

Matias Roje - Kubernetes exam

<https://github.com/MatiasRoje/datascientest-kubernetes-exam.git>

For the exam, I created a custom FastAPI Docker image due to the provided image being too heavy for the Datascientest Ubuntu server. I then used the Kompose tool to convert the `docker-compose.yaml` file into an initial Kubernetes manifest, which I further configured manually.

Although Helm and Kustomize are not explicitly showcased, they would be beneficial for implementing different namespaces and data configurations, if needed.

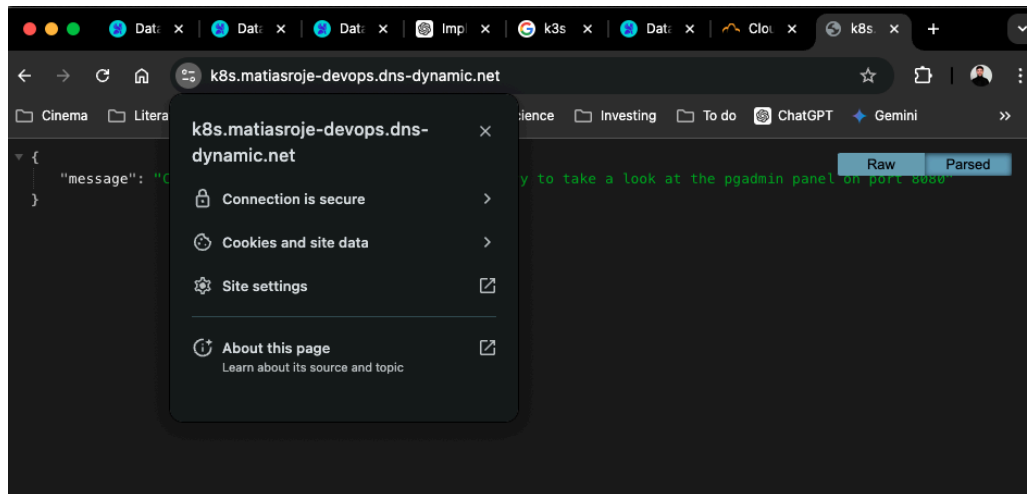
Here's a summary of the setup and observations:

- **FastAPI Server:** Accessible via HTTPS from the browser, functioning correctly and connected to the database.
- **PgAdmin:** Accessible locally and functioning as expected. Screenshots are available for verification.
- **Data Backup:** Backups are stored in an S3 bucket. The backup process follows the [official k3s documentation for SQLite databases](#). For etcd Datastore, the [native k3s solution could be used with the following command](#):

```
`bash
k3s etcd-snapshot save \
  --s3 \
  --s3-bucket=<S3-BUCKET-NAME> \
  --s3-access-key=<S3-ACCESS-KEY> \
  --s3-secret-key=<S3-SECRET-KEY>`
```

I opted for a custom backup script to avoid restarting the cluster and to demonstrate a more creative solution.

App accessible from the browser via HTTPS



Regular Kubernetes manifest running

```
ubuntu@ip-172-31-23-166:~/datascientest-kubernetes-exam/YAML-STANDARD$ kubectl get all -n standard
NAME                                READY    STATUS    RESTARTS   AGE
pod/db-0                            1/1      Running   1 (53m ago) 60m
pod/fastapi-6b8ddc5cdf-449hg        1/1      Running   4 (52m ago) 60m
pod/fastapi-6b8ddc5cdf-czhwp        1/1      Running   4 (52m ago) 60m
pod/fastapi-6b8ddc5cdf-f6t87        1/1      Running   4 (52m ago) 60m
pod/pgadmin-6b8fc6554c-hwgq2        1/1      Running   1 (53m ago) 60m

NAME                TYPE          CLUSTER-IP    EXTERNAL-IP    PORT(S)          AGE
service/db           ClusterIP     10.43.175.25  <none>         5432/TCP         60m
service/fastapi      ClusterIP     10.43.8.48    <none>         5000/TCP         60m
service/pgadmin      NodePort      10.43.209.70  <none>         80:30080/TCP     60m

NAME                READY    UP-TO-DATE    AVAILABLE    AGE
deployment.apps/fastapi  3/3      3              3            60m
deployment.apps/pgadmin  1/1      1              1            60m

NAME                DESIRED    CURRENT    READY    AGE
replicaset.apps/fastapi-6b8ddc5cdf  3          3          3        60m
replicaset.apps/pgadmin-6b8fc6554c  1          1          1        60m

NAME                READY    AGE
statefulset.apps/db  1/1      60m

NAME                REFERENCE          TARGETS          MINPODS    MAXPODS
DS    REPLICAS    AGE
horizontalpodautoscaler.autoscaling/fastapi-hpa  Deployment/fastapi  cpu: <unknown>/70%  3          6

ubuntu@ip-172-31-23-166:~/datascientest-kubernetes-exam/YAML-STANDARD$ kubectl get ingress -n standard
NAME                CLASS    HOSTS                                ADDRESS    PORTS    AGE
fastapi-ingress     traefik  k8s.matiasroje-devops.dns-dynamic.net  172.31.23.166  80      65m
```

