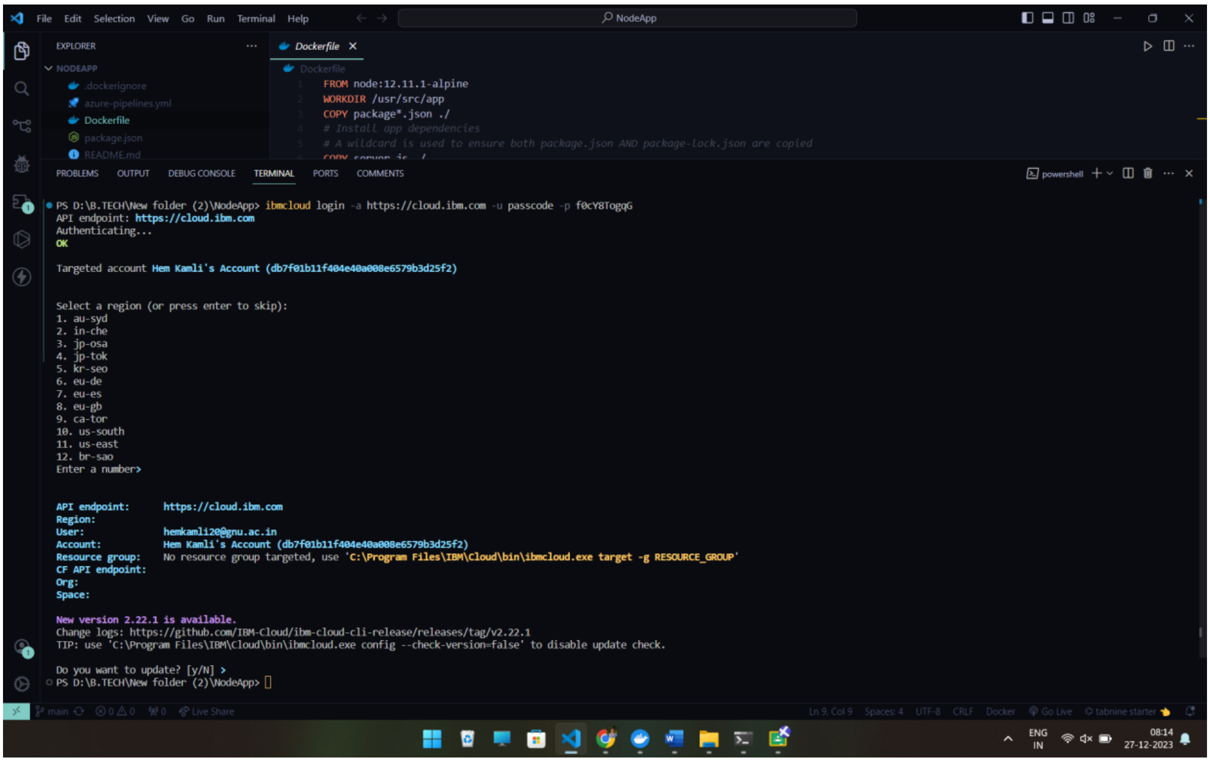
- Open the clone folder to access the code of the node sample application.



```
FROM node:12.11.1-alpine
WORKDIR /usr/src/app
COPY package*.json ./
# Install app dependencies
# A wildcard is used to ensure both package.json AND package-lock.json are copied
COPY server.js ./
RUN npm install
RUN npm install express
COPY . .
EXPOSE 8080
CMD [ "node", "server.js" ]
```

Do you want to install the recommended 'Docker' extension from Microsoft for the Docker language?

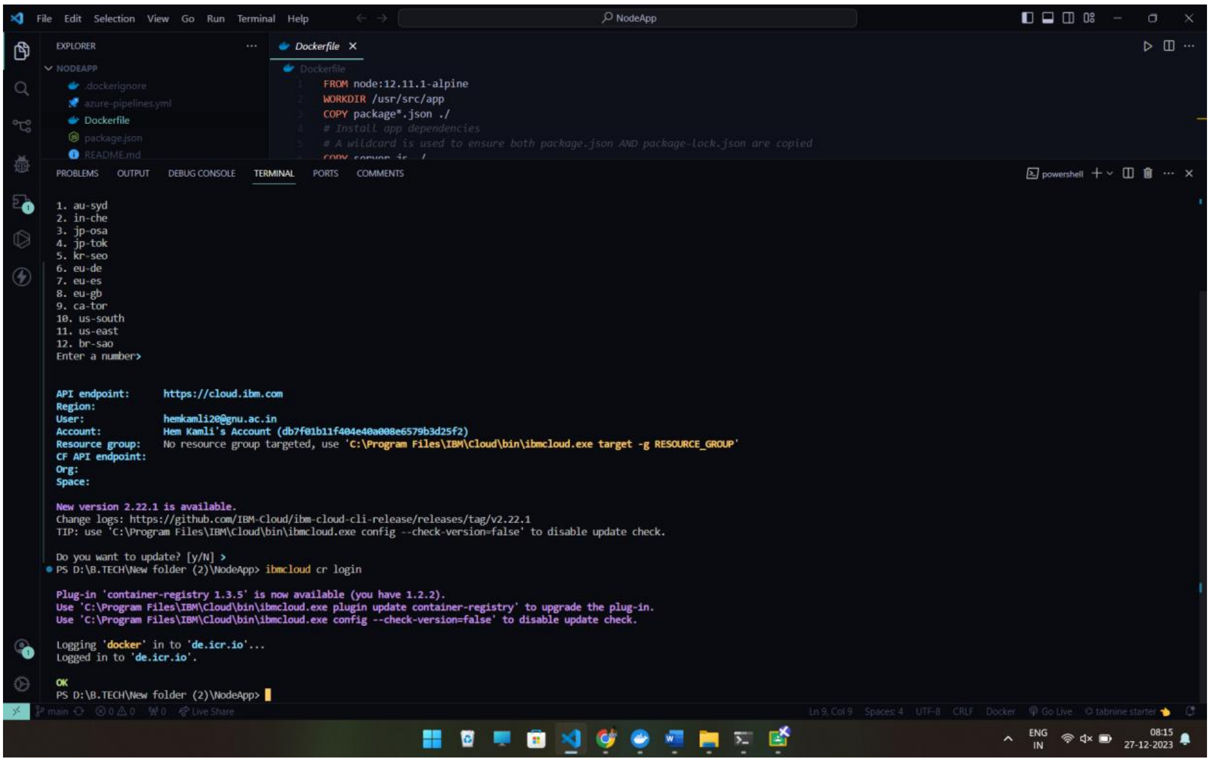
- Login to Ibm cloud using URL from the profile of IBM cloud.



```
FROM node:12.11.1-alpine
WORKDIR /usr/src/app
COPY package*.json ./
# Install app dependencies
# A wildcard is used to ensure both package.json AND package-lock.json are copied
```

