Action Manager , Fault Tolerance and Bootstrap

Group 1, Team 1

May 8, 2021

Souptik Mondal, Shubhankar Saha, Ramanjaneyulu Payala

Mentor and Professor : Prof. Ramesh Loganathan

TA: Pratik Tiwari, Shubham Agarwal, Jay Krishna

Contents

1 Introduction

One of the most important feature of any platform is monitoring and fault tolerance of that platform. These two are very related concepts. In order to add fault tolerance feature, all the components should be monitored continuously. Our team also implemented the platform initialization feature and scaling concept. When any platform boots up, each of the components should be initialized properly and in a specified order, so that all the dependencies are fulfilled. So, it is also a very important feature of our platform.

2 Action Manager

The Action Manager performs all the necessary actions as requested by the host machines , after they had completed the execution of a service. More specifically it performs 3 different types of tasks :

- It communicates with the sensor manager for controlling specific controller/sensor.
- It communicates with the user/client who requests the service.
- It notifies via SMS ,email to a list of users as mentioned in the input json file.

2.1 Technology used:

- The action manager will be a standalone module . It will run inside a docker container.
- It accepts the request/command as a Json.
- It notifies via SMS ,email to a list of users as mentioned in the input json file , inside the action manager module.
- For other communication it will use Kafka.

2.2 Interaction with other modules :

- The action manager will receive information from the host machines. The received information will contain the following details
  - User details like email id, mobile number
  - Commands for the Sensor Manager.

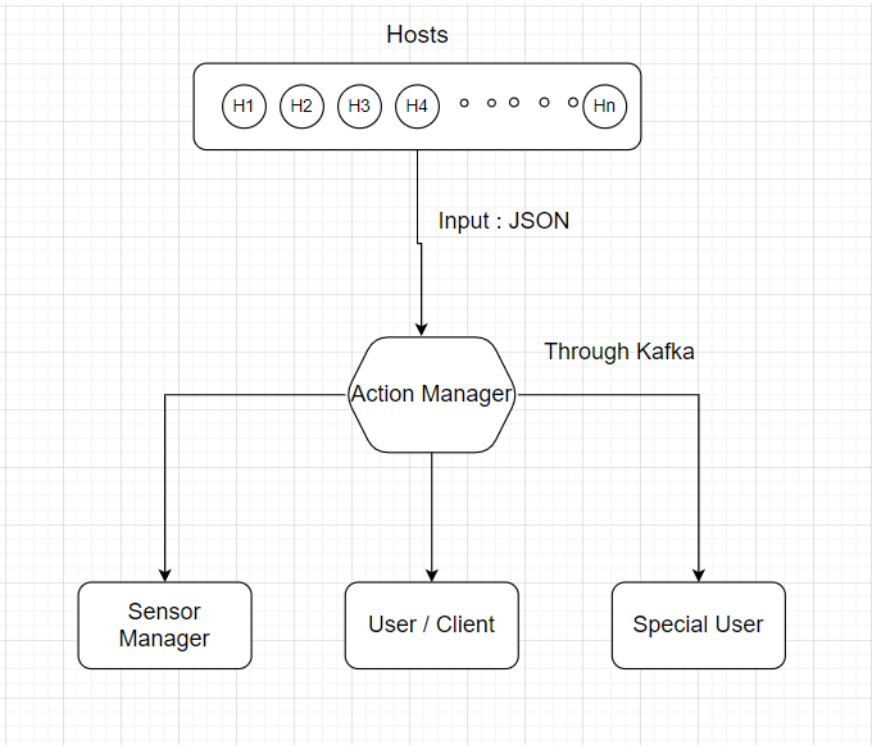

Figure 1: Action Manager

- The action manager will communicate with the sensor manager
  - It will receive instructions from the host machine using kafka_topic named "action_manager". Then it will parse all the information and differentiate what to do next i.e the instruction is for users or sensor manager.
  - It will communicate with the Sensor Manager via kafka_topic named "sensor_manager" and send the commands to the sensor manager. The Sensor Manager will communicate with Sensor Controllers to execute those commands.
- There is a heartbeat function in the Action manager. It will send status to the Monitoring module via a kafka topic named "HeartBeat".

3 Monitoring

This module will continuously monitor all the critical components so that the platform can detect if there is any failure of service or node.

- It is implemented by using heartbeat signal method. In this method we are continuously checking all the components by sending heartbeat signal and analysing the responses from those components.
- If there is any type of unusual behavior it will notify the fault tolerance module.

