A Product Authentication Scheme for Supply Chain system via Smart Contracts using Blockchain Technology and Facial Recognition

Srivatsa D¹, Nithin Aakash¹, Sahisnu S¹, Priyanka Kumar*¹,

Department of Computer Science and Engineering
¹,*¹ Amrita School of Engineering, Coimbatore,
Amrita Vishwa Vidyapeetham, India.
{cb.en.u4cse17459, cb.en.u4cse17443,cb.en.u4cse17450}@cb.students.amrita.edu
k_priyanka@cb.amrita.edu

Abstract. The supply chain management industry is struggling with inadequate resources for efficient authenticity verification. Blockchain technology and smart contracts can overcome such conventional limitations to authenticate products in an easy, economical and secure manner. Decentralized and immutable blockchain systems allow product tracking to its origin. In this paper we have proposed and implemented a system for product authentication using blockchain technology based on ethereum platform by making use of smart contracts. We have analysed the existing centralized system and the need to shift from the existing centralized system to a decentralized blockchain based ledger technology. The use of blockchain technology in supply chain technology has reduced the complexity in product authentication by making the entire history of the product available from its production stage till it reaches the customer.

1. Introduction

Blockchain technology is useful for projects that require real-time collaboration between mutually-suspicious contributors over the Internet, with the emergence of blockchain technology the method of sending money changes and gets more secured without any involvement from the third party. Blockchain has a common platform which helps us collaborate with other competitors without any fear, Applications built here can have any composition also user will get to have more flexibility. It maintains a ledger full of details of all the required information needed which is visible to everyone in that chain. As everyone in the network has a full copy of the running blockchain it is impossible to tamper the details. Also as programs and details reside in multiple locations, every detail is verified by multiple people.

![Figure 1. How a Block is added and validated in a Blockchain](image-url)
Currently blockchain Technology is mostly used in the field of Identity management, Supply chains, Internet of things and edge computing, Telemedicine and digitized records. We now face a major trust issue in the products we buy from any ecommerce website. We have these trust issues because there is lack of traceability and transparency in the Product, it is because product could be tampered at any stage of production by any entity. This paper revolves around the idea of providing traceability to user any by using Blockchain as a network using Ethereum platform. To resolve the problem we bring all production entities into our network, there are manufacturer, Assembler, distributor and finally user.

Every entity in the network is considered to be separate nodes, every action taken by each node is recorded and is made trace-able ,As the network is decentralized even the end user is able to find out the origin of the product , also one can track each process involved in making of the product

One of the important features of this blockchain network is when a block is added, even the owner doesn't have the permission to change it. Also this ledger doesn't have any special privileges for any user meaning anyone inside the network can upload the block, but it will validated and visible to all the others in the network.

![Figure 2. and Figure 3. Represents facial data points](image)

Each and every user who wishes to join our network will be given a username and password along with his face data stored in a database , so that users can have multiple ways to login and also to reduce the time it takes for them to login. Facial recognition is a way of recognizing a human face through technology. A facial recognition system uses biometrics to match facial features from an image . It compares the information with a database of known faces to find a match, it provides user a faster login option with a good security. Face recognition when used in multifactor authentication enhances the security and reliability of this system making it more trustworthy.
We have also given QR code for our product. It is a machine-scannable image that can instantly be read using a camera. Every QR code consists of a number of black squares and dots which represent certain pieces of information. When your Smartphone scans this code, it translates that information into something that can be easily understood by humans.

![QR Code Image]

**Figure 4.** QR code for products

Smart contracts help you exchange money, property, shares, or anything of value in a transparent, conflict-free way while avoiding the services of a middleman. They reduce complexity in transaction of products through automated verification and execution of the multiple business transactions involved. A decentralized, immutable record also ensures all the entities involved have equal access to information which helps us build trust. Its a code written and deployed in blockchain using ethereum platform, as smart contracts communicate with each other the sanity of product at each level is maintained.

![Smart Contract Image]

**Figure 5.** Smart Contract
Roadmap:
Initially this paper described our model working principle in the Architecture Section. Then the flow of our project is given in diagrammatic representation in the model diagram section, followed by the code we used in smart contracts is briefed in the implementation section. Results obtained so far are listed, finally the conclusion.