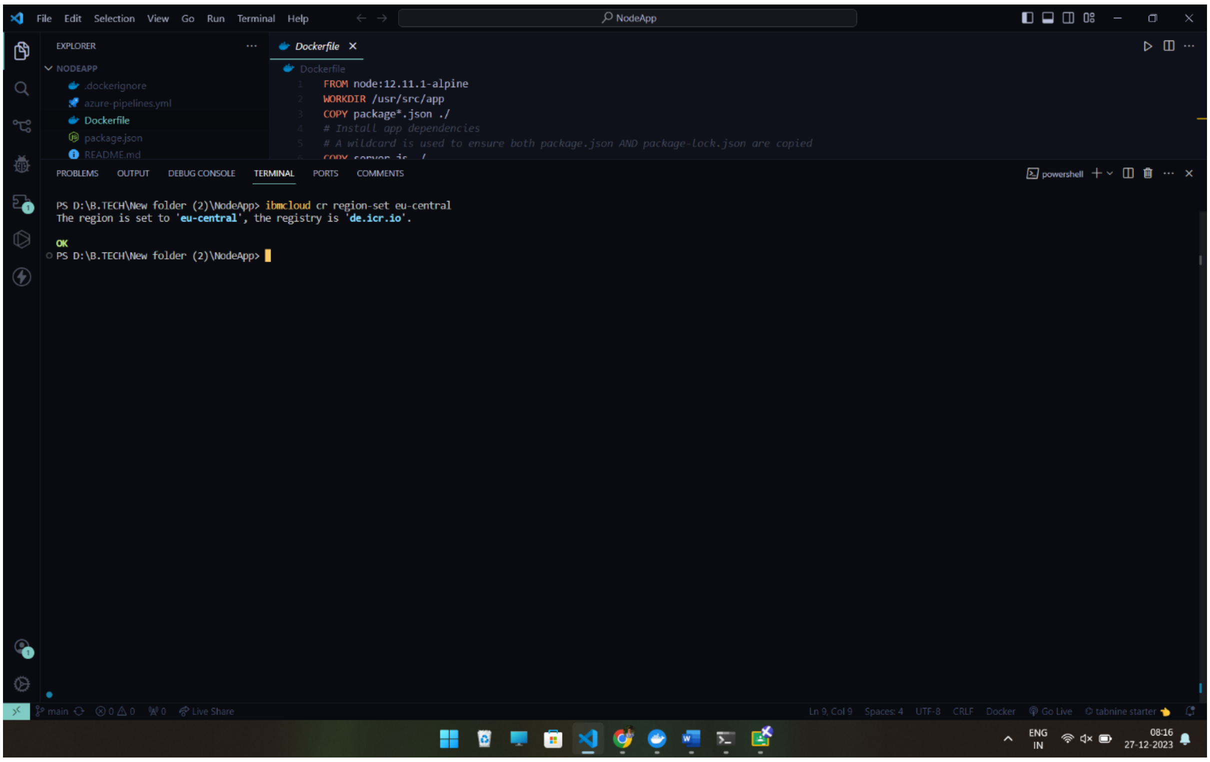
```
PS D:\B.TECH\New folder (2)\NodeApp> ibmcloud login -a https://cloud.ibm.com -u passcode -p f0cY8TogqG
API endpoint: https://cloud.ibm.com
Authenticating...
OK

Targeted account Hem Kamli's Account (db7f01b11f404e40a008e6579b3d25f2)


Select a region (or press enter to skip):
1. au-syd
2. in-che
3. jp-osa
4. jp-tok
5. kr-seo
6. eu-de
7. eu-es
8. eu-gb
9. ca-tor
10. us-south
11. us-east
12. br-sao
Enter a number>


API endpoint:      https://cloud.ibm.com
Region:
User:              hemkamli20@gnu.ac.in
Account:           Hem Kamli's Account (db7f01b11f404e40a008e6579b3d25f2)
Resource group:    No resource group targeted, use 'C:\Program Files\IBM\Cloud\bin\ibmcloud.exe target -g RESOURCE_GROUP'
CF API endpoint:
Org:
Space:

New version 2.22.1 is available.
Change logs: https://github.com/IBM-Cloud/ibm-cloud-cli-release/releases/tag/v2.22.1
TIP: use 'C:\Program Files\IBM\Cloud\bin\ibmcloud.exe config --check-version=false' to disable update check.

Do you want to update? [y/N] >
PS D:\B.TECH\New folder (2)\NodeApp>
```

- Login to cloud registry using this command:
  ibmcloud cr login



```
1. au-syd
2. in-che
3. jp-osa
4. jp-tok
5. kr-seo
6. eu-de
7. eu-es
8. eu-gb
9. ca-tor
10. us-south
11. us-east
12. br-sao
Enter a number>


API endpoint:      https://cloud.ibm.com
Region:
User:              hemkamli20@gnu.ac.in
Account:           Hem Kamli's Account (db7f01b11f404e40a008e6579b3d25f2)
Resource group:    No resource group targeted, use 'C:\Program Files\IBM\Cloud\bin\ibmcloud.exe target -g RESOURCE_GROUP'
CF API endpoint:
Org:
Space:

New version 2.22.1 is available.
Change logs: https://github.com/IBM-Cloud/ibm-cloud-cli-release/releases/tag/v2.22.1
TIP: use 'C:\Program Files\IBM\Cloud\bin\ibmcloud.exe config --check-version=false' to disable update check.

Do you want to update? [y/N] >
PS D:\B.TECH\New folder (2)\NodeApp> ibmcloud cr login

Plug-in 'container-registry 1.3.5' is now available (you have 1.2.2).
Use 'C:\Program Files\IBM\Cloud\bin\ibmcloud.exe plugin update container-registry' to upgrade the plug-in.
Use 'C:\Program Files\IBM\Cloud\bin\ibmcloud.exe config --check-version=false' to disable update check.

Logging 'docker' in to 'de.icr.io'...
Logged in to 'de.icr.io'.

OK
PS D:\B.TECH\New folder (2)\NodeApp>
```

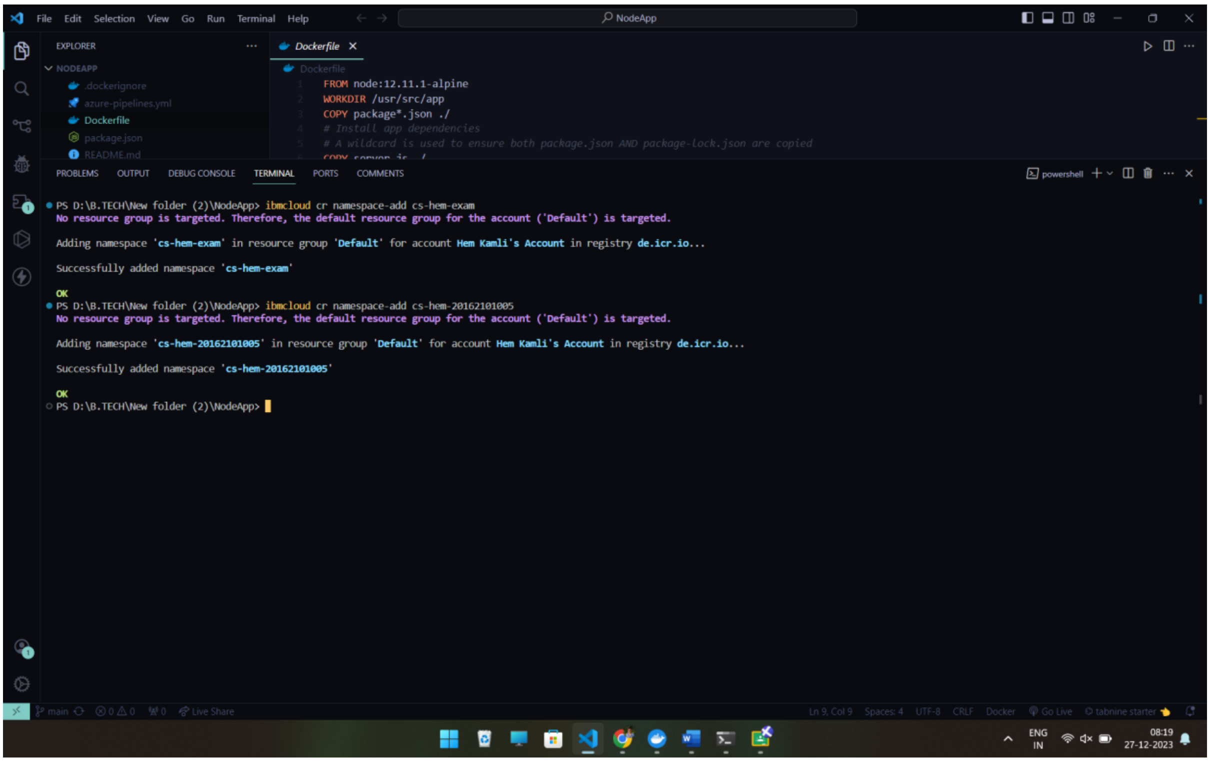
- Set the region for the cloud registry.



```
FROM node:12.11.1-alpine
WORKDIR /usr/src/app
COPY package*.json ./
# Install app dependencies
# A wildcard is used to ensure both package.json AND package-lock.json are copied
```

```
PS D:\B.TECH\New folder (2)\NodeApp> ibmcloud cr region-set eu-central
The region is set to 'eu-central', the registry is 'de.icr.io'.

OK
PS D:\B.TECH\New folder (2)\NodeApp>
```

