

Bugs & Solutions

Updated Project--

<https://github.com/19bcs22Ravi/CosmoCloud-Task-2>

1. Backend Application Issues

1.1 FastAPI Startup Command

Bug: Incorrect FastAPI startup command in Dockerfile

```
CMD ["fastapi", "dev", "main.py", "--host", "0.0.0.0", "--port", "8080"]
```

Problem:

- FastAPI doesn't have a direct command-line interface
- The application would fail to start
- The command structure is invalid for ASGI applications

Solution:

```
CMD ["uvicorn", "main:app", "--host", "0.0.0.0", "--port", "8080"]
```

- Used uvicorn as the ASGI server
- Correctly referenced the FastAPI app instance
- Properly configured host and port parameters

2. Kubernetes Service Configuration

2.1 Port Configuration

Bug: Inconsistent port configurations across services

```
# Original backend service
ports:
  - port: 8000
    targetPort: 8000
    nodePort: 31000
```

Problem:

- Backend service port (8000) didn't match container port (8080)

- Caused connection failures between services
- Made external access impossible
- Created confusion in service discovery

Solution:

```
# Fixed backend service
ports:
  - port: 8080
    targetPort: 8080
    nodePort: 31000
```

- Aligned all ports with container specifications
- Ensured consistent port mapping
- Maintained valid NodePort range (30000-32767)

2.2 Label Selector Inconsistencies

Bug: Mismatched label selectors

```
# Original deployment
selector:
  matchLabels:
    app: app
# Original service
selector:
  app: backend-svc
```

Problem:

- Services couldn't find their target pods
- Traffic routing failed
- Pods were not properly associated with services
- Load balancing will not work

Solution:

```
# Fixed deployment
selector:
  matchLabels:
    app: backend-app
# Fixed service
selector:
  app: backend-app
```

- Standardized labels across all components

- Ensured consistent service-to-pod routing
- Implemented proper Kubernetes label practices

3. Redis Configuration Issues

3.1 Redis Image and Port Configuration

Bug: Invalid Redis configuration

```
# Original Redis deployment
containers:
- name: redis:alppine
  image: redis
  ports:
  - containerPort: 6377
```

Problem:

- Typo in Redis image name
- Incorrect port number (6377 instead of 6379)
- Inconsistent container naming
- Missing image tag specification

Solution:

```
# Fixed Redis deployment
containers:
- name: redis
  image: redis:alpine
  ports:
  - containerPort: 6379
```

- Used correct image name and tag
- Set standard Redis port (6379)
- Implemented consistent naming
- Added specific image version

3.2 Redis Connection Configuration

Bug: Missing Redis connection configuration in backend

```
# Original: No environment variables defined
```

Problem:

- Backend couldn't connect to Redis

- No service discovery configuration
- Missing connection string
- No way to persist hit counts

Solution:

```
# Added to backend deployment
env:
- name: REDIS_URI
  value: "redis://redis-svc:6379"
```

- Added proper Redis connection URI
- Used Kubernetes service discovery
- Configured correct port
- Enabled persistent hit counter

4. Deployment Management Issues

4.1 Image Pull Configuration

Bug: Undefined image pull behavior

```
# Original: No pull policy specified
```

Problem:

- Kubernetes might try to pull from remote registry
- Local images might be ignored
- Unnecessary network requests
- Potential deployment failures

Solution:

```
# Added to deployments
imagePullPolicy: IfNotPresent
```

- Prioritized local images
- Reduced unnecessary pulls
- Improved deployment reliability
- Optimized resource usage

