Article

FoodSQRBlock: Digitizing food production & supply chain with blockchain & QR code in the cloud

Somdip Dey [1,2,†], Suman Saha [2], Amit Kumar Singh [1,‡] and Klaus McDonald-Maier [1,*]

1 CSEE, University of Essex, UK; {†somdip.dey, ‡a.k.singh, *kdm} @essex.ac.uk
2 Nosh Technologies; suman.saha@nosh.tech
* Correspondence: somdip.dey@essex.ac.uk; dey@nosh.tech
† Current address: University of Essex, Wivenhoe Park, Colchester. UK. CO4 3SQ.

Version February 3, 2021 submitted to Sustainability

Abstract: Food safety is an important issue in today's world. Traditional agri-food production system doesn't offer easy traceability of the produce at any point of the supply chain, and hence, during a food-borne outbreak, it is very difficult to sift through food production data to track produce and origin of the outbreak. In recent years, blockchain based food production system has resolved this challenge, however, none of the proposed methodologies makes the food production data easily accessible, traceable and verifiable by consumers or producers using mobile/edge devices. In this paper, we propose FoodSQRBlock (**F**ood **S**afety **Q**uick **R**esponse **Block**), a blockchain technology based framework, which digitizes the food production information, and makes it easily accessible, traceable and verifiable by the consumers and producers by using QR codes. We also propose a large scale integration of FoodSQRBlock in the cloud to show the feasibility and scalability of the framework, and experimental evaluation to prove that.

Keywords: food production, supply chain, blockchain, qr code, cloud computing, food safety, barcode, traceability system, agri-food, agriculture

1. Introduction

Over the decades the food production system - the way to get food from farm to the table in the household - has evolved to a complex network. Today's food production system provides the consumers more variety, convenient, economical and healthier source of food, however, such a system comes with its own challenges, such as the case where ingredients produced by one producer could end up in thousands of other products distributed in many different shops [1,2]. This challenge is an issue of food safety, which is potentially detrimental to consumers' health and seriously damage the consumer's trust on the food market. For example, some immoral food producer could use trench oil to produce cooking oil, which is then distributed to thousands of shops, which is retrospectively bought and consumed by the consumer, making them sick in the process. Several cases of such accidents or food safety scandals such as "horsemeat scandal" "Sudan red", "clenbuterol", "Sanlu toxic milk powder" and "trench oil" [3] have happened all over the world. These scandals not just harm the economy of the food market, but at the same time threaten the safety and stability of the society as well. Although there are standards available such as General Food Law in EU [4], Food Safety Modernization Act (FSMA) in US [5], which try to standardize traceability of digital information of food production in some of the stages of food supply chain, these standards are regional and currently there is no holistic standardization of tracking and recording data for food traceability purposes in all stages of food supply chain across the globe. Henceforth, in order to deal with such a challenge related to food safety, Blockchain Technology (BT) [6,7] may play a vital role in the traceability of food ingredients in recent times such that the consumers can trace the source of the food ingredients that they are buying/consuming.

Traditionally many producers still record data of their production on papers, whereas, some producers digitize the production data, which doesn't enable interaction with other parties in the food system. Moreover, traditional food production systems are centralized in nature and could result in the trust problem, such as fraud, corruption, tampering and falsifying information. During a food-borne outbreak, sifting through thousands of documents (digital or paper) to trace food ingredients could be slow and complicated. In recent times, several methodologies [2,3,8,9] based on BT have been proposed to solve the challenge of food traceability for food safety purposes. The key strengths of utilizing BT is its decentralized, distributed and trusted nature, which could be advantageously used for food traceability and transparency for consumers at any point of the food production system. However, all the proposed BT frameworks [2,3,8,9] only deal with effective traceability of food supply chains, but not with technical solutions to make the food traceability more accessible to consumers such that they can verify and track their bought food items, may be with an easy-to-access device such as a mobile phone.

7 04660 78400 2

(a) UPC Barcode

UPC-A:	704660784002
EAN-13:	0704660784002
Amazon ASIN:	B016CQWOAE
Category:	Grocery & Gourmet Food » Grocery
Brand:	Tropicana
Model:	
Last Scanned:	Jun 12, 2019 06:00

(b) Details corresponding UPC

Figure 1. UPC barcode for Tropicana Trop50 Blackberry Cherry juice as fetched from BarcodeSpider.com [10]