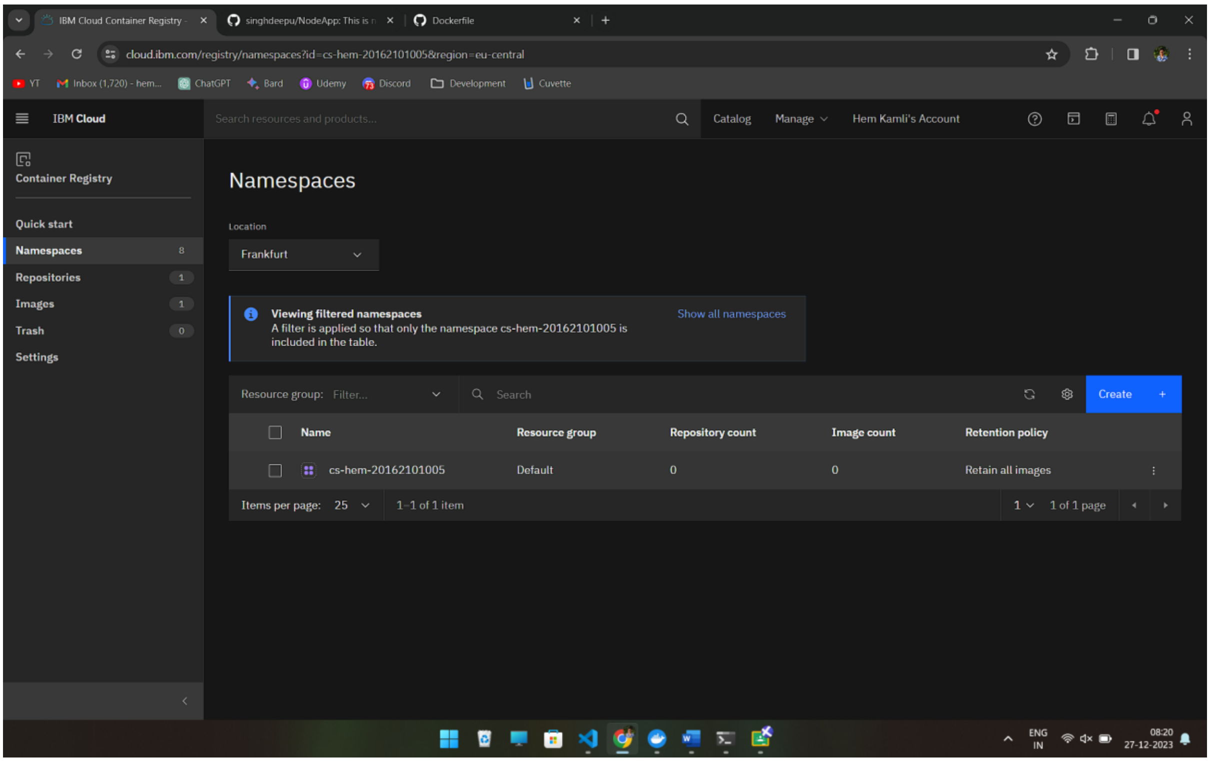
- Add a new namespace to the cloud registry.



```
FROM node:12.11.1-alpine
WORKDIR /usr/src/app
COPY package*.json ./
# Install app dependencies
# A wildcard is used to ensure both package.json AND package-lock.json are copied
```

```
PS D:\B.TECH\New folder (2)\NodeApp> ibmcloud cr namespace-add cs-hem-exam
No resource group is targeted. Therefore, the default resource group for the account ('Default') is targeted.

Adding namespace 'cs-hem-exam' in resource group 'Default' for account Hem Kamli's Account in registry de.icr.io...

Successfully added namespace 'cs-hem-exam'

OK
PS D:\B.TECH\New folder (2)\NodeApp> ibmcloud cr namespace-add cs-hem-20162101005
No resource group is targeted. Therefore, the default resource group for the account ('Default') is targeted.

Adding namespace 'cs-hem-20162101005' in resource group 'Default' for account Hem Kamli's Account in registry de.icr.io...

Successfully added namespace 'cs-hem-20162101005'

OK
PS D:\B.TECH\New folder (2)\NodeApp>
```

- Verify that namespace has been created.



Namespaces

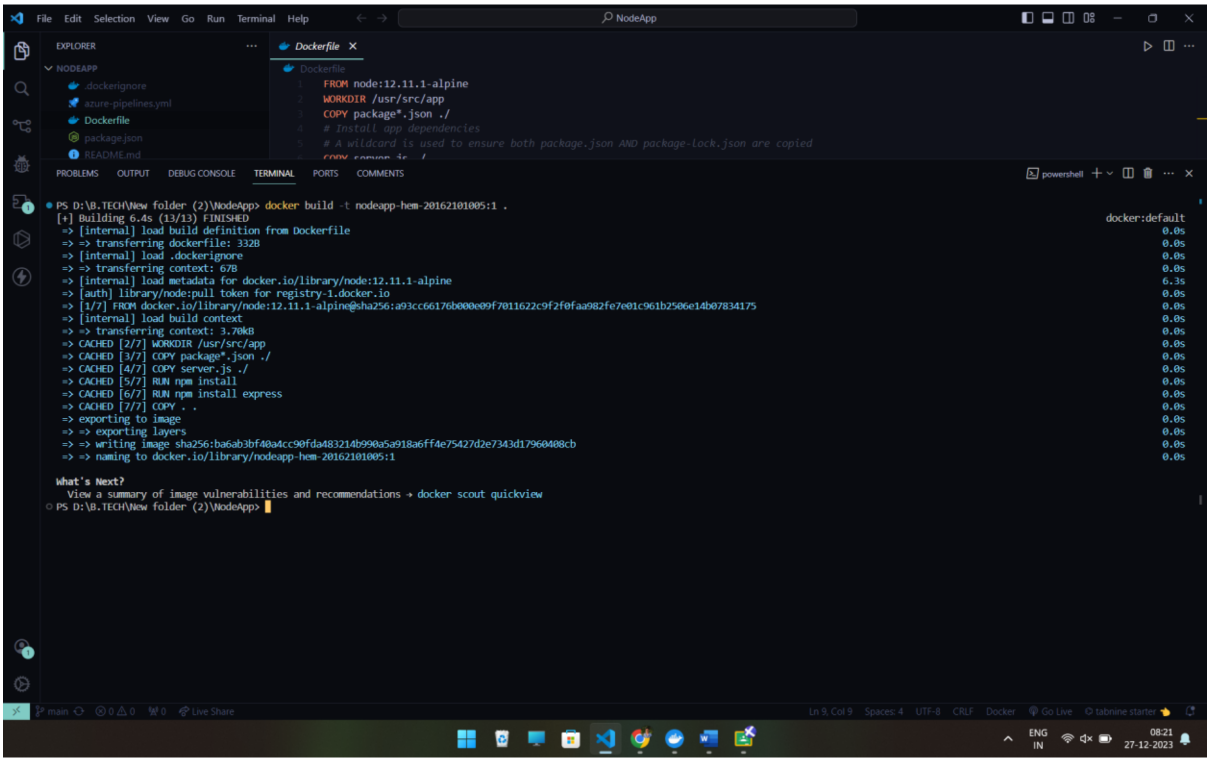
Location: Frankfurt

Viewing filtered namespaces
A filter is applied so that only the namespace cs-hem-20162101005 is included in the table.
Show all namespaces

