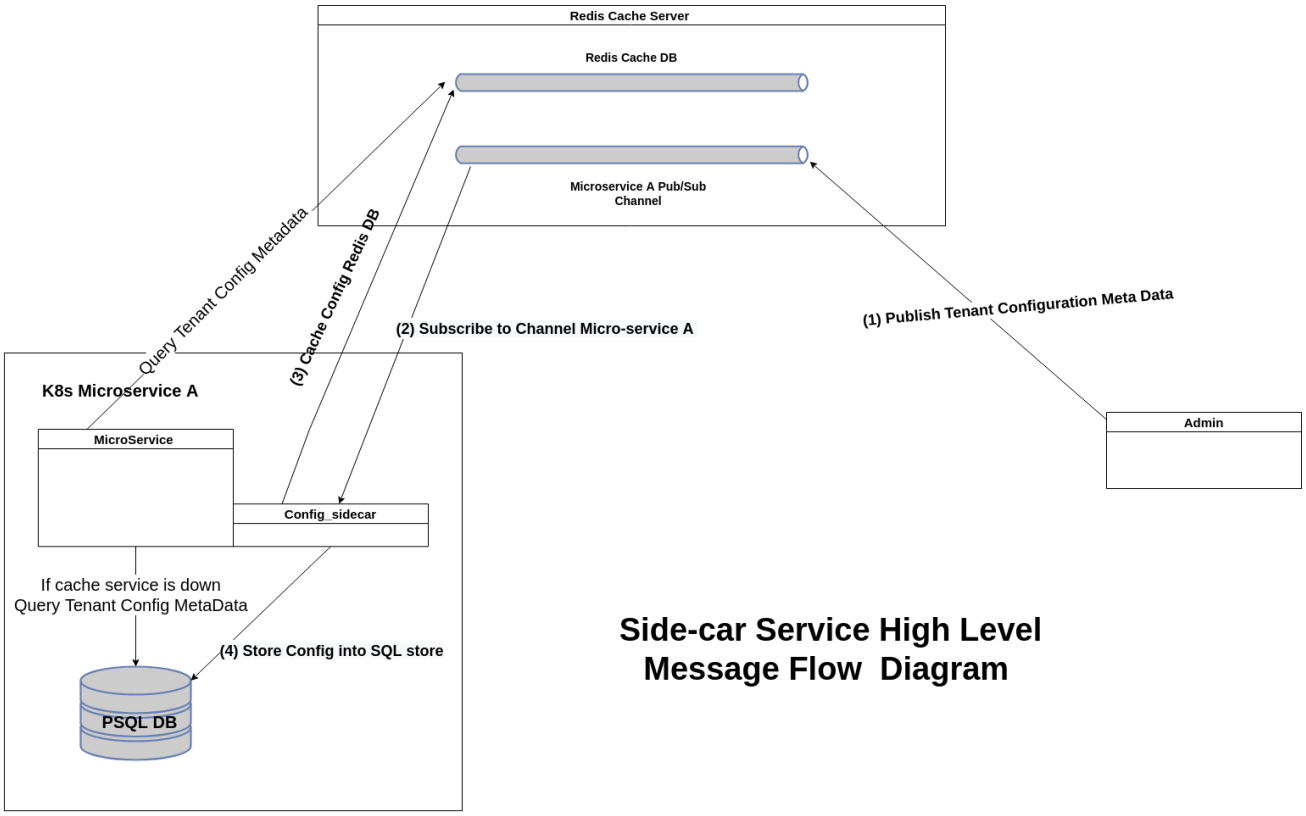
Config-sidecar

It’s k8s sidecar service which will be deployed alongside the primary k8s pod app

- For its initial version the app is built to listen to any tenant metadata configuration changes and persists its corresponding sql and temporary data stores



This means this particular app should be built for each microservice

- Their will be one redis message channel
- The sidecar service will create this channel if it doesn’t exist
- Admin module should know each MS config message channel, and it should publish configuration changes into this channels
- the app should build built for each micro services and should be deployed as a k8s sidecar service to each MS pod

How to build deployable Image for any MS

In the app source their is a reusable docker file to help build a deployable image for any MS.

Prerequisites

The following steps should be taken before building any deployable docker image

- each deployment of sidecar service needs to point to its-own postgres instance
- Postgres version 11 and above should be used
- A Database should be created , we should also create a db user and assign the new user to be

the owner of the database

- the user should have enough permission to connect remotely and perform basic database operations
- the database IP , the database name and credentials should be passed in the docker file as an env variable

The following are mandatory environment `` ENV SERVICE_NAME=*The microservice name OR k8s pod id where this sidecar services will be deployed*

ENV DB_HOST=*Postgres SQL server IP or Host*

ENV DB_PORT=*Postgres SQL server port*

ENV DB_NAME=*Postgres SQL Database Name (Note that the specified database needs to be created before running any image of this docker file*

ENV DB_USERNAME=*Postgres SQL connection username (Note that the specified DB user needs to be created before running any image of this docker file*

ENV DB_PASSWORD=*Postgres SQL connection password*

ENV REDIS_HOST=*Redis Server IP or Host*

ENV REDIS_PORT=*Redis Server Port`` *

Follow This Simple Steps To Build an Image

clone and/or checkout master branch https://git.sepa-cyber.com/microservices/wallet/config-sidecar.git

1) Go to the root source folder → Open `DockerFile txt` and change the environment parameters

2) To build an image just execute

`docker build -t config-sidecar-MSName:latest .` MSName - being the microservice module name or k8s pod-id

3) migrate the image to the desired deployment node

4) create a container & start the instance `docker run imageName`

It's Important to make sure that each IP address and ports which are passed as a parameter to the docker file are accessible to the running docker container

4) open your browser and check the following URL http://<host>:<port>/config/actuator/health

If every thing is fine you will see a json respose which summarizes app's services status
{"status":"UP","components":{"db":{"status":"UP","details":{"database":"PostgreSQL","validationQuery":"isValid()"}},"discoveryComposite":{"description":"Discovery Client not initialized","status":"UNKNOWN","components":{"discoveryClient":{"description":"Discovery Client not initialized","status":"UNKNOWN"}}},"diskSpace":{"status":"UP","details":{"total":502468108288,"

free":167600648192,"threshold":10485760,"exists":true}},"ping":{"status":"UP"},"redis":{"status":"UP","components":{"publisherRedisConnectionFactory":{"status":"UP","details":{"version":"5.0.7"}},"redisConnectionFactory":{"status":"UP","details":{"version":"5.0.7"}}}},"refreshScope":{"status":"UP"}}}