2. System Architecture
At present the existing way of storing data is centralised. Stakeholders in a supply chain rely on this centralised data to authenticate a product. This method is unable to efficiently protect against malicious activities. The International Chamber of Commerce has reported that unauthenticated products will drain 4.2 trillion USD by 2022. Blockchain technology is capable of providing an efficient solution for addressing the problem of counterfeit products in the supply chain.

Our proposed model is explained below. We use the smart contract ethereum platform and blockchain technology to realize the task of product authentication. Firstly we have four stages in this model namely manufactured, assembled, distributed and received. We create a file called ProductAuthentication with the following fields.

- product_id : It is a unique id associated with the product.
- product_stage : It identifies the stage of the product in the supply chain.

The following fields will be specific to manufacturers.

- prod_name : It holds the name of the product.
- prod_description: It holds the description about the product.
- owner: It denotes the owner of the product.
- manufacturer_name: It denotes the name of the manufacturer
- manufacturer_location: It holds the location of the manufacturer.
- manufacturer_time: It holds the manufacturing time of the product.

The following fields are specific to assemblers.

- assembler_name: It holds the name of the assembler.
- assembler_location :It holds the location of the assembler.
- assembled_time : It denotes the time when the product was assembled.

The following fields are specific to the distributors.

- distributor_name :It denotes the name of the distributor.
- distributor_location : It denotes the location of the distributors.
- distributed_time : It denotes the time at which the product was dispatched.
The manufacturers, assemblers, distributors will all have their own portals authenticated by face recognition and supported by password protection to view the details of the product details and also update details about the product while the customer only has privilege to view the details of the product. The manufacturers will only be able to add the products since he is the start of the supply chain. Followed by the manufacturer the assembler will be able to update the details regarding a specific product added by the manufacturer by making use of the product id. Following the assembler many distributors will be able to update the details of the product. The manufacturers, assemblers and the distributors will have to spend the ethers from their respective wallets for updating the product details and appending them to the blockchain, while none of the stakeholders will have to spend their ethers for viewing the product details.

In this system we achieved Facial Authentication by making use of the computationally less expensive and robust EigenFaces. It uses Eigenvalues and EigenVectors to transform dimensionality and project a training data on feature space. We made use of opencv4js node package for achieving facial authentication.

All the stakeholders involved in this system will have a secured two factor authentication portal for using the system. The stakeholders will have to first enter their username and password which is followed by a second factor face authentication. This multi-factor authentication will make the system more secure to use and trustworthy.
Blockchain technology is immutable as well as distributed technology where all the data is been replicated to all the participating clients in a blockchain. So if we want to tamper a record, we need to tamper data on each client to whom data has been distributed which is nearly impossible because of the availability of computing power. Hence the blockchain is seen as one of the most secure technology ever the world has witnessed.

3. **Our Contribution: Implementation on Ethereum platform**

This application is developed using smart contracts and run on Ethereum Blockchain using Ethereum Virtual Machine (EVM). Smart contracts contain the business logic of the application and are in charge of reading and writing data to the blockchain by executing the logic. The actual implementation of the smart contracts and all the core logic is done using a programming language called Solidity. The smart contracts preserve the state and whenever the state changes, it is written to the blockchain.

Writing transactions to the Ethereum blockchain requires a transaction fee called gas. To test the working of this application, ganache and truffle are used. Ganache is a local in-memory blockchain, simulates a private blockchain hosted on localhost (127.0.0.1) and port 8545(default) and provides a few accounts with fake ethers for transactions. Truffle framework enables us to deploy(migrate) smart contracts to the blockchain and provides a space to develop client-side applications. Interaction with smart contracts is done using truffle console(command), which provides a platform for calling functions and verifying the state inside the smart contracts.
Figure 8. Ganache interface - Ethereum accounts with fake ethers.

Figure 9. Ganache Interface - linked Smart Contracts.
Fig 10. Truffle Framework - Smart Contracts migration.

a) \textit{setter function}: The following methods are used to write information to the blockchain about the product.

