Ganpat University | Institute of Computer Technology

॥ विद्यया समाजोत्कर्षः ॥

Name: Tushar Panchal

En.No: 21162101014

Sub: CS(Cloud Security)

Branch: CBA

Batch:71

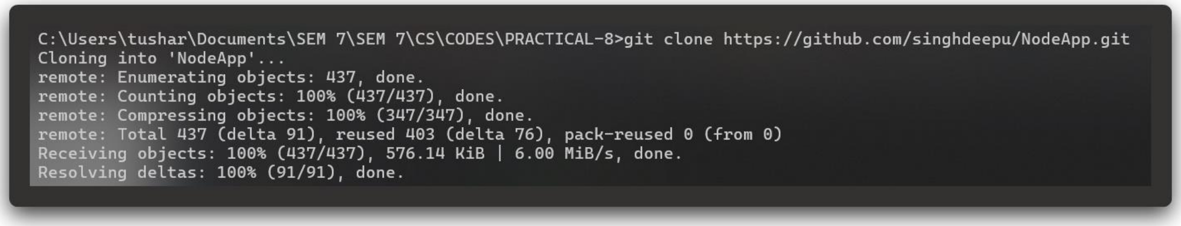
--------------------------PRACTICAL 08---------------------------

You are a DevOps engineer working for a software development company that is transitioning its applications to containerized environments using Docker. As part of the migration process, you are responsible for ensuring the security and reliability of containers and container images. Your team has developed a web application that is ready for deployment in a containerized environment.

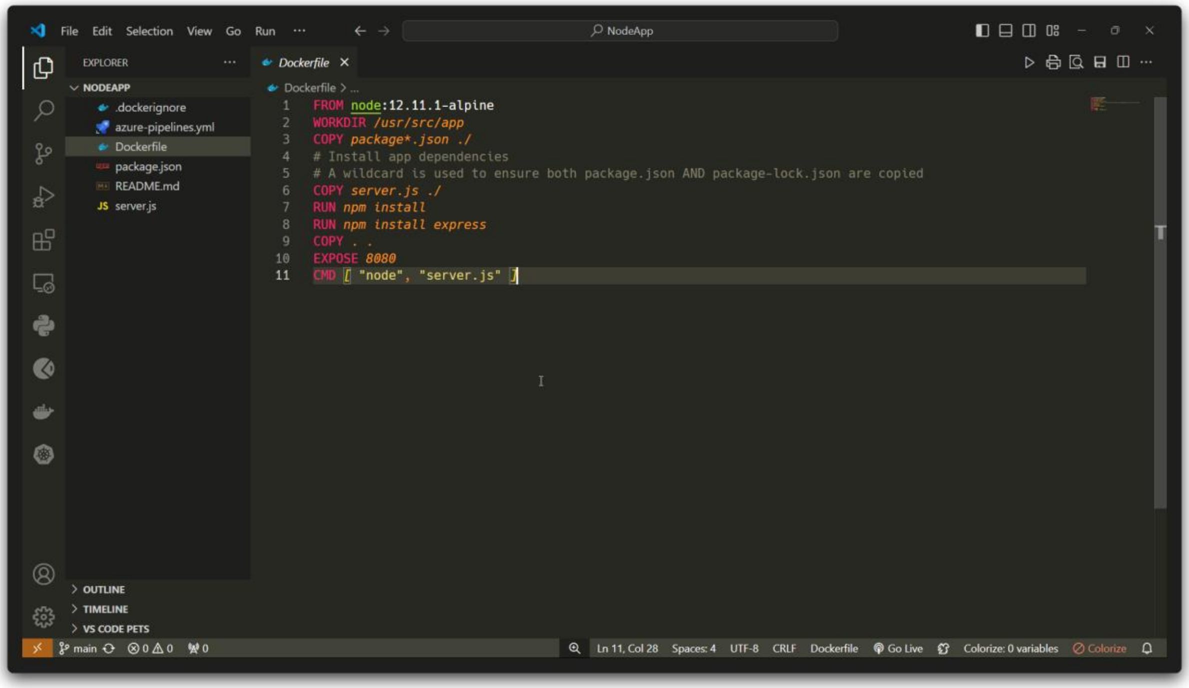
Your task is to assess the security and integrity of the container and image for the web application before it is deployed to production. Your assessment should cover potential vulnerabilities, security best practices, and ensure that the containerized application is properly configured and functional.