On the other hand, a popular way to store food data digitally is by using 1D barcodes such as Universal Product Code barcode [11,12]. The Universal Product Code (UPC) barcode consists of 12 numeric digits that are uniquely assigned to each trade/food item. Every region or country maintains a database which holds the record of these trade/food items along with the UPC unique code, which are capable of storing the following data: the type of product, size, manufacturer and the country of origin of the food item. Therefore, if a consumer wants to know more about the bought food item, they have to use a barcode reader (using mobile phone application), which will fetch the unique UPC from the barcode and then fetch the information from an online database using the UPC. Although the information is fetched from an online database, the amount of information available on the item is limited to the type of product, size, manufacturer and the country of origin based on the type of the database. Therefore, no accurate traceability of the item throughout the supply chain is available for the consumer to verify. For example, Fig. 1.(a) represents a typical 1D barcode, which accompanies the Tropicana juice's product label [10], and only reflects the following information (see Fig. 1.(b)) about the product: UPC number, European Article Number (EAN), Amazon Standard Identification Number (ASIN) product category, brand, model and last scan date-time, as fetched from BarcodeSpider.com. It should be kept in mind that the 1D barcode in Fig. 1.(a) only allows to store 12 digit UPC and no other information, therefore, if the correct database is not used to fetch the information on the product using UPC then the information might not get retrieved at all. Moreover, if the consumer is not connected to the Internet, s/he might not even retrieve any information based on the UPC by scanning traditional 1D barcode since a lookup on the online database using UPC is necessary.

Another issue is that many food products have shorter shelf life such as fresh vegetables and fruits should be consumed within few weeks of being produced and the expiry/best before date is printed on the label of the food item during the packaging and hence, is not available in the UPC barcode information. Many food management applications [13,14] are now being offered to remind the consumer of the expiry/best before dates of the food products to reduce food waste in the household. In these applications (apps), a consumer can record the bought food items along with their expiry/best before dates, to create a reminder to consume the items before they expire. These applications offer barcode scanning to automatically enter the bought food items in to the apps, however, given the lack of information stored in the barcode, the consumer still has to enter the expiry/best before date of each items individually. Therefore, this calls for a technology which enables the consumer not just to be able to verify the source of the bought food items throughout the supply chain for food traceability purposes, but also automatically fetch the respective expiry/best before date.

To resolve these challenges we propose *FoodSQRBlock* (**F**ood **S**afety **Q**uick **R**esponse **Block**), a BT based framework, which digitizes the food production information such that the consumer and producers can trace the food produce at any point of the food production system, and make the information easily accessible using Quick Response code (QR code) such that the information can be retrieved and verified easily by the consumers and producers. In this paper, we also provide a proposal for a large scale integration of FoodSQRBlock in the cloud such that the framework could be adopted easily given the improvement and accessibility of cloud technology. To this extent the main contributions of this paper are as follows:

1. Propose FoodSQRBlock, a BT and QR code based framework to digitize food production information and retrieval.
2. Large scale integration of FoodSQRBlock in the cloud and related experimental evaluational in Google Cloud Platform.
3. Analysis and limitations of large scale integration of FoodSQRBlock in the cloud.

2. Preliminaries

2.1. Blockchain Technology

When Satoshi Nakamoto [7] released the technology named Bitcoin, he revolutionised the industry not because he had invented a new currency system, which does not require intervention of institutional mediator while transferring money from one entity to another, but because he had gifted one of the most disruptive technologies, which has come to life in decades. With the introduction of Bitcoin, Blockchain got introduced to the world, which is a digital ledger in which all transactions are recorded chronologically and publicly. But the application of blockchain is not just limited to crypto-currencies [15,16] such as Bitcoin and have proved to be useful in tracking ownership, provenance of documents, digital assets, physical assets, voting rights, etc. Blockchain network is traditionally of three types as follows:

1. **Public**: In this network, everyone can check and verify the transaction made. The network is also open to anyone who wants to participate in the consensus process.
2. **Private**: In this type of network, strict restrictions are applied on data access and the nodes (user/entity) have restricted access to specific block chains, which are monitored by a governing body.
3. **Consortium**: Nodes in this type of network can form partnership with businesses or other authorities. This type of network may be public or private and hence, this could be seen as a hybrid approach as partly decentralized.

Blockchain Technology is popular because of its design features, which are composed of six key elements as follows:

1. **Decentralized**: Blockchain data could be recorded, stored, updated and distributed without depending on a central authority or node.
2. **Transparency**: Data recorded and stored are transparent and are visible. Therefore, leveraging trust among its users.
3. **Open Source**: The source code as well as the most of the blockchain dependent systems are open to view, free to use and provide the ease of extension for other applications.
4. **Autonomous**: Blockchain updates are consensus based and thus data could be updated securely from a single user to the whole system. This feature provides autonomy to the system to update data securely.
5. **Immutability**: All data in the blockchain are reserved forever.
6. **Anonymity**: Blockchain also provides anonymity to its users and make the system more trust worthy by only using the users' blockchain addresses instead of their personal information.