Helm chart running

```

ubuntu@ip-172-31-23-166:~/datascientest-kubernetes-exam/HELM$ kubectl get all -n helm
NAME                                READY    STATUS    RESTARTS   AGE
pod/db-0                            1/1      Running   0           3m32s
pod/fastapi-fd8dbbcd5-2vhwk         1/1      Running   2 (3m16s ago)  3m31s
pod/fastapi-fd8dbbcd5-8m7sf         1/1      Running   2 (3m17s ago)  3m32s
pod/fastapi-fd8dbbcd5-s8qkx         1/1      Running   2 (3m14s ago)  3m31s
pod/pgadmin-558c4b4799-4hb64        1/1      Running   0           3m32s

NAME                                TYPE      CLUSTER-IP    EXTERNAL-IP    PORT(S)        AGE
service/db                          ClusterIP    10.43.176.188  <none>          5432/TCP        3m32s
service/fastapi                     ClusterIP    10.43.4.3      <none>          5000/TCP        3m32s
service/pgadmin                     NodePort     10.43.178.180  <none>          80:30080/TCP    3m32s

NAME                                READY    UP-TO-DATE    AVAILABLE    AGE
deployment.apps/fastapi              3/3      3              3            3m32s
deployment.apps/pgadmin              1/1      1              1            3m32s

NAME                                DESIRED    CURRENT    READY    AGE
replicaset.apps/fastapi-fd8dbbcd5    3          3          3        3m32s
replicaset.apps/pgadmin-558c4b4799    1          1          1        3m32s

NAME                                READY    AGE
statefulset.apps/db                 1/1      3m32s

NAME                                REFERENCE    TARGETS    MINPODS    MAXPODS
DS REPLICAS AGE
horizontalpodautoscaler.autoscaling/fastapi-hpa  Deployment/fastapi  cpu: <unknown>/70%  3        6

ubuntu@ip-172-31-23-166:~/datascientest-kubernetes-exam/HELM$ kubectl get ingress -n helm
NAME                                CLASS    HOSTS                                ADDRESS    PORTS    AGE
fastapi-ingress                     traefik  k8s.matiasroje-devops.dns-dynamic.net  172.31.23.166  80       3m46s