Name	Resource group	Repository count	Image count	Retention policy
cs-hem-20162101005	Default	0	0	Retain all images

Items per page: 25 1–1 of 1 item 1 1 of 1 page

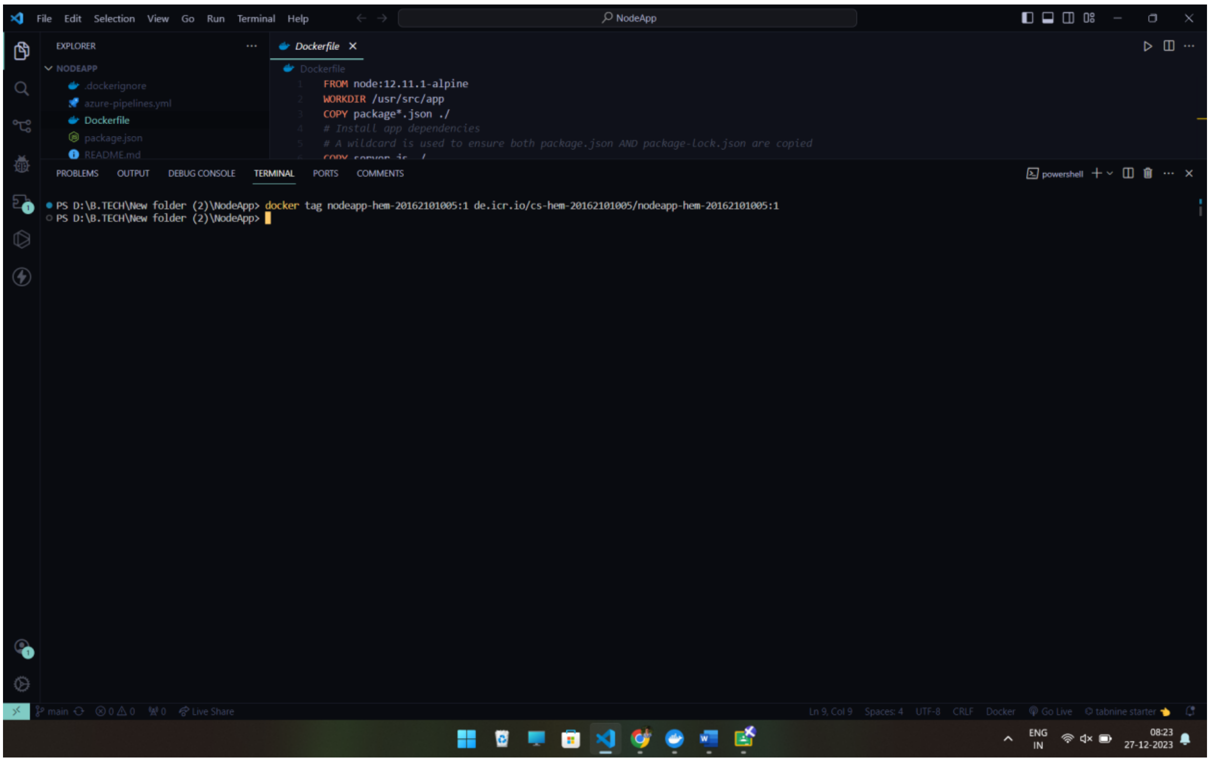
- Now, Build the docker image using the given instruction in dockerfile.



```
FROM node:12.11.1-alpine
WORKDIR /usr/src/app
COPY package*.json ./
# Install app dependencies
# A wildcard is used to ensure both package.json AND package-lock.json are copied
```

```
PS D:\B.TECH\New folder (2)\NodeApp> docker build -t nodeapp-hem-20162101005:1 .
[+] Building 6.4s (13/13) FINISHED                                                   docker:default
 => [internal] load build definition from Dockerfile                                            0.0s
 => => transferring dockerfile: 332B                                                            0.0s
 => [internal] load .dockerignore                                                               0.0s
 => => transferring context: 67B                                                                0.0s
 => [internal] load metadata for docker.io/library/node:12.11.1-alpine                          6.3s
 => [auth] library/node:pull token for registry-1.docker.io                                     0.0s
 => [1/7] FROM docker.io/library/node:12.11.1-alpine@sha256:a93cc66176b000e09f7011622c9f2f0faa982fe7e01c961b2506e14b07834175  0.0s
 => [internal] load build context                                                               0.0s
 => => transferring context: 3.70kB                                                             0.0s
 => CACHED [2/7] WORKDIR /usr/src/app                                                           0.0s
 => CACHED [3/7] COPY package*.json ./                                                          0.0s
 => CACHED [4/7] COPY server.js ./                                                              0.0s
 => CACHED [5/7] RUN npm install                                                                0.0s
 => CACHED [6/7] RUN npm install express                                                        0.0s
 => CACHED [7/7] COPY . .                                                                       0.0s
 => exporting to image                                                                          0.0s
 => => exporting layers                                                                         0.0s
 => => writing image sha256:ba6ab3bf40a4cc90fda483214b990a5a918a6ff4e75427d2e7343d17960408cb    0.0s
 => => naming to docker.io/library/nodeapp-hem-20162101005:1                                    0.0s

What's Next?
  View a summary of image vulnerabilities and recommendations → docker scout quickview
PS D:\B.TECH\New folder (2)\NodeApp>
```