2.2. QR Code

QR code [17–19] is an effective information transmission medium, which is widely used in product traceability, advertising, mobile payment, passport verification and other fields. QR code is defined into 40 symbol versions (to carry various data payloads) and 4 user-selectable error correction level (ECL): L, M, Q and H, which can correct up to 7%, 15%, 25% and 30% error codewords respectively when attacked by defacement. The larger QR version can offer higher data payload where the QR code can hold a maximum capacity of 2,956 bytes for a version 40 code. The error correction capability of QR code is one of the key features of this type of barcode introduced by QR code standard and allows the barcode reader to retrieve the data correctly if portions of the barcode are damaged. QR code utilizes Reed-Solomon error correction algorithm to realize this fault tolerance, where the error correction codewords would be generated by Reed-Solomon algorithm and added in the tail of QR code data codewords [20–23]. Usually two error correction codewords could be used to correct codeword data error. Obviously, the larger the QR code version and the error correction level, it can offer higher data payload and reliability. Fig. 2 represents a QR code in which a simple, "Hello, World!" message is embedded.

Figure 2. Representational QR code with "Hello, World!" message embedding

2.3. Cloud Computing

Cloud computing (cloud) [24,25] is a model to enable ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources such as networks, servers, storage, applications, and services, which can be rapidly provisioned and released with minimal management effort or service provider interaction [26]. The essential characteristics of Cloud computing are summarized as follows:

1. **On-demand self-service**: For cloud computing, capabilities can be provided automatically when needed, without requiring any human interaction between the consumer and the service provider.

2. **Broad network access**: In this type of service, computing capabilities are available over the network and accessible through several mechanisms disposable for a wide range of consumer platforms such as workstations, laptops, and smartphones.
3. **Resource pooling**: In cloud, computing resources are pooled to accommodate multiple consumers, and hence, dynamically allocating and deallocating them according to consumer's demand. Moreover, the provider resources are location independent, i.e. the consumer does not have any knowledge or control of their exact location.
4. **Rapid elasticity**: In cloud, computing capabilities can be provided flexibly and released to scale in and out according to the consumer's demand. Therefore, the consumer has the perception of unlimited, and always adequate, computing capabilities.
5. **Measured service**: In cloud, resource usage can be monitored and reported according to the type of service being offered. This is particularly relevant in pay-per-use, or pay-per-user services because it grants a great transparency between the provider and the consumer of such services.

Cloud services can be provided to consumers in a variety of ways, and one such service is Software as a service (SaaS) [27], where the software and its related data are centrally hosted in the cloud computing environment such that the software could be provided to numerous consumers.

3. Related work

Agri-food production and supply chains [2,3,8,9] have been studied extensively. Li et al. [28] developed a dynamic planning method for agri-food supply chain. This methodology minimizes the losses of agri-food products while simultaneously maximizing the profits for agri-food supply chain members. In 2015, Foroglou et al. [29] produced a study on applications of blockchain, which included electronic cash system. In the same study, Foroglou et al. also showed applications of blockchain in different domains in the future such as contracts, voting, intellectual property rights, smart property and finance. In 2016, Tian [3] proposed an agri-food blockchain framework using RFID technology to implement data acquisition, circulation and sharing in production, processing, warehousing, distribution and sales links of agri-food supply chain. In [2], Zhao et al. foresaw the storage capacity of BT being one of the most pressing challenges for applying BT in agri-food supply chain. Another challenge on implementing BT pointed out in [2] was the demanding computational power of utilizing such framework. In [8], Astill et al. mentioned the use of Internet-of-Things (IoT) (by using sensors embedded in different stages of food production) for data acquisition of food production and supply chain, and utilizing BT to create transparency of such data.

In [30], Bogner et al. proposed a decentralized application to share everyday objects/devices based on a smart contract using Ethereum blockchain and QR code. In this application [30], smart contract hosted on the blockchain, the local Ethereum client, and a web app is used to identify each object or device based on QR code, which is unique to each device. Although this approach is utilizing QR code and BT together to share devices in a decentralized manner, the logistics of food supply chain is much more complex and hierarchical, which requires a more holistic approach specific to such agri-industry. In [31], Kumar et al. proposed an appraoch to stop counterfeit of products in medicine supply chain using BT and encrypted QR code such that the consumer can trace each active ingredients in the medicine. In this approach each ingredient is traced using BT and then embedded as encrypted data in the QR code for further verification by the consumer. The approach in this method [31] is complex in terms of implementation in the agri-industry, especially given the distributed nature of the food supply-chain. Moreover, tracking each ingredient in the product also means additional storage memory usage to trace the ingredients. On the other hand, in [32] Baralla et al. proposed an approach to trace European food supply chain by using BT and QR code, however, the approach utilizes Hyperledger Sawtooth, which is suitable for enterprises while discouraging individual and small farmers with small production capacity to utilize such technology. Additionally, this approach does not encourage recovery and verification of data such as expiry date of the food product during the shopping phase of the food supply chain. The multi-level system architecture proposed in [32] also generates the QR code at every step of the supply chain, which also makes the QR codes more redundant and unnecessary.

From the published aforementioned studies two important challenges arise from using BT in food production system:

1. Improving the storage capacity of blockchain data while not increasing the computational power requirement.
2. Making BT based food production framework more accessible to consumers such that the food items could be verified easily at any point in the supply chain.

