MONASH University

Algorand

deSense

Monash University Blockchain Hackathon 2021

Description:

deSense solution is a proof of concept for using Blockchain for sensor networks data management & provenance plus operations security & control by smart contracts logic. deSense is written in Go, TEAL4 and Shell Scipt.
deSense , stands for decentralized remote sensing through power , security, provenance, performance and reliability of Algorand blockchain technologies. deSense uses all layer 1 features of Algorand in advanced and dynamic contexts of sensor networks and remote sensing.
deSense can be considered a fully fleged prototype, proof of concept and also used for educational purposes in these domains (themes in terms of Monash Hackathon):

- Energy : Energy related sensor networks are every where in energy cycle from production to consumption therefore the main focus of this participation (dSense).
- Building: In information era and dawn of AI no building process is complete without IOT, from supply chain of building material and technologies to daily usage by people.
- E-Health: Who does not care about patience vitals and medical records being secured, impossible to tamper with and guaranteed provenance.

Links:

deSense on GitHub

deSense YouTube Video

deSense PDF document

Author

Mohammad Ghiasi
Technology Architect
mailto: emg110@gmail.com
linkedin.com/in/emg110
github.com/emg110
@emg110

Databases
BigData
DataScience
Apps & Dapps
Service & API
Analytical processing

JS / ES
R
SCALA
Go
DSLs

Co-Author

Shaghayegh Gharehgozli
Frontend Expert
mailto: shm135810@gmail.com
github.com/shegbzo

Node.js
React
Redux
Redux-thunk
Databases
Web components
Web APIs

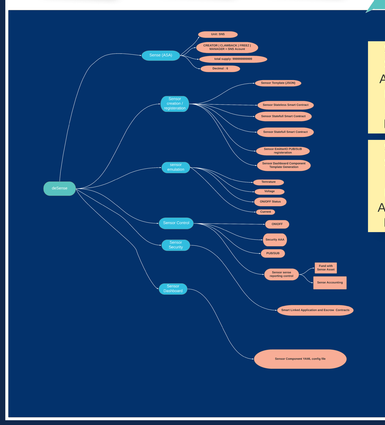
JS/ES
HTML5
CSS3
SCSS
JSS

Proof of Concept

Full fleged prototype

Block chain for SNS and decentralized sensing

The Concept of deSense



Scenario1: Sensor---> Algorand---> Broker ---> MQTT ---> Dashboard

Scenario2: Sensor---> MQTT---> Broker ---> Algorand ---> Dashboard

How it works:

deSense simulates a sensor network and it's sensors , which can be created via deSense CLI. Every sensor is in fact a sensor emulator which creates dummy but in-style and schema based sensing data on given time basis from configuration.
Sensing Data can be transfered to blockchain at any level. Immidiately after generation by emulator, via emitterio SNS broker after publication by emulator to network via emitterio client (MQTT on WebSockets), or even by another emitterio clinet just to receive a copy of remote sensing to keep records on blockchain (provenance only mode).
An Algorand standard Asset is created on sandnet (solution is dev only therefore developed on sandbox but can transfer to mainnet or testnet in minutes). The created ASA's name is SENSE and the unit is SNS. Every generated sensor has three components:
1- A sensor schema for emulator to use
2- An Escrow account stateless contract using sensor verification info.
3- An Statefull Application to govern the secirity and authenticity of sensed data and sending sensor info.
The two contracts are linked and sensor's escrow account is funded with Algos (only to cover fees and min balance) and SENSE units equal to total number of observations expected but in microSense(1000 observations === 1000 microSense === 0.001 SENSE) .
During operation sensor is turned on/off by operator and when turned on starts to send sensing data (observations) to either blockchain or emitterio MQTT relay or another emitterio MQTT client , given the chosen configuration.