Kubernetes Cluster Micro-service Task

This task is centred around the knowledge of micro-services in containers. The goal of this task is understanding how communication, security and visibility play a key role in ensuring a safe a reliable system. We will be managing our container infrastructure through orchestration. Our aim is to have a functioning restful micro-service structure that allows us to query for results.

Specification

We would like you to create a system that will allow a command line terminal to request a light weight message. A simple curl GET request will be used. Since we want to build a micro-service design, the message will contain strings that will identify each service in the cluster. You have the choice of constructing the messaging for each service provided that it's unique enough to identify each service. We require you to build two restful services, one public facing service and one private service. The public service will interact with the outside world and will be responsible for receiving our curl requests. The private service will **ONLY** accept requests from the public service. The servers will be built using java and the frameworks of your choice to host restful servers. Some examples of frameworks include but not limited to: Java Jersey, Spring Framework, Play Framework, Quarkus, etc. Messaging between micro-services and the terminal will use JSON as the standard.

Authentication will be supplied in curl requests initiated by the terminal. JWT will be used as or token standard and you will be responsible for providing a secret that will be used towards symmetrical encryption when we sign the payload. We will be using the `HS256` algorithm for signing the token payloads. The payload will contain 3 required items:

a. A time when the JWT was issued
b. A time when the JWT will expire
c. A list of scopes that define which micro-service is allowed to be accessed.

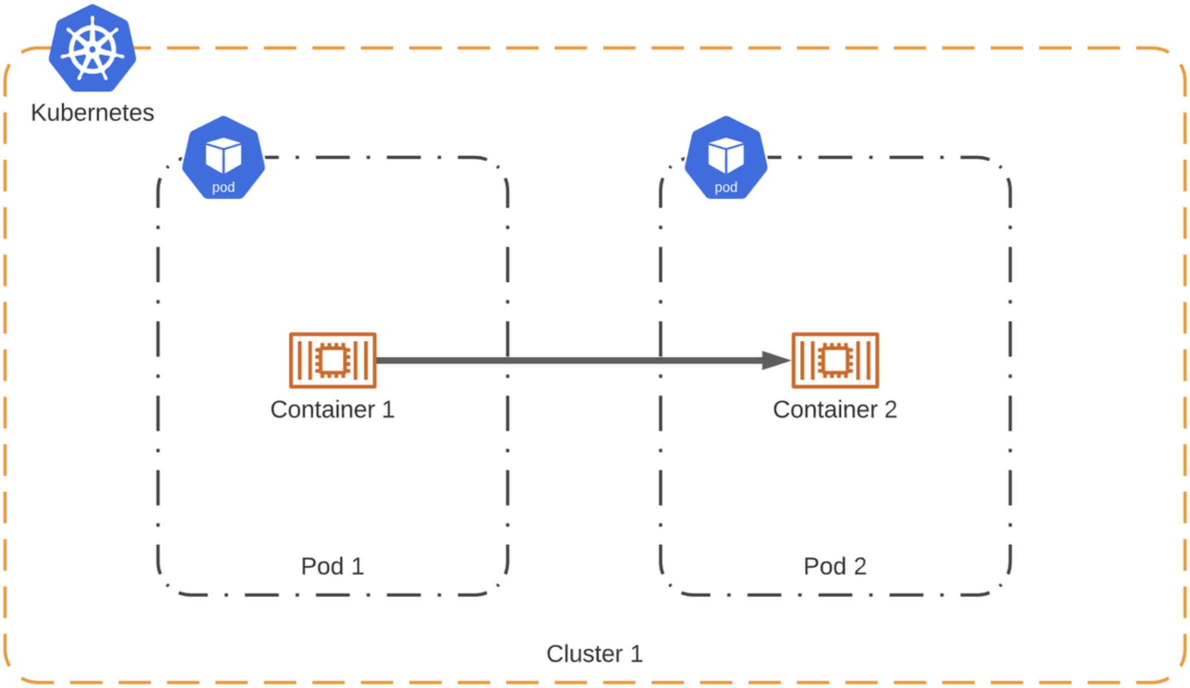
```
{
  "iat": 1516239022.
  "exp": 1517239022,
  "scope": PUBLIC_SERVICE, PRIVATE_SERVICE
}
```

The services will reject requests that

a. have missing payload entries
b. provide an expired JWT token
c. incorrect scope definitions
d. Signatures do not match

Messages returned from the public service will depend on the scopes provided in the JWT token. If `PUBLIC_SERVICE` is the only scope entry in the token for the request, the response should contain messaging from the public service. If `PRIVATE_SERVICE` is the only scope entry in the token for the request, the response should contain messaging from the private service. If both scopes are provided, the response should contain messaging from both public and private services.

Our project will be built utilizing docker as our container and orchestrated under Kubernetes. You will be creating two container images, one for each service. Each service will be deployed in a pod. You will have to configure the environment such that public access is allowed on the container that runs the public service and private access between pods on the container the runs the private service. You should be able to manage the environments with kubernetes; terminate containers, spawn new containers, etc.



Submission

The best way to share access to your project is with a cloud based repository like Github, bitbucket, or Gitlab. You may also provide an archive of the source and infrastructure code through email or your choice of a cloud based file storage such as shared google drive directory, for example. Please include a Readme that outlines the restful endpoints, both public and private. The readme should also include details on configuring a local environment to run the project. You can assume that the local environment has Kubernetes and Docker installed. We will be running the project in our local systems and evaluating the solution.

Questions

In a separate document please answer the following questions:

1. What ways can you have visibility into the services?
2. How can you ensure the services are behaving as expected?
3. If we require our services to scale, what modification would you make to support horizontal scaling.