In this paper, we propose, FoodSQRBlock, BT based framework to make food production data more traceable and then make it available/accessible to the consumers using QR code. Since, the whole framework is implemented on a large scale in the cloud, which offers flexible scalability as per consumer's demand, the challenge faced due to storage capacity is not an issue.

4. Proposed Framework: FoodSQRBlock

In order to design BT based framework to make food production data more traceable and accessible, first, we have to analyze the different phases and activities present within a generic food supply chain. In our proposed framework, we focus our research on Farm-to-Fork supply chain, which has five main phases [32] characterizing a generic food supply chain, as follows.

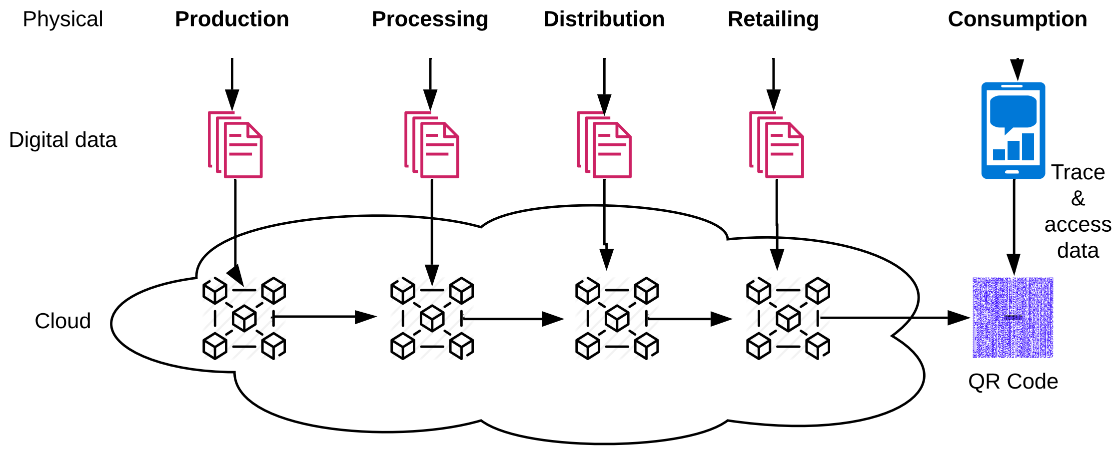


Figure 3. Overview of the System Architecture of FoodSQRBlock based on Farm-to-Fork supply chain

- **Production**: This is the primary production phase which represents all the activities related to agriculture within the farm.
- **Processing**: In this phase, harvesting the produce into products is performed. Preparation and packaging of the produce is also performed in this phase, where each package is uniquely identified trough a production batch code.
- **Distribution**: In this phase, once the product is packaged and labeled, it is released for distribution to different warehouses and other distribution centers for product storage.
- **Retailing**: In this phase, the products are delivered to the retailers, who sell the products to the consumers, from the distribution centers.
- **Consumption**: The consumer is the end user of the food supply chain, where s/he buys the product and requires the quality standards verifying traceability and access other relevant data about the product such as the expiry date.

In our proposed framework, FoodSQRBlock, we focus on digitizing data from the first four phases (production, processing, distribution and retailing) and then use these data in the BT for traceability and accessibility by the consumer in the consumption phase. All digital data regarding the first four phases will be recorded and maintained in the blockchain in the cloud.

System Architecture: To design our BT framework we propose a multilevel system, whose architecture is represented in Fig. 3. Our system has three layers as follows.

- **Physical layer**: This layers consists of different food products from different farmers and producers within the supply chain.
- **Digital data layer**: This layer includes every single digital data associated with the produce belonging to the physical layer, which will be used for traceability and accessibility. One example of data about the produce could be the expiry date of the food product.
- **Cloud layer**: In this layer, the digital data is processed in the Cloud using BT, which is used for traceability and accessibility.

Now, we introduce our FoodSQRBlock framework with an example to digitize the food production data at the four different phases (production, processing, distribution & retailing) of the supply chain and make the data available and accessible to consumers (consumption) for verification purposes using QR code. Fig. 4 shows the exemplar conceptual FoodSQRBlock framework of an agri-food supply chain traceability system. For this exemplar supply chain we have considered the production and processing phases to happen in the farm, whereas, the warehouse represents the distribution phase and the shops represent the retailing phase. In Fig. 4, the food item is produced and processed in a farm, where the relevant information are digitized and stored in block (genesis block/block 0) and then the item is transported to a warehouse/distribution center, from where the item is finally transported to a shop for consumers to buy. In each step/phase of this supply chain, a new block is created which stores the hash of the previous block such that at any point the item (produce) could be tracked and traced. We utilize SHA256 algorithm [33], which is very popular in BT nowadays, for the hash function to generate the hash of the previous block. In our FoodSQRBlock, we utilize SHA256 for the hash function since it provides the required security for the associated computational cost on the cloud. If we utilize a different hashing algorithm such as SHA512, it is computationally more expensive and takes longer to compute on the cloud, which ultimately increases the computational cost, especially if thousands of digital data of the produce are processed on the cloud every day.