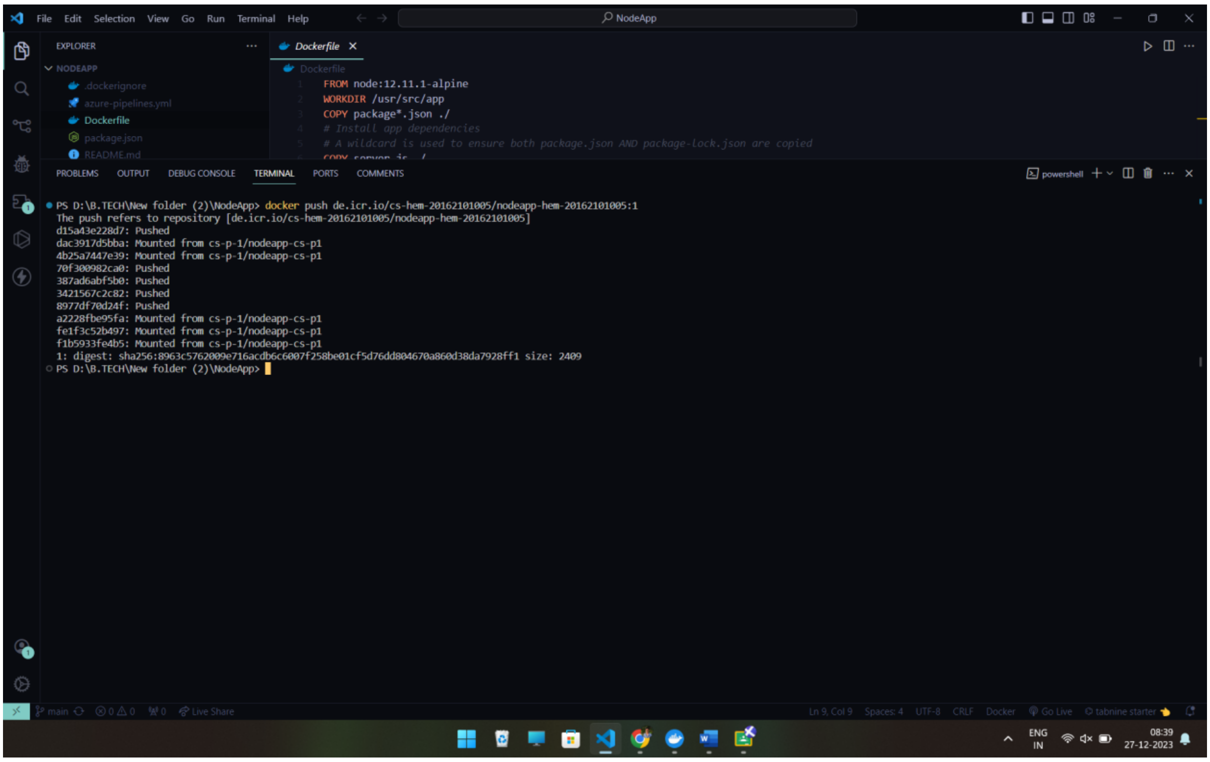
- Tag the docker image for the docker hub repository.


File Edit Selection View Go Run Terminal Help
NodeApp
EXPLORER
NODEAPP
.dockerignore
azure-pipelines.yml
Dockerfile
package.json
README.md
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS
powershell
main
Live Share
Ln 9, Col 9 Spaces: 4 UTF-8 CRLF Docker Go Live tabnine starter
ENG IN
08:23
27-12-2023

```
FROM node:12.11.1-alpine
WORKDIR /usr/src/app
COPY package*.json ./
# Install app dependencies
# A wildcard is used to ensure both package.json AND package-lock.json are copied
COPY server.js ./
```

```
PS D:\B.TECH\New folder (2)\NodeApp> docker tag nodeapp-hem-20162101005:1 de.icr.io/cs-hem-20162101005/nodeapp-hem-20162101005:1
PS D:\B.TECH\New folder (2)\NodeApp>
```

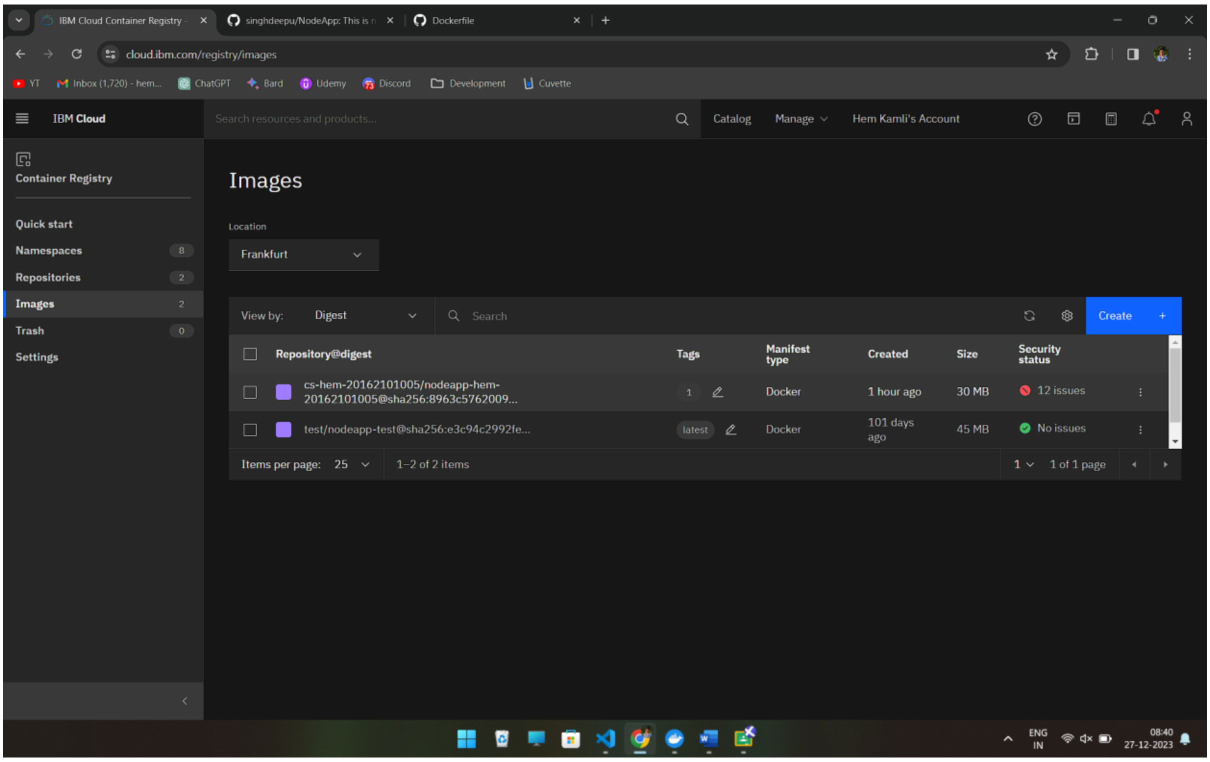
- Push the docker image to the container registry.


File Edit Selection View Go Run Terminal Help
NodeApp
EXPLORER
NODEAPP
.dockerignore
azure-pipelines.yml
Dockerfile
package.json
README.md
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS
powershell
main
Live Share
Ln 9, Col 9 Spaces: 4 UTF-8 CRLF Docker Go Live tabnine starter
ENG IN
08:39
27-12-2023

```
FROM node:12.11.1-alpine
WORKDIR /usr/src/app
COPY package*.json ./
# Install app dependencies
# A wildcard is used to ensure both package.json AND package-lock.json are copied
COPY server.js ./
```

```
PS D:\B.TECH\New folder (2)\NodeApp> docker push de.icr.io/cs-hem-20162101005/nodeapp-hem-20162101005:1
The push refers to repository [de.icr.io/cs-hem-20162101005/nodeapp-hem-20162101005]
d15a43e228d7: Pushed
dac3917d5bba: Mounted from cs-p-1/nodeapp-cs-p1
4b25a7447e39: Mounted from cs-p-1/nodeapp-cs-p1
70f300982ca0: Pushed
387ad6abf5b0: Pushed
3421567c2c82: Pushed
8977df70d24f: Pushed
a2228fbe95fa: Mounted from cs-p-1/nodeapp-cs-p1
fe1f3c52b497: Mounted from cs-p-1/nodeapp-cs-p1
f1b5933fe4b5: Mounted from cs-p-1/nodeapp-cs-p1
1: digest: sha256:8963c5762009e716acdb6c6007f258be01cf5d76dd804670a860d38da7928ff1 size: 2409
PS D:\B.TECH\New folder (2)\NodeApp>
```

- Go to images and select the region Frankfurt to view the pushed image.