```

Kustomized deployment running

```

ubuntu@ip-172-31-23-166:~/datascientest-kubernetes-exam/k8s/KUSTOMIZE/overlays/kustomize$ kubectl get all -n kustomize
NAME                                READY    STATUS    RESTARTS   AGE
pod/db-0                            1/1      Running   0           2m6s
pod/fastapi-6b8ddc5cdf-8jr72        1/1      Running   1           2m6s
pod/fastapi-6b8ddc5cdf-b8r7t        1/1      Running   2 (113s ago)  2m6s
pod/fastapi-6b8ddc5cdf-xxtkr        1/1      Running   2 (111s ago)  2m6s
pod/pgadmin-7cc64d6d59-klcst        1/1      Running   0           2m6s

NAME                                TYPE      CLUSTER-IP    EXTERNAL-IP    PORT(S)        AGE
service/db                          ClusterIP    10.43.157.84   <none>          5432/TCP        2m7s
service/fastapi                     ClusterIP    10.43.9.116    <none>          5000/TCP        2m7s
service/pgadmin                     NodePort     10.43.41.235   <none>          80:30080/TCP    2m7s

NAME                                READY    UP-TO-DATE    AVAILABLE    AGE
deployment.apps/fastapi              3/3      3              3            2m6s
deployment.apps/pgadmin              1/1      1              1            2m6s

NAME                                DESIRED    CURRENT    READY    AGE
replicaset.apps/fastapi-6b8ddc5cdf    3          3          3        2m6s
replicaset.apps/pgadmin-7cc64d6d59    1          1          1        2m6s

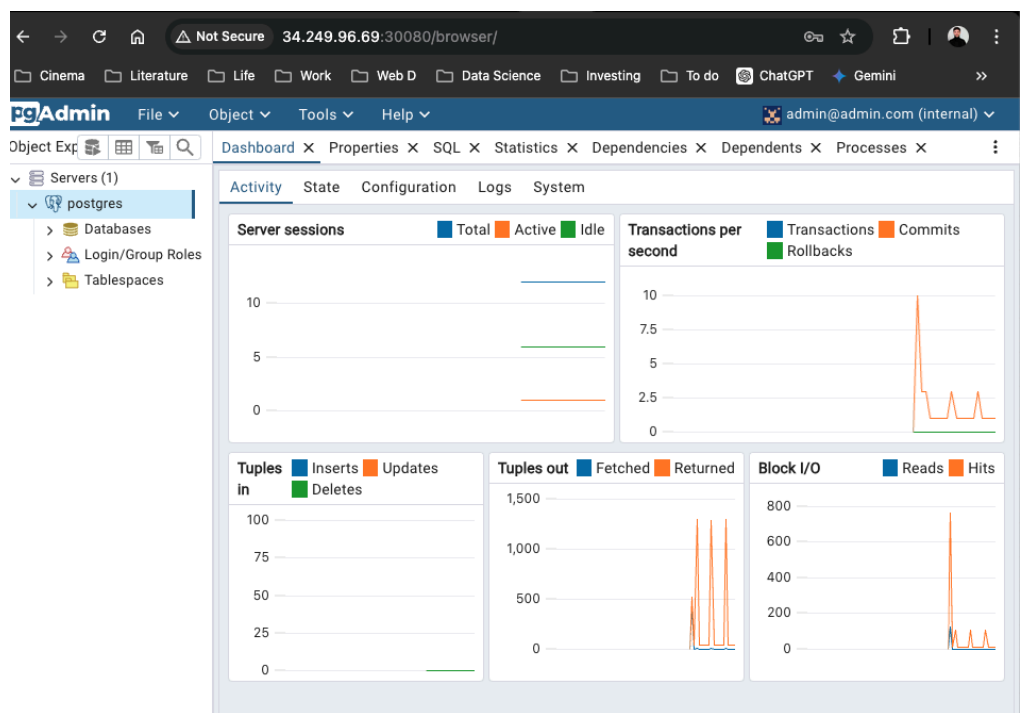
NAME                                READY    AGE
statefulset.apps/db                 1/1      2m6s

NAME                                REFERENCE    TARGETS    MINPODS    MAXPODS
DS REPLICAS AGE
horizontalpodautoscaler.autoscaling/fastapi-hpa  Deployment/fastapi  cpu: <unknown>/70%  3        6

ubuntu@ip-172-31-23-166:~/datascientest-kubernetes-exam/k8s/KUSTOMIZE/overlays/kustomize$ kubectl get ingress -n kustomize
NAME                                CLASS    HOSTS                                ADDRESS    PORTS    AGE
fastapi-ingress                     traefik  k8s.matiasroje-devops.dns-dynamic.net  172.31.23.166  80       2m19s

```

PgAdmin working locally



Data stored in S3

Amazon S3 > Buckets > datascientest-kubernetes-exam-etcd-backup

datascientest-kubernetes-exam-etcd-backup [Info](#)

[Objects](#) | [Properties](#) | [Permissions](#) | [Metrics](#) | [Management](#) | [Access Points](#)

Objects (1) [Info](#) [Refresh](#) [Copy S3 URI](#) [Copy URL](#) [Download](#) [Open](#) [Delete](#) [Actions](#) [Create](#)

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant the

<input type="checkbox"/>	Name	Type	Last modified	Size
<input type="checkbox"/>	k3s_backup_31082024.zip	zip	September 1, 2024, 01:40:41 (UTC+02:00)	2.1 MB

- **Sensitive Information:** I have included sensitive information in the Git repository, encoded for the purpose of this exam. In a real-world scenario, I would use AWS Secrets Manager or a similar service for secure handling of sensitive data. That assignment is out of the scope of the exam.

Thank you for reviewing my work.