Within the FoodSQRBlock we have two modules: *Encoding* & *Decoding*. The Encoding module digitizes the produce information, generates the blocks and also generates the QR code holding the information. The Decode module, which is an open source software (algorithm), enables the consumers to fetch and verify the information about the produce. The program source code for Encoding and Decoding modules are provided in Sec. 7.

Encoding Module: If we consider f_i as the i_{th} instance of the food item being produced at a farm or manufacturing plant, then the following information ($info(f_i)$) about the produce: produce name (p_i), type (t_i), farm/manufacturing plant id ($farm_i$), size of produce (s_i), production date ($pdate_i$), expiry/best before date ($edate_i$), could be digitized such that these information could be passed along with the block for traceability and verification purposes by the consumer. Therefore, the digitized information could be represented as follows:

$$info(f_i) = \{p_i, t_i, farm_i, s_i, pdate_i, edate_i\} \tag{1}$$

In $info(f_i)$, the unique farm id ($farm_i$) is stored, which correlates with the farm data (farm name, Geo-location of the farm), stored in a database maintaining records of all the farms/manufacturing plants. Here, the unique farm id is also generated using the hash function on the stored details of the farm/manufacturing plant, and hence, ensuring that the farm id is unique for each farm. In the genesis block (block 0), $info(f_i)$ is stored. Whenever the produce is transported or processed by an entity in the supply chain a new block is generated, which holds the original $info(f_i)$ as well as the hash of the previous block. QR code could be generated at any point in the supply chain and it holds the information passed in the block ($info(f_i)$ and hash of previous block). Fig. 5.(a) shows

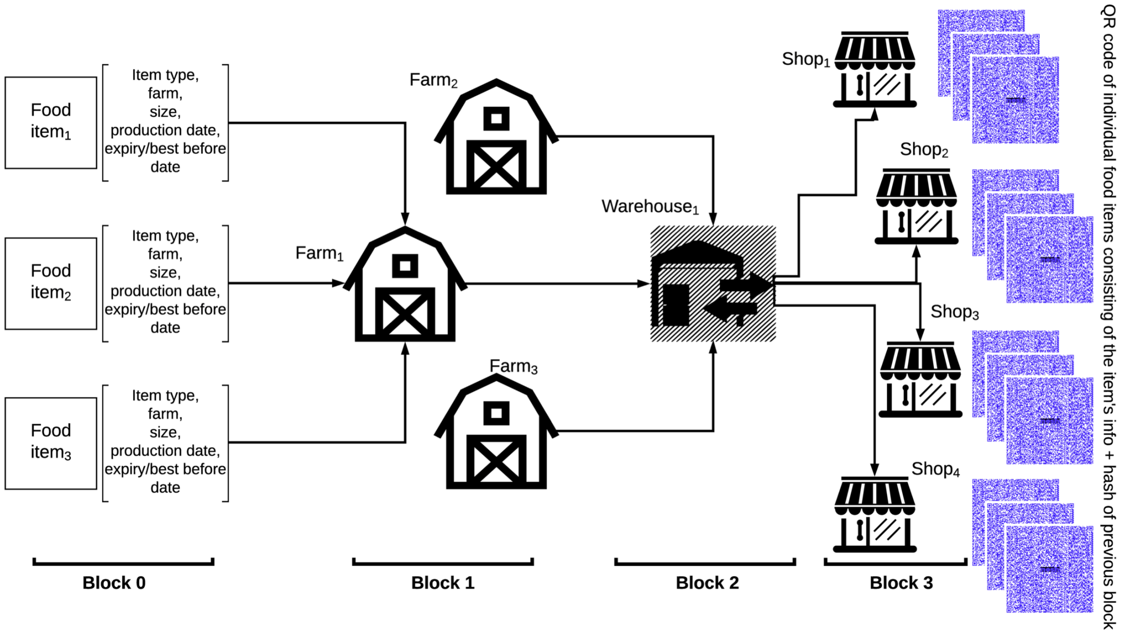


Figure 4. Conceptual FoodSQRBlock framework of an agri-food supply chain traceability system based on QR code & blockchain technology