Images

Location

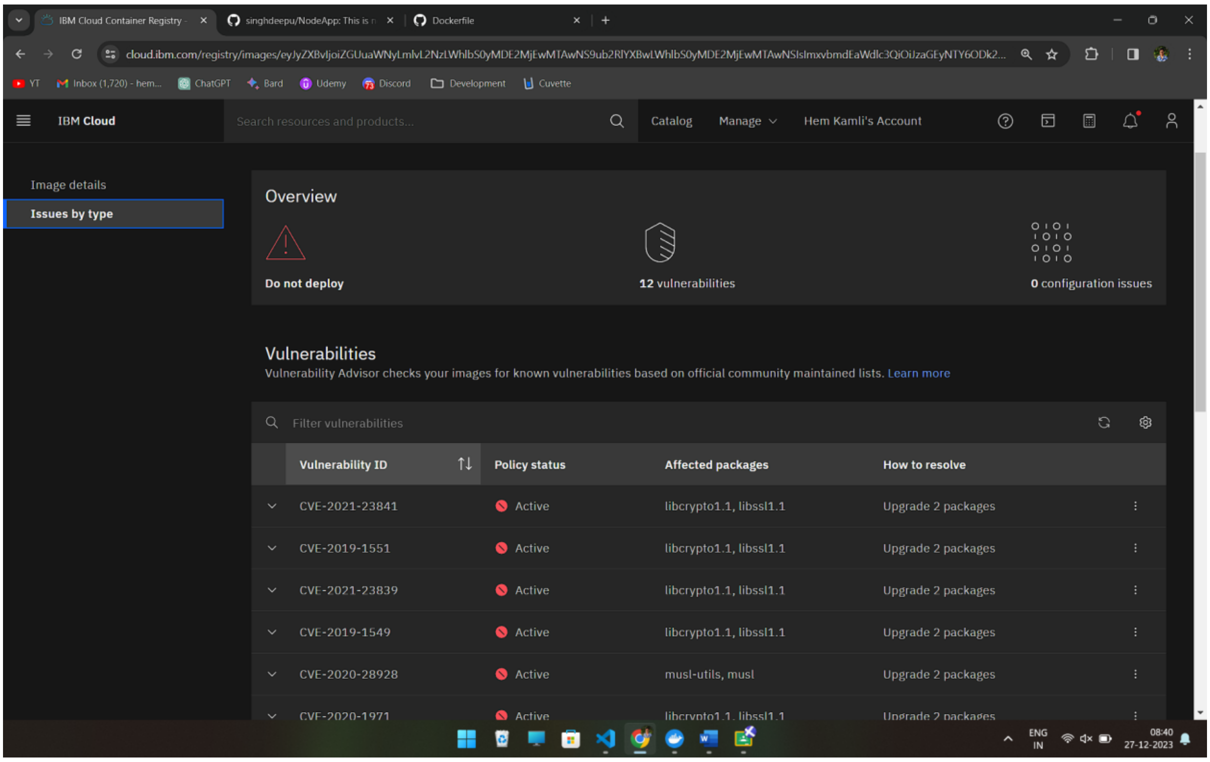
Frankfurt

View by: Digest

	Repository@digest	Tags	Manifest type	Created	Size	Security status
□	cs-hem-20162101005/nodeapp-hem-20162101005@sha256:8963c5762009...	1	Docker	1 hour ago	30 MB	12 issues
□	test/nodeapp-test@sha256:e3c94c2992fe...	latest	Docker	101 days ago	45 MB	No issues

Items per page: 25 1–2 of 2 items 1 1 of 1 page

- In the image, issue by type section shows the vulnerabilities in the docker image or the application.


IBM Cloud Container Registry
singhdeepu/NodeApp: This is
Dockerfile
cloud.ibm.com/registry/images/eyJyZXBvljoiZGUuaWNyLmlvL2NzLWhlbS0yMDE2MjEwMTAwNS9ub2RlYXBwLWhlbS0yMDE2MjEwMTAwNSIsImxvbmdEaWdlc3QiOiJzaGEyNTY6ODk2...
YT
Inbox (1,720) - hem...
ChatGPT
Bard
Udemy
Discord
Development
Cuvette
IBM Cloud
Search resources and products...
Catalog
Manage
Hem Kamli's Account
Image details
Issues by type
ENG
IN
08:40
27-12-2023

Overview

Do not deploy

12 vulnerabilities

0 configuration issues

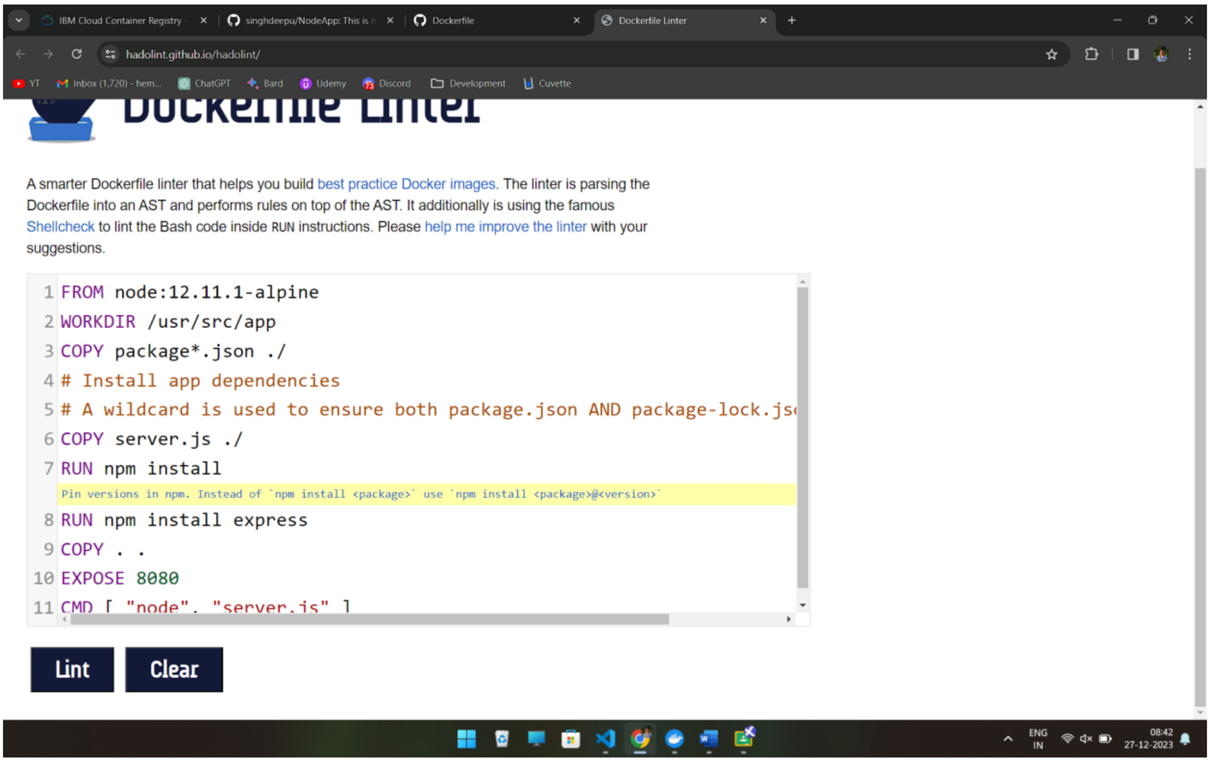
Vulnerabilities

Vulnerability Advisor checks your images for known vulnerabilities based on official community maintained lists. Learn more

Filter vulnerabilities

Vulnerability ID	Policy status	Affected packages	How to resolve
CVE-2021-23841	Active	libcrypto1.1, libssl1.1	Upgrade 2 packages
CVE-2019-1551	Active	libcrypto1.1, libssl1.1	Upgrade 2 packages
CVE-2021-23839	Active	libcrypto1.1, libssl1.1	Upgrade 2 packages
CVE-2019-1549	Active	libcrypto1.1, libssl1.1	Upgrade 2 packages
CVE-2020-28928	Active	musl-utils, musl	Upgrade 2 packages
CVE-2020-1971	Active	libcrypto1.1, libssl1.1	Upgrade 2 packages

- Check the docker file using hadolint to remove the vulnerabilities from the application.

