IoT for Improving 1st Responders' Situational Awareness and Safety Network Application

SUMMARY

This Network Application supports the implementation of a situational awareness platform in support of Central Command Centre (CCC) operations, so as to obtain full awareness from the field during a disaster response. It utilises the Mobitrust Platform, which allows CCC to monitor PPDR agents in the field equipped with Body-Kit Devices, by automatically collecting, retrieving and monitoring data from different types of sources: agent bio-sensors (e.g., Electrocardiography [ECG], Oxygen Saturation [SpO2], respiration rate); geographical/indoor positioning; internal communication systems; vehicles; devices (e.g., drones); shared services (e.g., private websites or shared folders); and real-time text, audio and video transmissions. The UC4 vertical application is fully compatible with Kubernetes-based deployments. Following the paradigm of cloud-native solutions, all the internal components exist as CNFs.

DEPENDENCIES

The UC4 vertical application is fully compatible with Kubernetes-based deployments. Following the paradigm of cloud-native solutions, all the internal components exist as CNFs. This contributes to the few hardware dependencies that the lightweight version of this vertical application presents:

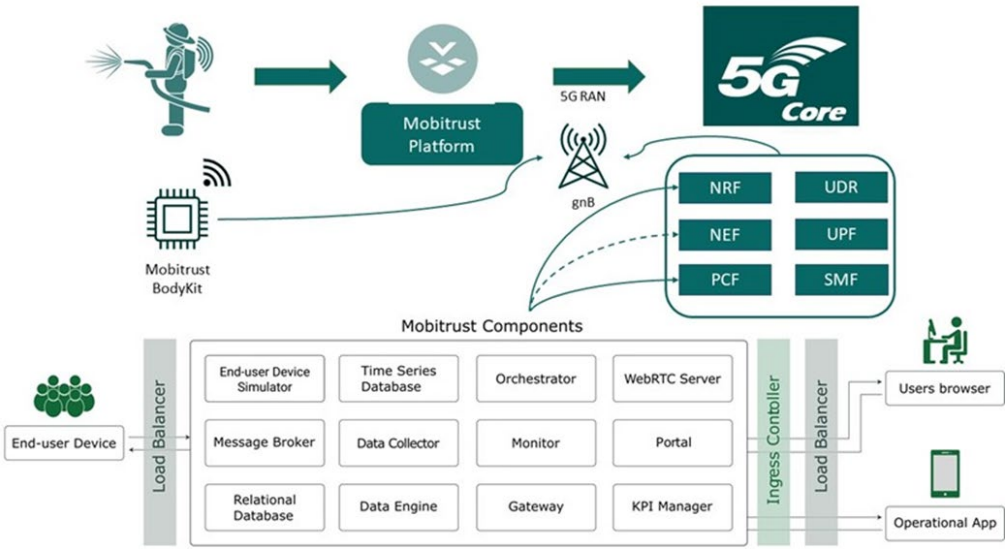
- 8GB RAM.
- 16GB storage.
- i5 1.80GHz processor.

ARCHITECTURE & DEPLOYMENT

In terms of deployment, the internal components of the application may be deployed using a script that in-vokes sequentially a specific set of YAML files (one per component). This process uses readiness probes, to make sure that the dependencies among the different components are fulfilled, and the application is successfully deployed. In the future, the application will also be deployable through Helm package manager. In terms of cluster constraints, there are a few that need to be met:

- Kubernetes cluster has a Load Balancer installed, as well as a Nginx-controller.
- Storage is available through Persistent Volume Claims (PVCs).
- A DNS core change needs to be conducted by the cluster admin in order to redirect traffic aiming to reach the application for the correct deployment.

It is also advised to create a dedicated namespace for the application, aiming to prevent any co-existence problems (arising from the convenience of different applications in the same namespace). Internet access must be granted, mainly to deal with the loading of maps for geo-location.



5G-EPICENTRE's UC4 vertical system under test - specific architecture

This project has received funding from the European Union's Horizon 2020 Innovation Action programme under Grant Agreement No 101016521.

CONTACT

For more information, do not hesitate to visit the website https://www.5gepicentre.eu/ and/or contact the 5G-EPICENTRE team.

Contact the 5G-EPICENTRE team by filling in the **form** provided. Apply **here**!

Follow Us on our social media for more NetApps trends:

MICROSERVICES OF MCX APPLICATION

The following list describes the main components of the Network Application (which are all CNFs):

- **End-user Device Simulator:** This component is used for integration tests and debug purposes. It pretends to simulate the data streams usually established between a real end-user device and the Mobitrust several components.
- **InfluxDB:** This represents the DB used to store the information collected from the multiple sensors present in the Mobitrust BodyKits (BK).
- **Orchestrator:** The orchestrator is responsible for the management of the control data. It deals with the authentication and authorisation of users. Moreover, it is responsible for the setup of the end-user device components, including its drivers, and the establishment of the data channels for sensors and communication devices (cameras and microphones).
- **Web Real-Time Communication (WebRTC) Server:** The WebRTC Server is the component that deals with audio and video transmission in real time from the field to the CCC.
- **Message Broker:** This component represents the communication backhaul of the system. It follows a publish/subscribe model. The Message Broker is responsible for all the communication among components.
- **Telegraf:** A plugin-driven server agent for collecting and reporting metrics. Through connecting to the Message Broker, it collects data from the system, mainly sensor data from the end-user devices.
- **Monitor:** This micro-service is responsible for watching and reporting on the state of the end-user devices.
- **PostgreSQL:** The relational DB stores the information regarding users, end-user devices, WebRTC mount points, and their associations, as well as the access control policies.
- **Kapacitor:** This is a native data processing engine. It can process both stream and batch data from InfluxDB. With Kapacitor, it is possible to plug in custom logic, or user-defined functions, to process alerts with dynamic thresholds and perform specific actions based on these alerts.
- **Gateway:** The operational controller is responsible for the services provided by the CCC. It has all the backend operations that enable the visualisation of the data collected by the platform, as well as the processing of requests of the human operators.

The vertical application used to demonstrate the Network Application composed of the above services is a Portal which is the frontend of the platform developed by ONE. It corresponds to the actual CCC application to be used by human operators and it provides a way to obtain situational awareness by visualising all the data collected by the platform.

This project has received funding from the European Union's Horizon 2020 Innovation Action programme under Grant Agreement No 101016521.