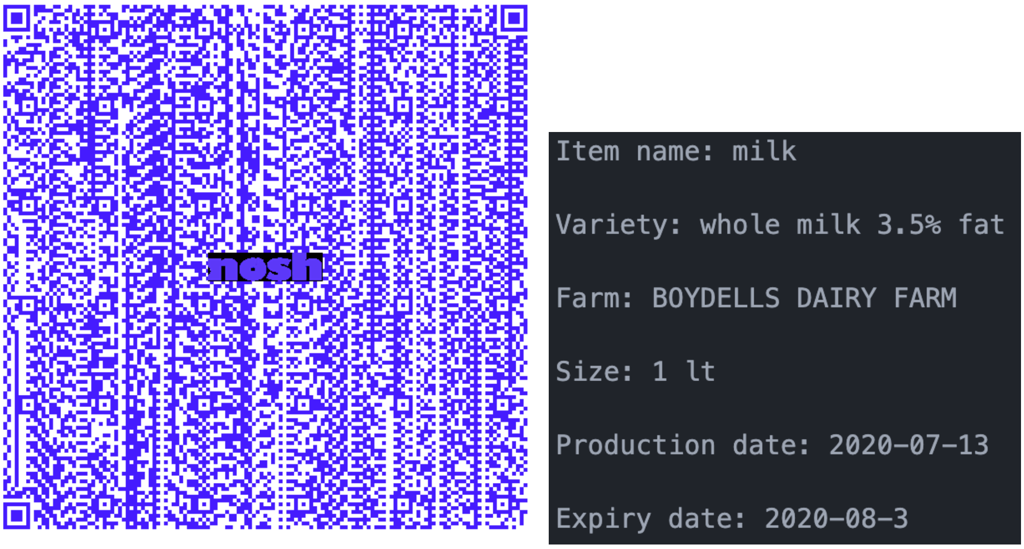
the QR code generated at the shop when the produced milk is transported to the shop (from farm to warehouse/distribution center to shop).

Decoding Module: In this module, the information ($info(f_i)$) and the hash of the previous block are fetched using a QR code scanner (smartphone) from the QR code. Since, the information of the previous block is made available online for the consumers to verify, the Decode module performs a hash function on the information from the previous block and compares the value with the hash value fetched from the QR code. If the hash values match then it means the information about the produce is authentic (leveraging immutability characteristic of BT). A reverse search on the unique farm id fetched from the QR code is performed to fetch the details of the farm/manufacturing plant where the produce originated and the information is displayed to the consumer along with other information about the produce ($p_i, t_i, s_i, pdate_i, edate_i$). Since, the details of the farm/manufacturing plant is stored in a database, the reverse search on the database could be performed using the unique farm id. Fig. 5.(b) shows the decoded information about the dairy product, fetched from the QR code in Fig. 5.(a).

Note: Given the storage capacity of QR codes, more information could be stored/embedded in the QR code in the Encoding module. As a proof of concept and to test run FoodSQRBlock framework, we found $info(f_i)$ embedding $p_i, t_i, farm_i, s_i, pdate_i, edate_i$ are most suitable for the purpose.

5. Experimental evaluation: case study & large scale integration of FoodSQRBlock

In order to implement and evaluate the FoodSQRBlock in the cloud, we chose Google Cloud Platform (GCP)'s [34] Compute Engine service, which is a virtual machine consisting of 8 vCPUs, 16 GB RAM memory and 500 GB disk memory space, to setup our cloud server. The Compute Engine was running on Debian GNU OS (Linux) version 10 (code name "buster"). We chose the following two food items to test run our implementation: *milk* & *pumpkin*. Milk is sourced from a local farm (Boydells Dairy Farm in Essex, UK) and pumpkin is also sourced locally (Foxes Farm Produce in Essex, UK). The milk is sourced from the farm and then sold directly in the farmer's market, whereas, the pumpkins are moved to a warehouse distributing facility after being sourced from the farm and then were sold in the farmer's market. We implemented the FoodSQRBlock as a webservice (SaaS) on the GCP Compute Engine server to process the blocks and generate QR code, containing all the relevant information, for individual produce. We simulated the production in the aforementioned farms (Boydells Dairy Farm



(a) QR code (b) Output of decode module

Figure 5. QR code holding information of a dairy product, which is produced at Boydells Dairy Farm in the UK, generated by FoodSQRBlock

& Foxes Farm Produce) in the cloud platform to replicate the scenario of a real world food production system. For a batch of milk being sourced around the same time, are all accumulated into the same block and blockchain consisting of two blocks (genesis block and hash of genesis block) are generated for each individual batch. Whereas, for pumpkins, every one hundred pumpkins are accumulated into the same block and since, each batch of pumpkin made it to farmer's market (shop) via distribution center, each batch was processed to consist of 3 blocks in the chain (genesis block, hash of genesis block when it went to distribution center and hash of the block from distribution center after making it to the farmer's market).