5. Testing and Verification

Command Reference

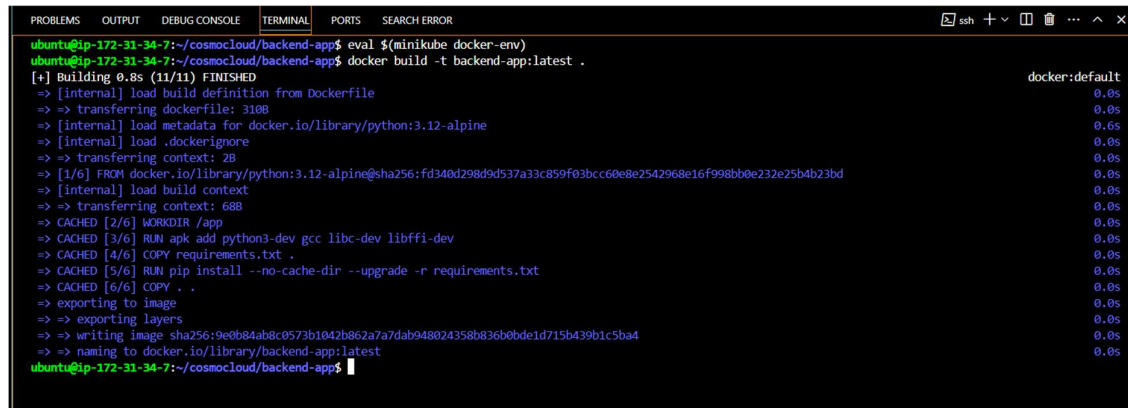
```
# Check pod status
kubectl get pods

# Verify services
kubectl get svc

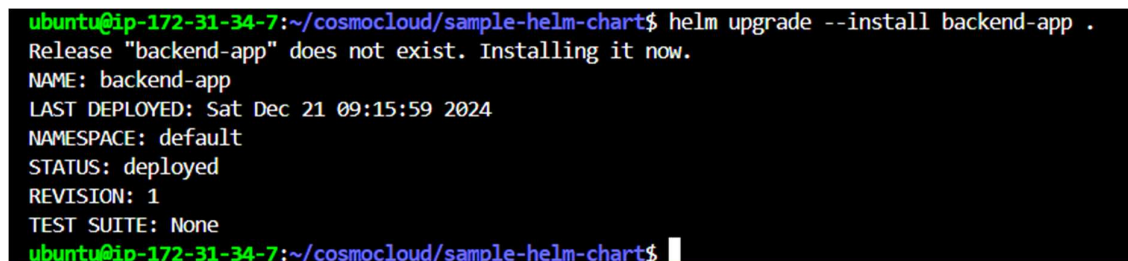
# Test API (replace with actual minikube IP)
curl http://<minikube-ip>:31000/

# Check Redis data
kubectl exec -it <redis-pod-name> -- redis-cli
```

Snapshots:



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS SEARCH ERROR
ubuntu@ip-172-31-34-7:~/cosmocloud/backend-app$ eval $(minikube docker-env)
ubuntu@ip-172-31-34-7:~/cosmocloud/backend-app$ docker build -t backend-app:latest .
[+] Building 0.8s (11/11) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 310B
=> [internal] load metadata for docker.io/library/python:3.12-alpine
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [1/6] FROM docker.io/library/python:3.12-alpine@sha256:fd340d298d9d537a33c859f03bcc60e8e2542968e16f998bb0e232e25b4b23bd
=> [internal] load build context
=> => transferring context: 68B
=> CACHED [2/6] WORKDIR /app
=> CACHED [3/6] RUN apk add python3-dev gcc libc-dev libffi-dev
=> CACHED [4/6] COPY requirements.txt .
=> CACHED [5/6] RUN pip install --no-cache-dir --upgrade -r requirements.txt
=> CACHED [6/6] COPY . .
=> exporting to image
=> => exporting layers
=> => writing image sha256:9e0b84ab8c0573b1042b062a7a7dab948024358b836b0bde1d715b439b1c5ba4
=> => naming to docker.io/library/backend-app:latest
ubuntu@ip-172-31-34-7:~/cosmocloud/backend-app$
```