First we have to clone the file using cmd

git clone https://github.com/singhdeepu/NodeApp.git



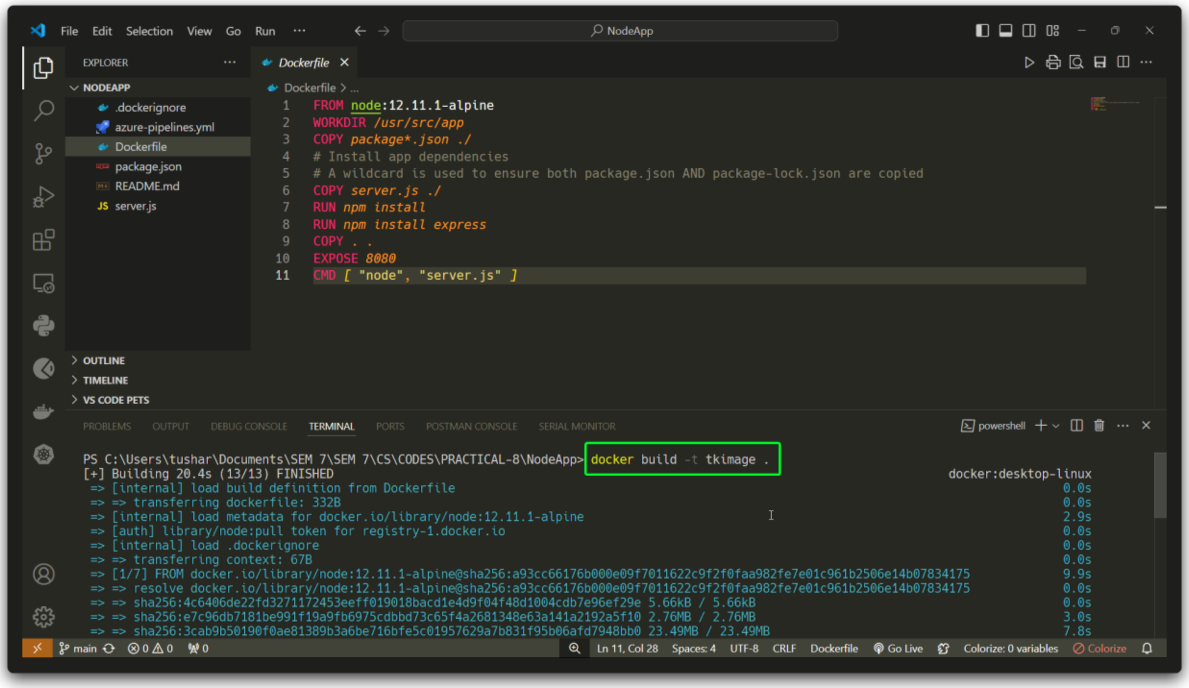
```
C:\Users\tushar\Documents\SEM 7\SEM 7\CS\CODES\PRACTICAL-8>git clone https://github.com/singhdeepu/NodeApp.git
Cloning into 'NodeApp'...
remote: Enumerating objects: 437, done.
remote: Counting objects: 100% (437/437), done.
remote: Compressing objects: 100% (347/347), done.
remote: Total 437 (delta 91), reused 403 (delta 76), pack-reused 0 (from 0)
Receiving objects: 100% (437/437), 576.14 KiB | 6.00 MiB/s, done.
Resolving deltas: 100% (91/91), done.
```



```
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" ]
```

Now build the image using cmd

Docker build -t tkimage .