5.1. Experimental Evaluation

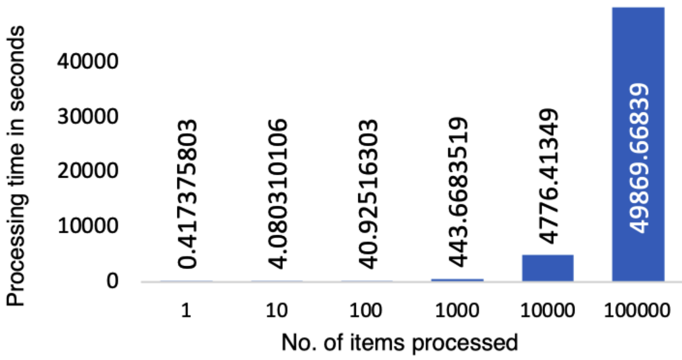


Figure 6. Time taken to process different number of items (using FoodSQRBlock) in Google Cloud Platform's Compute Engine

In the GCP's Compute Engine, the webservice of FoodSQRBlock processed (generate hash of block along with stored information as mentioned in the Encoding module in Sec. 4) different numbers of produce to see the time taken for processing such information. Fig. 6 shows the processing time in seconds for 1, 10, 100, 1000, 10000, 100000 number of produces on the GCP's Compute Engine.

5.2. Analysis & Discussion

From Sec. 5.1 it is evident that given the resources of GCP's Compute Engine, 10,000 items could be processed easily under 80 minutes. However, if a farm or manufacturing plant produces more than 10,000 items in a day then a more computationally powerful cloud server is required to be able to process the items within a reasonable time. Because GCP allows flexible ad-hoc computation resources for its Compute Engine, it is easy for us to scale up or down based on the numbers of items to be processed, therefore, solving the storage capacity issue of BT as mentioned in Sec. 3.

It should also be kept in mind that currently there is no standards or regulations on standardize traceability of digital information of food production across the globe and henceforth, implementation of the proposed FoodSQRBlock would require collaboration across regions.

6. Conclusion

In this paper, we proposed FoodSQRBlock, a blockchain technology based framework, which digitizes the food production information, and makes it easily accessible, traceable and verifiable by the consumers and producers by using QR codes to embed the information. We also implemented FoodSQRBlock in Google Cloud Platform to replicate a real life food production scenario using milk & pumpkins as produce examples from real farms in the UK. Experimental evaluation proves the feasibility and scalability of FoodSQRBlock implementation in the cloud.

7. Code availability

The program source code to implement the FoodSQRBlock (Encoding & Decoding modules) could be accessed from ##Code will be provided upon acceptance##.

Author Contributions: Conceptualization, Somdip Dey; methodology, Somdip Dey; software, Somdip Dey; validation, Somdip Dey and Suman Saha.; formal analysis, Somdip Dey; investigation, Somdip Dey; resources, Somdip Dey and Suman Saha; data curation, Somdip Dey; writing–original draft preparation, Somdip Dey; writing–review and editing, Somdip Dey and Suman Saha and Amit Singh and Klaus McDonald-Maier; visualization, Somdip Dey; supervision, Somdip Dey and Amit Singh; project administration, Somdip Dey; funding acquisition, Somdip Dey. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by Nosh Technologies under Grant nosh/agri-tech-000002 and by the U.K. Engineering and Physical Sciences Research Council (EPSRC) under Grant EP/R02572X/1 and Grant EP/P017487/1.

Conflicts of Interest: This research was pursued such that part of the proposed methodology could be implemented as a feature in the commercial mobile application named nosh -Food Stock Management https://nosh.tech

References

1. Yiannas, F. A new era of food transparency powered by blockchain. *Innovations: Technology, Governance, Globalization* **2018**, *12*, 46–56.
2. Zhao, G.; Liu, S.; Lopez, C.; Lu, H.; Elgueta, S.; Chen, H.; Boshkoska, B.M. Blockchain technology in agri-food value chain management: A synthesis of applications, challenges and future research directions. *Computers in Industry* **2019**, *109*, 83–99.
3. Tian, F. An agri-food supply chain traceability system for China based on RFID & blockchain technology. 2016 13th international conference on service systems and service management (ICSSSM). IEEE, 2016, pp. 1–6.
4. General Food Law. https://ec.europa.eu/food/safety/general_food_law_en. Accessed: 2020-06-26.
5. Food Safety Modernization Act (FSMA). https://www.fda.gov/food/guidance-regulation-food-and-dietary-supplements/food-safety-modernization-act-fsma. Accessed: 2020-06-26.
6. Dey, S. A Proof of Work: Securing Majority-Attack in Blockchain Using Machine Learning and Algorithmic Game Theory. *International Journal of Wireless and Microwave Technologies* **2018**.
7. Dey, S. Securing majority-attack in blockchain using machine learning and algorithmic game theory: A proof of work. 2018 10th computer science and electronic engineering (CEEC). IEEE, 2018, pp. 7–10.

