Carbon Labelling - Blockchain based product carbon footprint system.

Jack Pellen-Pickersgill, Raja Naeem Akram, Konstantinos Markantonakis

Information Security Group, Smart Card and IoT Security Center
Royal Holloway, University of London

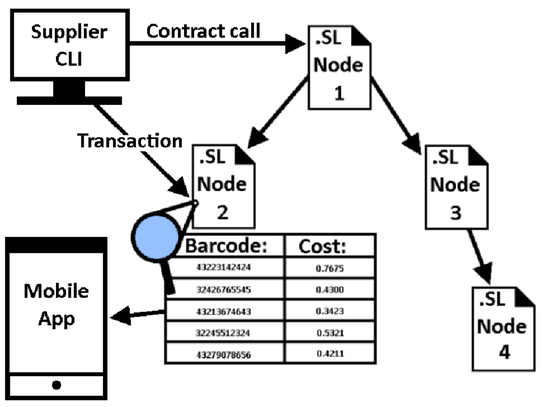
The Smart Card and Internet of Things Security Centre

Objectives

- Create a blockchain based system for the monitoring and storage of carbon data across a supply chain.;
- Create an application for users to scan product barcodes and retrieve carbon footprint data.

Introduction

- Carbon usage is an increasingly important issue in todays world, and as a result ethical consumerism is becoming more and more important in helping to reduce our carbon impact on an individual level.
- Companies wishing to track all of their carbon data may be reluctant to adopt traditional centralized systems, as a result of the security and data integrity issues they create.
- The goal of this project was to create a system which would allow companies and their suppliers to log carbon expenditure of individual products on an Ethereum blockchain. A seperate system would then retrieve data from the blockchain for use by a mobile app, which would allow users to scan product barcodes and see their carbon data..



System layout diagram

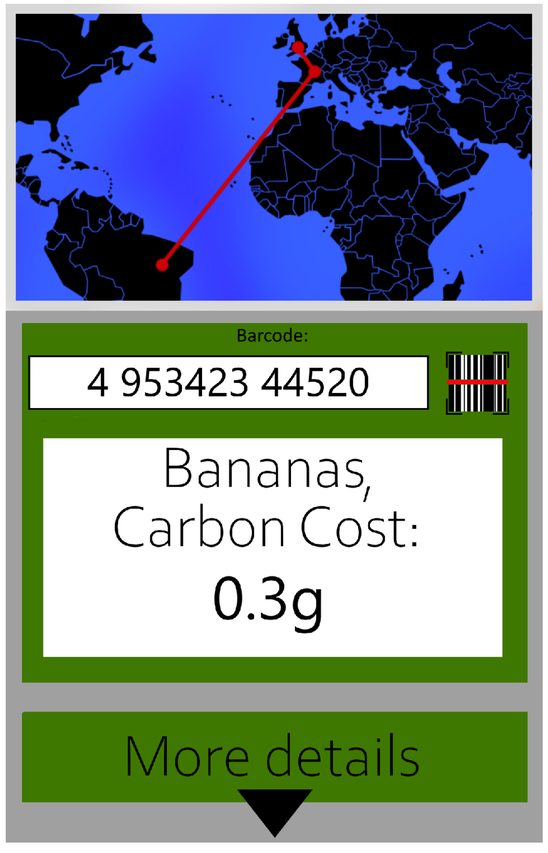
Supply Chain Representation

Modern supply chains consist of many suppliers. A company may source products from hundreds of suppliers, who in turn contract or buy from their own downstream suppliers. This continues all the way down to level of farms, mines and other sources of raw materials. A supply chain of this nation is essentially a tree, with each node a supplier, and each edge an exchange. The approach taken to storing and representing the supply chain was to represent each node, or supplier, as a contract, and the carbon costs for a product at each supplier as transactions. Each nodes contract uses the same source code, and therefore the same ABI. The contract stores: the name of the supplier; the suppliers children (their downstream suppliers) and a list of allowed senders. These senders are chosen when the contract is initialized, and can be added by any current allowed sender. This allows a supplier to create new contracts as they introduce new nodes to their supply chain. Transactions sent to the contracts will store information on how much carbon is used, and for what product it is used on, at the contracts real world location.

System Usage

The system has two main components, the program which suppliers use to update their carbon usage, and the mobile application used by customers to see a products carbon data. Since this system will need to know the barcodes of all products in a supermarket throughout their lifespan, the responsibility of supplying this data falls on the supermarket wishing to use the system. The suppliers will use a CLI tool to perform any actions needed on the blockchain, this includes: login carbon costs of products; adding new suppliers; removing suppliers and adding or removing allowed senders. Upon creating a new contract, one allowed sender can be given as an argument, this would normally be the address of the new supplier.

The mobile application will not communicate directly with the blockchain, since this would be too slow. Instead a database, which the app communicates with, will retrieve the new carbon data every 24 hours, therefore the retrieval of data is not a very time sensitive operation. Upon scanning a product, the user will see it's carbon cost, and a general map of where it has come from.



Mock up UI for the mobile app.

Benefits

- The system is secure and helps alleviate security pressure from the providing company
- The system could be integrated into a modern trade environment alongside 4IE technologies such as NFC.

Development and Evaluation

- The system was tested using ganache, since you can create blocks on the fly rather than waiting for one to be mined. The RPC is a good representation of the real Ethereum mainnet, so a system demonstrated to work on it, would also work on a live network.
- I used Python for the CLI, working with the Web3Py library. All contracts are written in Solidity. Using Ethereum allows the use of a turing complete language for contracts.

Conclusion

The system does solve some of the data integrity and security issues which would be faced by a normal centralised system, and does help demonstrate the possibility of using blockchain as a component of this kind of system. However the speed of operations on a live blockchain network may make some aspects of the system slower than ideal. This type of system could be adopted in a 4IR (4th Industrial Revolution) environment, and does help users to be more carbon conscious in their purchasing.

Contact Information

- Web: https://scc.rhul.ac.uk/
- Email: jack.pellen-pickersgill.2019@live.rhul.ac.uk