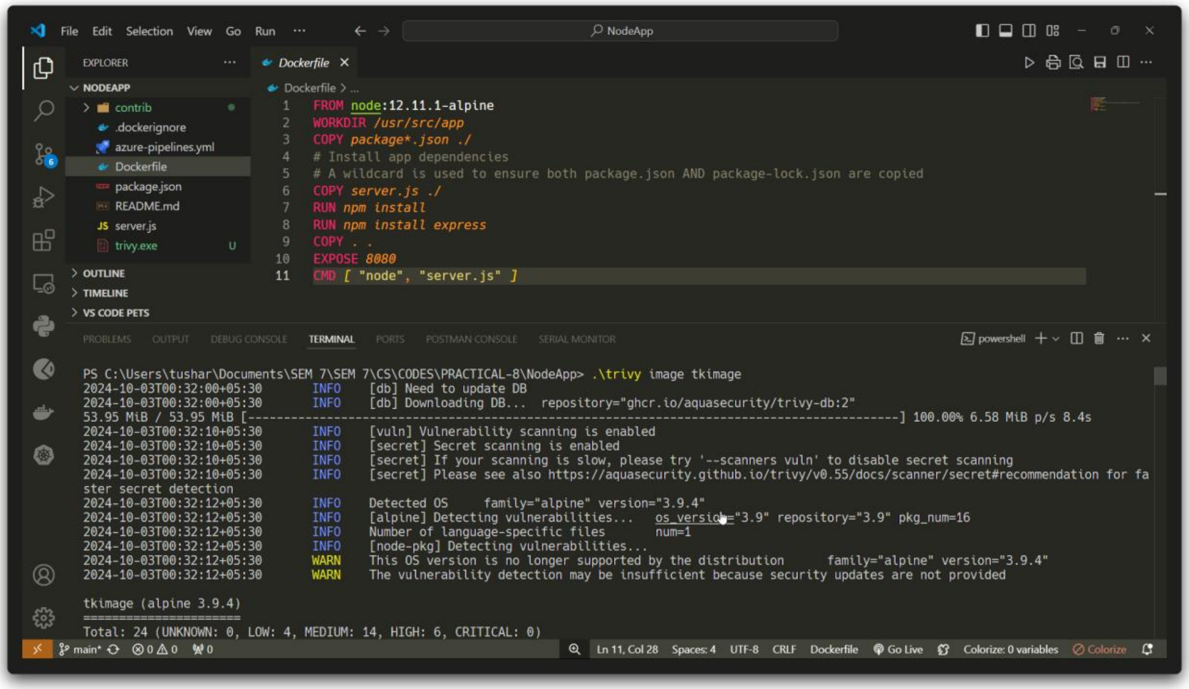

File Edit Selection View Go Run ···
NodeApp
EXPLORER
NODEAPP
.dockerignore
azure-pipelines.yml
Dockerfile
package.json
README.md
server.js
OUTLINE
TIMELINE
VS CODE PETS
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS POSTMAN CONSOLE SERIAL MONITOR
powershell
main
Ln 11, Col 28 Spaces: 4 UTF-8 CRLF Dockerfile Go Live Colorize: 0 variables Colorize

```
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" ]
```

```
PS C:\Users\tushar\Documents\SEM 7\SEM 7\CS\CODES\PRACTICAL-8\NodeApp> docker build -t tkimage .
[+] Building 20.4s (13/13) FINISHED                                                                  docker:desktop-linux
 => [internal] load build definition from Dockerfile                                                                 0.0s
 => => transferring dockerfile: 332B                                                                                 0.0s
 => [internal] load metadata for docker.io/library/node:12.11.1-alpine                                               2.9s
 => [auth] library/node:pull token for registry-1.docker.io                                                          0.0s
 => [internal] load .dockerignore                                                                                    0.0s
 => => transferring context: 67B                                                                                     0.0s
 => [1/7] FROM docker.io/library/node:12.11.1-alpine@sha256:a93cc66176b000e09f7011622c9f2f0faa982fe7e01c961b2506e14b07834175  9.9s
 => => resolve docker.io/library/node:12.11.1-alpine@sha256:a93cc66176b000e09f7011622c9f2f0faa982fe7e01c961b2506e14b07834175  0.0s
 => => sha256:4c6406de22fd3271172453eeff019018bacd1e4d9f04f48d1004cdb7e96ef29e 5.66kB / 5.66kB                       0.0s
 => => sha256:e7c96db7181be991f19a9fb6975cdbbd73c65f4a2681348e63a141a2192a5f10 2.76MB / 2.76MB                       3.0s
 => => sha256:3cab9b50190f0ae81389b3a6be716bfe5c01957629a7b831f95b06afd7948bb0 23.49MB / 23.49MB                     7.8s
```