8. Astill, J.; Dara, R.A.; Campbell, M.; Farber, J.M.; Fraser, E.D.; Sharif, S.; Yada, R.Y. Transparency in food supply chains: A review of enabling technology solutions. *Trends in Food Science & Technology* **2019**, *91*, 240–247.
9. Kamble, S.S.; Gunasekaran, A.; Sharma, R. Modeling the blockchain enabled traceability in agriculture supply chain. *International Journal of Information Management* **2020**, *52*, 101967.
10. UPC 704660784002 Lookup. https://www.barcodespider.com/704660784002. Accessed: 2020-07-07.
11. Teach ICT - Barcodes. https://biturl.top/QJzuYz. Accessed: 2020-07-07.
12. Drobnik, O. *Barcodes with iOS: Bringing together the digital and physical worlds*; Manning, 2015.
13. nosh - Food Stock Management. https://nosh.tech. Accessed: 2020-07-07.
14. NoWaste - Food Inventory List. https://www.nowasteapp.com. Accessed: 2020-07-07.
15. Blockchain is useful for a lot more than Bitcoin. http://theconversation.com/blockchain-is-useful-fora-lot-more-than-just-bitcoin-58921. Accessed: 2020-07-07.
16. How could blockchain be used in the enterprise. https://www.computerworlduk.com/galleries/security/how-could-blockchain-be-used-the-enterprise3628558/. Accessed: 2020-07-07.
17. Dey, S.; Mondal, K.; Nath, J.; Nath, A. Advanced Steganography Algorithm Using Randomized Intermediate QR Host Embedded With Any Encrypted Secret Message: ASA_QR Algorithm. *International Journal of Modern Education & Computer Science* **2012**, *4*.
18. Huang, P.C.; Chang, C.C.; Li, Y.H.; Liu, Y. Efficient QR code secret embedding mechanism based on Hamming code. *IEEE Access* **2020**, *8*, 86706–86714.
19. Dey, S. Sd-eqr: A new technique to use qr codestm in cryptography. *arXiv preprint arXiv:1205.4829* **2012**.
20. Dey, S.; Agarwal, S.; Nath, A. Confidential encrypted data hiding and retrieval using qr authentication system. 2013 International Conference on Communication Systems and Network Technologies. IEEE, 2013, pp. 512–517.
21. Dey, A.S.; Nath, B.J.; Nath, C.A. A New Technique to Hide Encrypted Data in QR Codes (TM). Proceedings on the International Conference on Internet Computing (ICOMP). The Steering Committee of The World Congress in Computer Science, Computer …, 2012, p. 1.
22. Lin, P.Y. Distributed secret sharing approach with cheater prevention based on QR code. *IEEE Transactions on Industrial Informatics* **2016**, *12*, 384–392.
23. Dey, S. New generation of digital academic-transcripts using encrypted QR code$^{\mathrm{TM}}$: Use of encrypted QR code$^{\mathrm{TM}}$ in mark-sheets (academic transcripts). 2013 International Mutli-Conference on Automation, Computing, Communication, Control and Compressed Sensing (iMac4s). IEEE, 2013, pp. 313–317.
24. De Donno, M.; Tange, K.; Dragoni, N. Foundations and evolution of modern computing paradigms: Cloud, iot, edge, and fog. *Ieee Access* **2019**, *7*, 150936–150948.
25. Qi, Q.; Tao, F. A smart manufacturing service system based on edge computing, fog computing, and cloud computing. *IEEE Access* **2019**, *7*, 86769–86777.
26. Bohn, R.B.; Messina, J.; Liu, F.; Tong, J.; Mao, J. NIST cloud computing reference architecture. 2011 IEEE World Congress on Services. IEEE, 2011, pp. 594–596.
27. Ali, A.Q.; Sultan, A.B.M.; Abd Ghani, A.A.; Zulzalil, H. A Systematic Mapping Study on the Customization Solutions of Software as a Service Applications. *IEEE Access* **2019**, *7*, 88196–88217.
28. Li, D.; Kehoe, D.; Drake, P. Dynamic planning with a wireless product identification technology in food supply chains. *The International Journal of Advanced Manufacturing Technology* **2006**, *30*, 938–944.
29. Foroglou, G.; Tsilidou, A.L. Further applications of the blockchain. 12th student conference on managerial science and technology, 2015, pp. 1–8.
30. Bogner, A.; Chanson, M.; Meeuw, A. A decentralised sharing app running a smart contract on the ethereum blockchain. Proceedings of the 6th International Conference on the Internet of Things, 2016, pp. 177–178.
31. Kumar, R.; Tripathi, R. Traceability of counterfeit medicine supply chain through Blockchain. 2019 11th International Conference on Communication Systems & Networks (COMSNETS). IEEE, 2019, pp. 568–570.
32. Baralla, G.; others. Ensure traceability in European food supply chain by using a blockchain system. 2019 IEEE/ACM 2nd International Workshop on Emerging Trends in Software Engineering for Blockchain (WETSEB). IEEE, 2019.
33. Irving, G.; Holden, J. How blockchain-timestamped protocols could improve the trustworthiness of medical science. *F1000Research* **2016**, *5*.

34. Krishnan, S.; Gonzalez, J.L.U. *Building your next big thing with google cloud platform: A guide for developers and enterprise architects*; Springer, 2015.