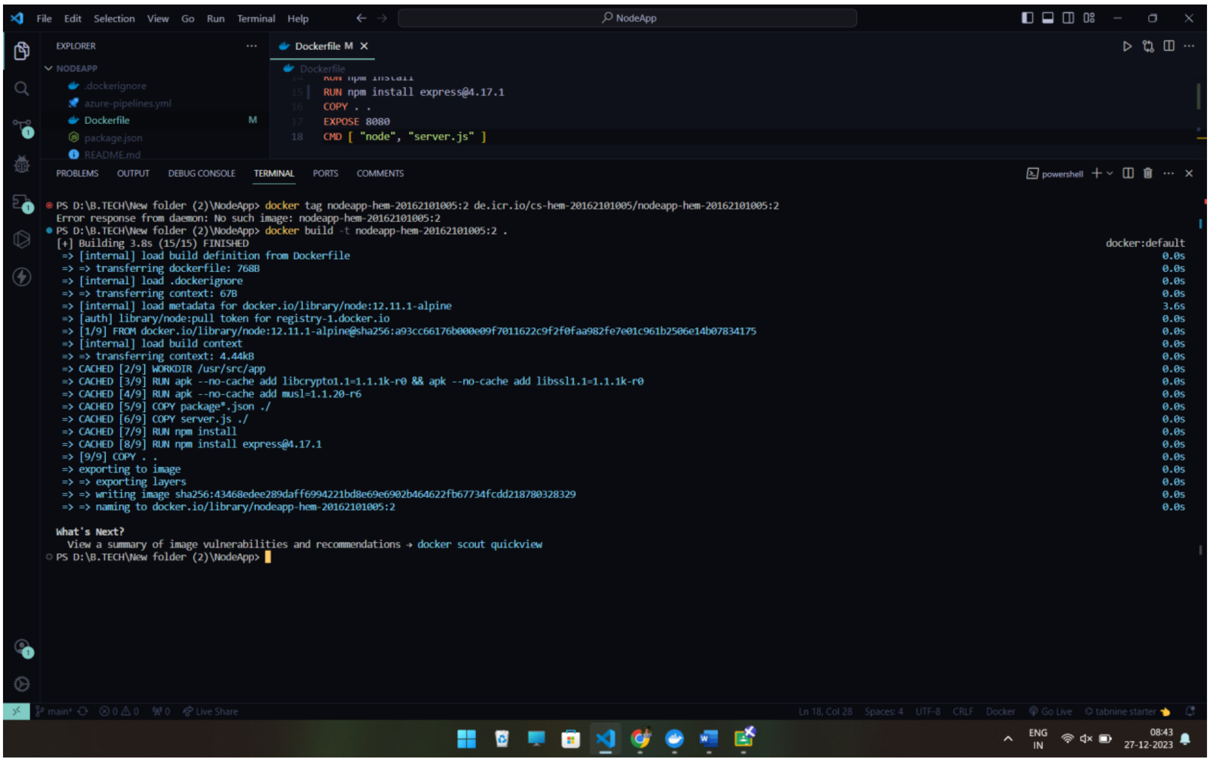


Dockerfile Linter

A smarter Dockerfile linter that helps you build best practice Docker images. The linter is parsing the Dockerfile into an AST and performs rules on top of the AST. It additionally is using the famous Shellcheck to lint the Bash code inside RUN instructions. Please help me improve the linter with your suggestions.

```
FROM node:12.11.1-alpine
WORKDIR /usr/src/app
COPY package*.json ./
# Install app dependencies
# A wildcard is used to ensure both package.json AND package-lock.js
COPY server.js ./
RUN npm install
Pin versions in npm. Instead of `npm install <package>` use `npm install <package>@<version>`
RUN npm install express
COPY . .
EXPOSE 8080
CMD [ "node", "server.js" ]
```

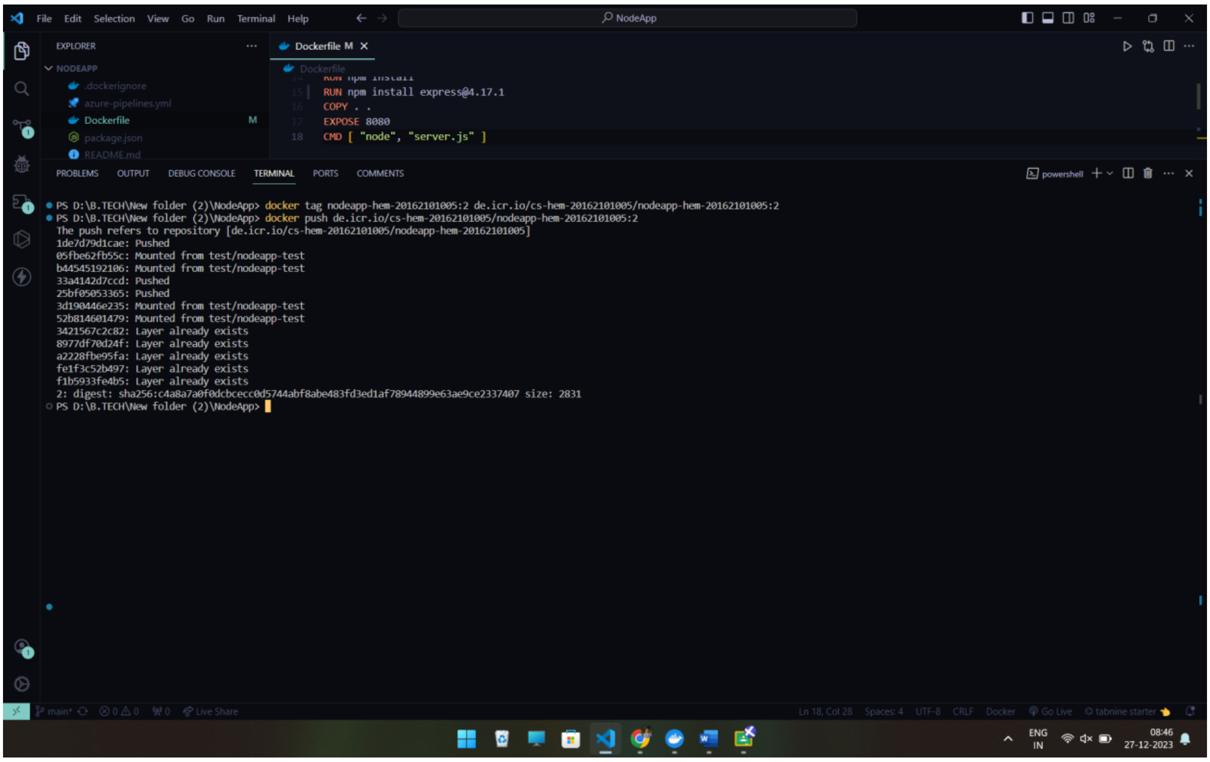
- Again build the image using new updated dockerfile.



```
RUN npm install express@4.17.1
COPY . .
EXPOSE 8080
CMD [ "node", "server.js" ]
```

```
PS D:\B.TECH\New folder (2)\NodeApp> docker tag nodeapp-hem-20162101005:2 de.icr.io/cs-hem-20162101005/nodeapp-hem-20162101005:2
Error response from daemon: No such image: nodeapp-hem-20162101005:2
PS D:\B.TECH\New folder (2)\NodeApp> docker build -t nodeapp-hem-20162101005:2 .
[+] Building 3.8s (15/15) FINISHED                                                              docker:default
 => [internal] load build definition from Dockerfile                                                      0.0s
 => => transferring dockerfile: 768B                                                                      0.0s
 => [internal] load .dockerignore                                                                         0.0s
 => => transferring context: 67B                                                                          0.0s
 => [internal] load metadata for docker.io/library/node:12.11.1-alpine                                    3.6s
 => [auth] library/node:pull token for registry-1.docker.io                                               0.0s
 => [1/9] FROM docker.io/library/node:12.11.1-alpine@sha256:a93cc66176b000e09f7011622c9f2f0faa982fe7e01c961b2506e14b07834175  0.0s
 => [internal] load build context                                                                         0.0s
 => => transferring context: 4.44kB                                                                       0.0s
 => CACHED [2/9] WORKDIR /usr/src/app                                                                     0.0s
 => CACHED [3/9] RUN apk --no-cache add libcrypto1.1=1.1.1k-r0 && apk --no-cache add libssl1.1=1.1.1k-r0  0.0s
 => CACHED [4/9] RUN apk --no-cache add musl=1.1.20-r6                                                    0.0s
 => CACHED [5/9] COPY package*.json ./                                                                    0.0s
 => CACHED [6/9] COPY server.js ./                                                                        0.0s
 => CACHED [7/9] RUN npm install                                                                          0.0s
 => CACHED [8/9] RUN npm install express@4.17.1                                                           0.0s
 => [9/9] COPY . .                                                                                        0.0s
 => exporting to image                                                                                    0.0s
 => => exporting layers                                                                                   0.0s
 => => writing image sha256:43468edee289daff6994221bd8e69e6902b464622fb67734fcdd218780328329               0.0s
 => => naming to docker.io/library/nodeapp-hem-20162101005:2                                              0.0s

What's Next?
  View a summary of image vulnerabilities and recommendations → docker scout quickview
PS D:\B.TECH\New folder (2)\NodeApp>
```