After that you have to run the trivy command to check vulnerability

./trivy image tkimage

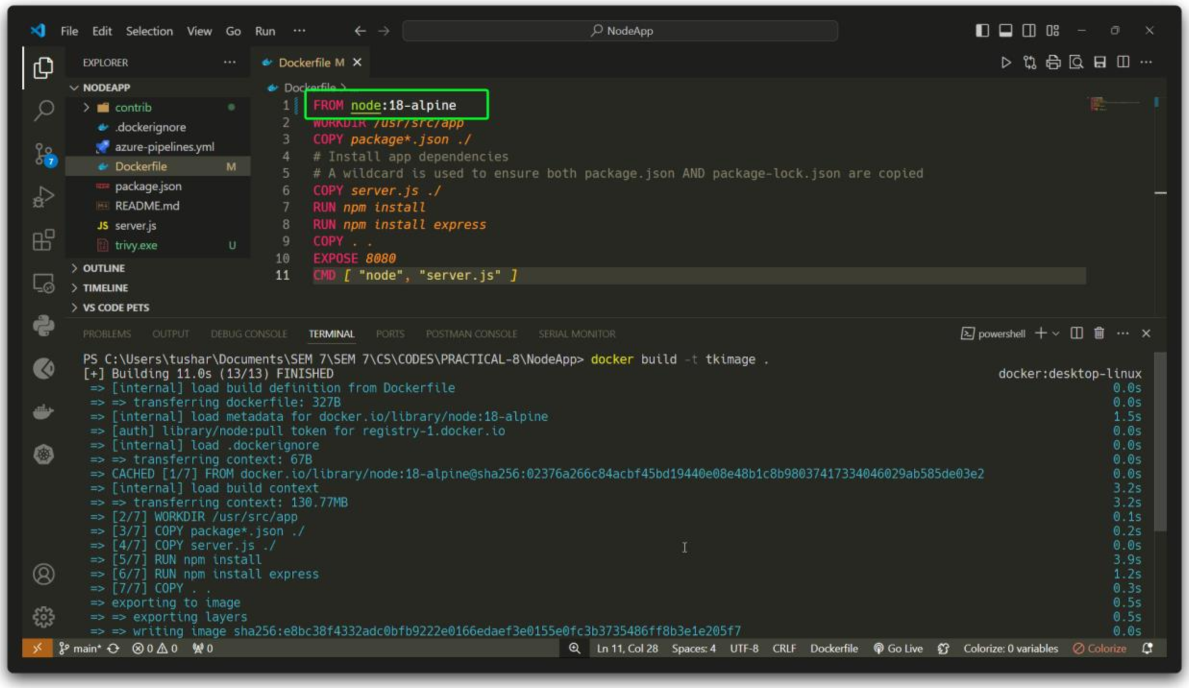

File Edit Selection View Go Run ···
NodeApp
EXPLORER
NODEAPP
contrib
.dockerignore
azure-pipelines.yml
Dockerfile
package.json
README.md
server.js
trivy.exe
U
OUTLINE
TIMELINE
VS CODE PETS
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS POSTMAN CONSOLE SERIAL MONITOR
powershell
main*
Ln 11, Col 28 Spaces: 4 UTF-8 CRLF Dockerfile Go Live Colorize: 0 variables Colorize

```
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" ]
```

```
PS C:\Users\tushar\Documents\SEM 7\SEM 7\CS\CODES\PRACTICAL-8\NodeApp> .\trivy image tkimage
2024-10-03T00:32:00+05:30       INFO    [db] Need to update DB
2024-10-03T00:32:00+05:30       INFO    [db] Downloading DB...  repository="ghcr.io/aquasecurity/trivy-db:2"
53.95 MiB / 53.95 MiB [---------------------------------------------------------------------------------------] 100.00% 6.58 MiB p/s 8.4s
2024-10-03T00:32:10+05:30       INFO    [vuln] Vulnerability scanning is enabled
2024-10-03T00:32:10+05:30       INFO    [secret] Secret scanning is enabled
2024-10-03T00:32:10+05:30       INFO    [secret] If your scanning is slow, please try '--scanners vuln' to disable secret scanning
2024-10-03T00:32:10+05:30       INFO    [secret] Please see also https://aquasecurity.github.io/trivy/v0.55/docs/scanner/secret#recommendation for fa
ster secret detection
2024-10-03T00:32:12+05:30       INFO    Detected OS     family="alpine" version="3.9.4"
2024-10-03T00:32:12+05:30       INFO    [alpine] Detecting vulnerabilities...   os_version="3.9" repository="3.9" pkg_num=16
2024-10-03T00:32:12+05:30       INFO    Number of language-specific files       num=1
2024-10-03T00:32:12+05:30       INFO    [node-pkg] Detecting vulnerabilities...
2024-10-03T00:32:12+05:30       WARN    This OS version is no longer supported by the distribution      family="alpine" version="3.9.4"
2024-10-03T00:32:12+05:30       WARN    The vulnerability detection may be insufficient because security updates are not provided

tkimage (alpine 3.9.4)
======================
Total: 24 (UNKNOWN: 0, LOW: 4, MEDIUM: 14, HIGH: 6, CRITICAL: 0)
```