3.1 Technology used:

- The Monitoring Module will be a standalone module. It will run inside a docker container.
- For communication it will use Kafka.

3.2 Interaction with other modules :

- In monitoring, implemented heartbeat kind of mechanism which keep track of all micro services/modules mainly critical components.
- Basically this module will communicate with each critical modules to get the status of that module for example Dead or Alive.
- If any module is down it will notify the fault tolerance module using API call.

4 Bootstrap and Fault Tolerance

The bootstrap module is responsible for initializing all the modules in the platform. The fault tolerance module will make sure that if any module or service is down, it will restart it again.

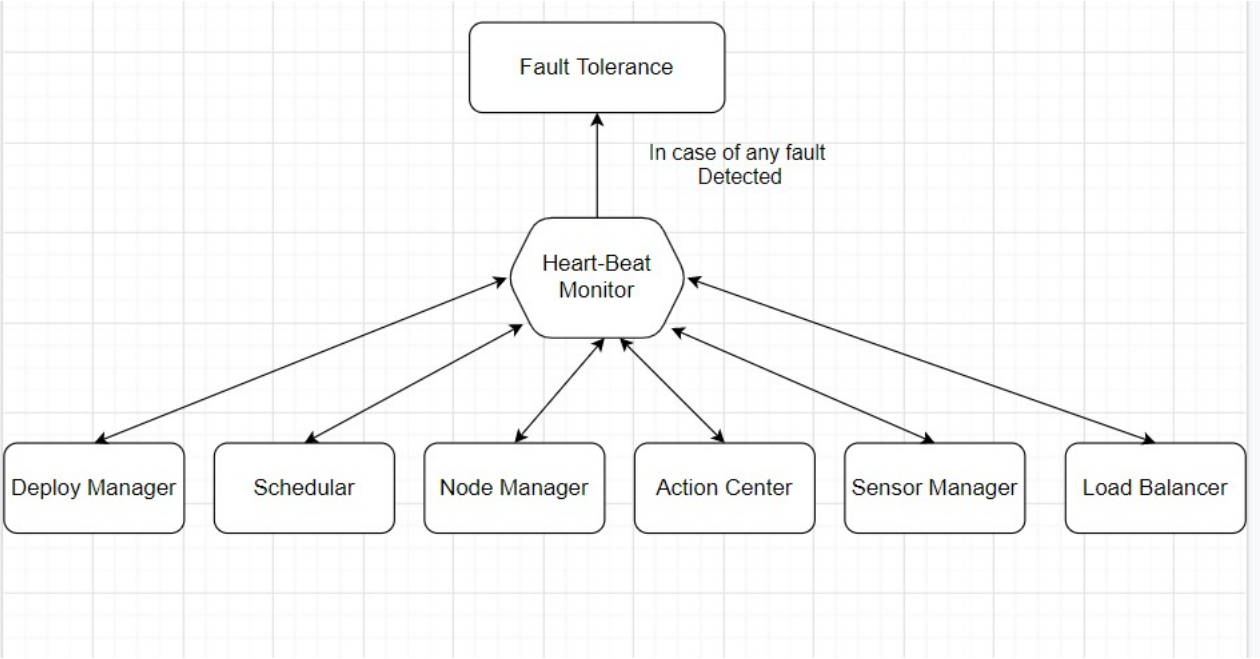

Figure 2: Monitoring

- This module clears any previous log that are stored in database.
- It creates a new docker network.
- The bootstrap follows a sequence to initialize the modules. The sequence is like :
  - communication module
  - app_ui
  - sensor_manager
  - app_repo
  - service_lcm
  - scheduler
  - load_balancer
  - worker_nodes

  - deployer
  - action_manager
  - monitoring
- If there is any type of unusual behavior the fault tolerance module will take the necessary action. The fault tolerance module will restart the module which is down, by the help of load balancer module.

4.1 Technology used:

- The Bootstrap and Fault Tolerance Module will be a standalone module. It will run outside a docker container.
- For communication it will use Kafka and Flask.

4.2 Interaction with other modules :

- The fault tolerance module will communicate with the monitoring module and the load balancer module.

5 Scaling

The scaling will add new worker node with the help of load balancer module if required.

- If the load of the running worker nodes are high then the scaling will add new worker node to reduce the load by the help of the load balancer module.
- The new requests will be run on the new worker node.