```
ubuntu@ip-172-31-34-7:~/cosmocloud/sample-helm-chart$ helm upgrade --install backend-app .
Release "backend-app" does not exist. Installing it now.
NAME: backend-app
LAST DEPLOYED: Sat Dec 21 09:15:59 2024
NAMESPACE: default
STATUS: deployed
REVISION: 1
TEST SUITE: None
ubuntu@ip-172-31-34-7:~/cosmocloud/sample-helm-chart$
```

PROBLEMS	OUTPUT	DEBUG CONSOLE	TERMINAL	PORTS	SEARCH ERROR
<pre>ubuntu@ip-172-31-36-169:~/cosmocloud/sample-helm-chart\$ kubectl get pods NAME READY STATUS RESTARTS AGE backend-app-64499f8674-2hk85 1/1 Running 0 13s redis-6bd69b7674-n5wbg 1/1 Running 0 13s ubuntu@ip-172-31-36-169:~/cosmocloud/sample-helm-chart\$ kubectl get svc NAME TYPE CLUSTER-IP EXTERNAL-IP PORT(S) AGE backend-svc NodePort 10.101.138.4 <none> 8080:31000/TCP 27s kubernetes ClusterIP 10.96.0.1 <none> 443/TCP 30m redis-svc ClusterIP 10.107.151.183 <none> 6379/TCP 27s ubuntu@ip-172-31-36-169:~/cosmocloud/sample-helm-chart\$ kubectl get deployment NAME READY UP-TO-DATE AVAILABLE AGE backend-app 1/1 1 1 42s redis 1/1 1 1 42s ubuntu@ip-172-31-36-169:~/cosmocloud/sample-helm-chart\$ minikube ip 192.168.49.2 ubuntu@ip-172-31-36-169:~/cosmocloud/sample-helm-chart\$</pre>					

PROBLEMS	OUTPUT	DEBUG CONSOLE	TERMINAL	PORTS	SEARCH ERROR
<pre>ubuntu@ip-172-31-36-169:~/cosmocloud/sample-helm-chart\$ kubectl get pods NAME READY STATUS RESTARTS AGE backend-app-64499f8674-2hk85 1/1 Running 0 13s redis-6bd69b7674-n5wbg 1/1 Running 0 13s ubuntu@ip-172-31-36-169:~/cosmocloud/sample-helm-chart\$ kubectl get svc NAME TYPE CLUSTER-IP EXTERNAL-IP PORT(S) AGE backend-svc NodePort 10.101.138.4 <none> 8080:31000/TCP 27s kubernetes ClusterIP 10.96.0.1 <none> 443/TCP 30m redis-svc ClusterIP 10.107.151.183 <none> 6379/TCP 27s ubuntu@ip-172-31-36-169:~/cosmocloud/sample-helm-chart\$ kubectl get deployment NAME READY UP-TO-DATE AVAILABLE AGE backend-app 1/1 1 1 42s redis 1/1 1 1 42s ubuntu@ip-172-31-36-169:~/cosmocloud/sample-helm-chart\$ minikube ip 192.168.49.2 ubuntu@ip-172-31-36-169:~/cosmocloud/sample-helm-chart\$ curl http://192.168.49.2:31000 {"hits":1}ubuntu@ip-172-31-36-169:~/cosmocloud/sample-helm-chart\$ curl http://192.168.49.2:31000 {"hits":2}ubuntu@ip-172-31-36-169:~/cosmocloud/sample-helm-chart\$ curl http://192.168.49.2:31000 {"hits":3}ubuntu@ip-172-31-36-169:~/cosmocloud/sample-helm-chart\$ curl http://192.168.49.2:31000 {"hits":4}ubuntu@ip-172-31-36-169:~/cosmocloud/sample-helm-chart\$ curl http://192.168.49.2:31000 {"hits":5}ubuntu@ip-172-31-36-169:~/cosmocloud/sample-helm-chart\$</pre>					

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS SEARCH ERROR
ubuntu@ip-172-31-36-169:~/cosmocloud/sample-helm-chart$ kubectl get pods
NAME READY STATUS RESTARTS AGE
backend-app-64499f8674-2hk85 1/1 Running 0 13s
redis-6bd69b7674-n5wbg 1/1 Running 0 13s
ubuntu@ip-172-31-36-169:~/cosmocloud/sample-helm-chart$ kubectl get svc
NAME TYPE CLUSTER-IP EXTERNAL-IP PORT(S) AGE
backend-svc NodePort 10.101.138.4 <none> 8080:31000/TCP 27s
kubernetes ClusterIP 10.96.0.1 <none> 443/TCP 30m
redis-svc ClusterIP 10.107.151.183 <none> 6379/TCP 27s
ubuntu@ip-172-31-36-169:~/cosmocloud/sample-helm-chart$ kubectl get deployment
NAME READY UP-TO-DATE AVAILABLE AGE
backend-app 1/1 1 1 42s
redis 1/1 1 1 42s
ubuntu@ip-172-31-36-169:~/cosmocloud/sample-helm-chart$ minikube ip
192.168.49.2
ubuntu@ip-172-31-36-169:~/cosmocloud/sample-helm-chart$ curl http://192.168.49.2:31000
{"hits":1}
ubuntu@ip-172-31-36-169:~/cosmocloud/sample-helm-chart$ curl http://192.168.49.2:31000
{"hits":2}
ubuntu@ip-172-31-36-169:~/cosmocloud/sample-helm-chart$ curl http://192.168.49.2:31000
{"hits":3}
ubuntu@ip-172-31-36-169:~/cosmocloud/sample-helm-chart$ curl http://192.168.49.2:31000
{"hits":4}
ubuntu@ip-172-31-36-169:~/cosmocloud/sample-helm-chart$ curl http://192.168.49.2:31000
{"hits":5}
ubuntu@ip-172-31-36-169:~/cosmocloud/sample-helm-chart$ kubectl exec -it redis-6bd69b7674-n5wbg -- redis-cli
127.0.0.1:6379> GET hits
"5"
127.0.0.1:6379> |
```