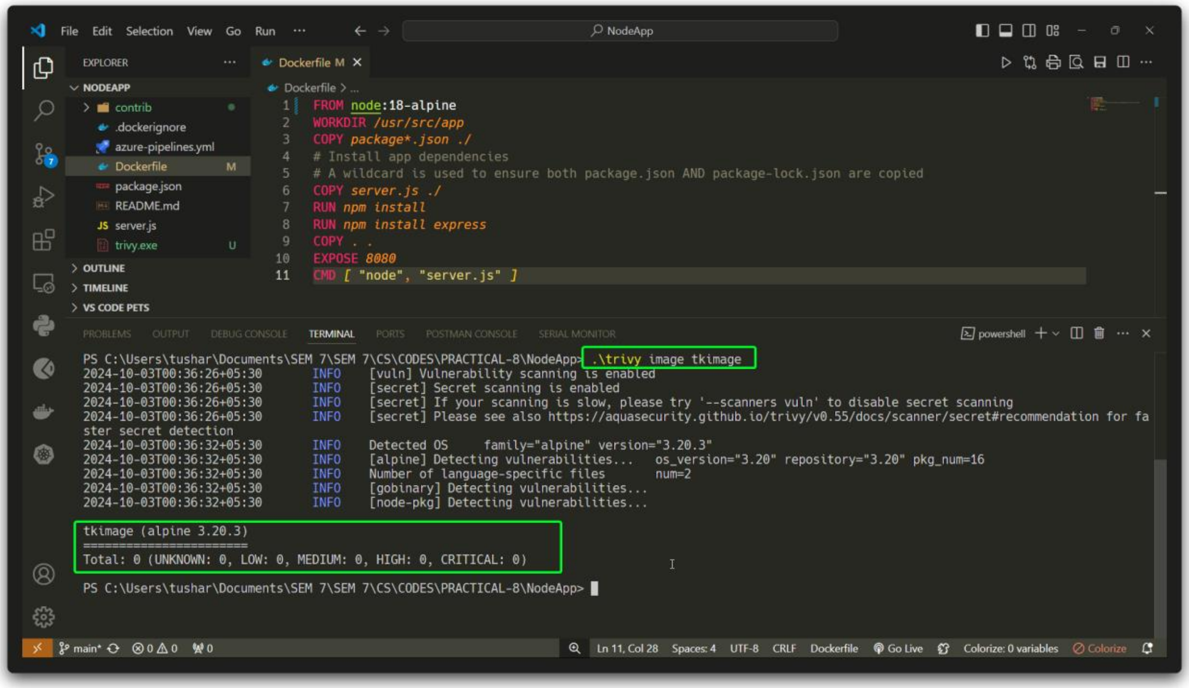
Now make changes in the version of alpine and build the image.



```
FROM node:18-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" ]
```

```
PS C:\Users\tushar\Documents\SEM 7\SEM 7\CS\CODES\PRACTICAL-8\NodeApp> docker build -t tkimage .
[+] Building 11.0s (13/13) FINISHED                                                          docker:desktop-linux
 => [internal] load build definition from Dockerfile                                                         0.0s
 => => transferring dockerfile: 327B                                                                         0.0s
 => [internal] load metadata for docker.io/library/node:18-alpine                                            1.5s
 => [auth] library/node:pull token for registry-1.docker.io                                                  0.0s
 => [internal] load .dockerignore                                                                            0.0s
 => => transferring context: 67B                                                                             0.0s
 => CACHED [1/7] FROM docker.io/library/node:18-alpine@sha256:02376a266c84acbf45bd19440e08e48b1c8b98037417334046029ab585de03e2  0.0s
 => [internal] load build context                                                                            3.2s
 => => transferring context: 130.77MB                                                                        3.2s
 => [2/7] WORKDIR /usr/src/app                                                                               0.1s
 => [3/7] COPY package*.json ./                                                                              0.2s
 => [4/7] COPY server.js ./                                                                                  0.0s
 => [5/7] RUN npm install                                                                                    3.9s