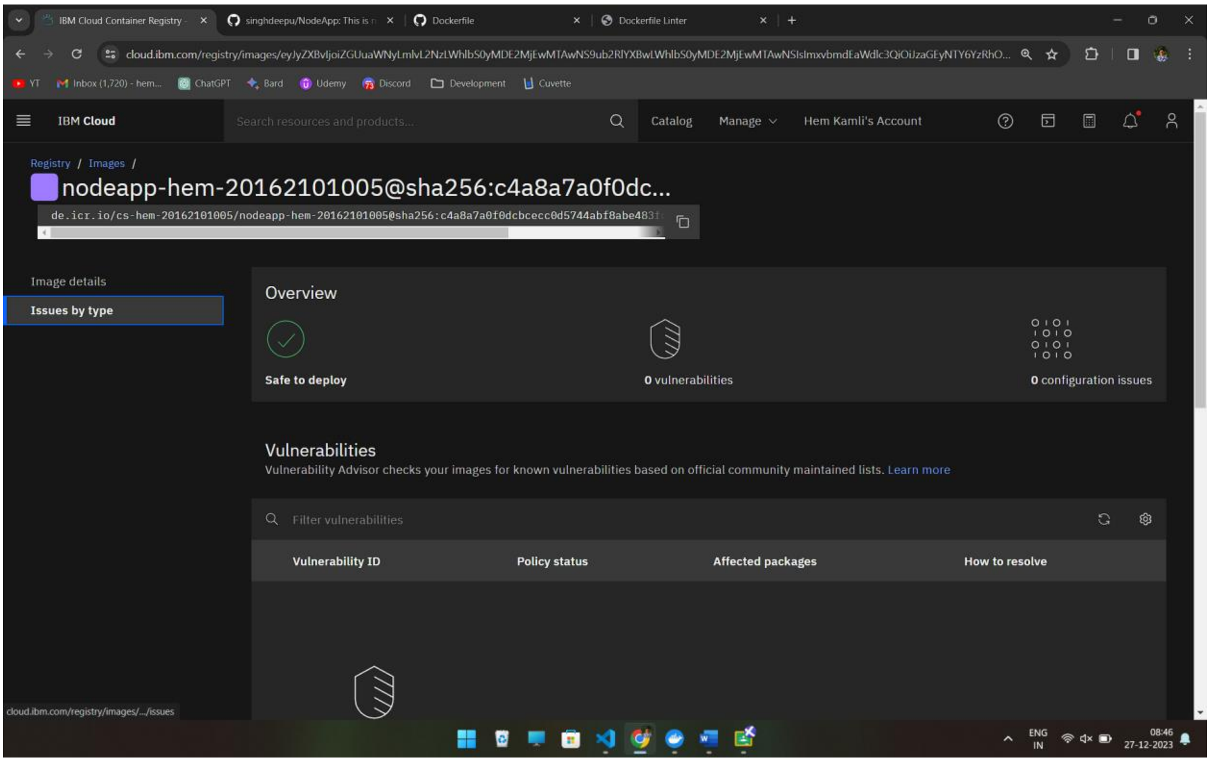
- And tag and again push the image to the repository.



```
RUN npm install express@4.17.1
COPY . .
EXPOSE 8080
CMD [ "node", "server.js" ]
```

```
PS D:\B.TECH\New folder (2)\NodeApp> docker tag nodeapp-hem-20162101005:2 de.icr.io/cs-hem-20162101005/nodeapp-hem-20162101005:2
PS D:\B.TECH\New folder (2)\NodeApp> docker push de.icr.io/cs-hem-20162101005/nodeapp-hem-20162101005:2
The push refers to repository [de.icr.io/cs-hem-20162101005/nodeapp-hem-20162101005]
1de7d79d1cae: Pushed
05fbe62fb55c: Mounted from test/nodeapp-test
b44545192106: Mounted from test/nodeapp-test
33a4142d7ccd: Pushed
25bf05053365: Pushed
3d190446e235: Mounted from test/nodeapp-test
52b814601479: Mounted from test/nodeapp-test
3421567c2c82: Layer already exists
8977df70d24f: Layer already exists
a2228fbe95fa: Layer already exists
fe1f3c52b497: Layer already exists
f1b5933fe4b5: Layer already exists
2: digest: sha256:c4a8a7a0f0dcbcecc0d5744abf8abe483fd3ed1af78944899e63ae9ce2337407 size: 2831
PS D:\B.TECH\New folder (2)\NodeApp>
```

- After that if we go back to the image on cloud registry.
- The vulnerabilities will be removed.



Registry / Images /

nodeapp-hem-20162101005@sha256:c4a8a7a0f0dc...

```
de.icr.io/cs-hem-20162101005/nodeapp-hem-20162101005@sha256:c4a8a7a0f0dcbcecc0d5744abf8abe483f
```

Overview

Safe to deploy

0 vulnerabilities

0 configuration issues

Vulnerabilities

Vulnerability Advisor checks your images for known vulnerabilities based on official community maintained lists. Learn more

Vulnerability ID	Policy status	Affected packages	How to resolve