 => [6/7] RUN npm install express                                                                            1.2s
 => [7/7] COPY . .                                                                                           0.3s
 => exporting to image                                                                                       0.5s
 => => exporting layers                                                                                      0.5s
 => => writing image sha256:e8bc38f4332adc0bfb9222e0166edaef3e0155e0fc3b3735486ff8b3e1e205f7                 0.0s
```

Now you can see the zero vulnerability.



```
FROM node:18-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" ]
```

```
PS C:\Users\tushar\Documents\SEM 7\SEM 7\CS\CODES\PRACTICAL-8\NodeApp> .\trivy image tkimage
2024-10-03T00:36:26+05:30       INFO    [vuln] Vulnerability scanning is enabled
2024-10-03T00:36:26+05:30       INFO    [secret] Secret scanning is enabled
2024-10-03T00:36:26+05:30       INFO    [secret] If your scanning is slow, please try '--scanners vuln' to disable secret scanning
2024-10-03T00:36:26+05:30       INFO    [secret] Please see also https://aquasecurity.github.io/trivy/v0.55/docs/scanner/secret#recommendation for fa
ster secret detection
2024-10-03T00:36:32+05:30       INFO    Detected OS     family="alpine" version="3.20.3"
2024-10-03T00:36:32+05:30       INFO    [alpine] Detecting vulnerabilities...   os_version="3.20" repository="3.20" pkg_num=16
2024-10-03T00:36:32+05:30       INFO    Number of language-specific files       num=2
2024-10-03T00:36:32+05:30       INFO    [gobinary] Detecting vulnerabilities...
2024-10-03T00:36:32+05:30       INFO    [node-pkg] Detecting vulnerabilities...

tkimage (alpine 3.20.3)
=======================
Total: 0 (UNKNOWN: 0, LOW: 0, MEDIUM: 0, HIGH: 0, CRITICAL: 0)

PS C:\Users\tushar\Documents\SEM 7\SEM 7\CS\CODES\PRACTICAL-8\NodeApp>
```