\textit{manufacturerBlock}(): method used to add manufacturer and product details to the product_info for every new product created. It returns the product_id for every new product and it should be used later to refer to this product. This also stores the manufactured_time as the system's current time and sets the product_stage to MANUFACTURED in the product_info.

\textit{assemblerBlock}(): method used to add assembler specific details to the product_info which mapped to the product_id returned in manufacturerBlock() method. It first verifies that the product has completed it's manufacturing stage and those information are already added to the blockchain. This stores the product's assembled_time as system's current time and changes the product_stage to ASSEMBLED in the product_info.

\textit{distributorBlock}(): method takes the product_id, distributor_name and distributor_location as input and it first checks if the product_stage is ASSEMBLED and only then adds the details to the product_info. This stores the distributed_time as system's current time and changes the product_stage to DISTRIBUTED in the product_info.
b) getter function: The following are view methods which cannot change the state and which return the validated information for the product_id specified. It is designed such a way that consumers can check all the details for a product and the users in the intermediate stages of production can check the details of all previous stages for a product.

getManufacturerDetails(): method takes product_id as input and checks if it's valid and verifies the product stage, it then returns the product and manufacturer details for that product.

getAssemblerDetails(): method takes product_id as input and checks if it's valid and verifies the product stage, it then returns the product and assembler details for that product.

getDistributorDetails(): method takes product_id as input and checks if it's valid and verifies the product stage, it then returns the product and distributor details for that product.
truffle(development)> app.getManufacturerDetails(0,{from: accounts[2]})
Result {
  '0': 'OnePlus 8',
  '1': 'Mobile phone with 6 gb ram',
  '2': '0x6002769318b7d69b80618c39e9bce00c3257913e9190',
  '3': 'OnePlus',
  '4': 'South Korea',
  '5': '<BN: 5eedd377>',
  name: 'OnePlus 8',
description: 'Mobile phone with 6 gb ram',
owner: '0x6002769318b7d69b80618c39e9bce00c3257913e9190',
manufacturer_name: 'OnePlus',
manufacturer_location: 'South Korea',
manufactured_time: '<BN: 5eedd377>'}

Figure 13. Execution of getManufacturerDetails()
4. Results

We modelled the Product Authentication system by utilising Ethereum Platform. The platform integrates the business logic with Ethereum blockchain using smart contracts. Smart contracts include methods to add the product details at each stage of production and also check the validated details of any product included in the system. This is then migrated to the Ethereum network. The Ethereum Virtual Machine (EVM) executes the code for every function call. During execution every state is recorded permanently in the blockchain. All product related information is stored as state variables thus provides integrity as it cannot be manipulated or reversed and is not always true when considering conventional databases. Hence this is considered as the easy and practical way of product authenticity.

Figure 16. Deploying Production Authentication application on Ethereum.

5. Conclusion and Future Direction

We have developed a command line application for product authentication using technologies like blockchain and Smart Contracts and tested its integrity. We have shown the results on ethereum platform, when product details were added in stages. For future work, we would like to expand this to a client side application and concentrate more on providing confidentiality and availability. We will also work on linking facial recognition for authentication of users into the application.

References

[1]. Nakamoto, Satoshi. “Bitcoin: A peer-to-peer electronic cash system.” (2008)
[2]. Lo, S.K., Xu, X., Chiam, Y.K., Lu, Q.: Evaluating suitability of applying blockchain. In: International Conference on Engineering of Complex Computer Systems (2017)
[3]. Nomura Research Institute, Survey on Blockchain Technologies and Related Services (2016)
[4]. Thompson, S.: The preservation of digital signatures on the blockchain - Thompson - See Also, University British Columbia iSchool Student Journal, vol. 3, no. Spring (2017)

[5]. https://blockchain.ieee.org/images/files/pdf/ieee-future-directions-blockchain-white-paper.pdf

[6]. https://link.springer.com/chapter/10.1007/978-3-030-36987-3_23

[7]. Truffle Suite. https://www.trufflesuite.com

[8]. Truffle. https://www.trufflesuite.com/truffle

[9]. Ganache. https://www.trufflesuite.com/ganache

[10]. Tschorsch, F., Scheuermann, B.: Bitcoin and beyond: a technical survey on decentralized digital currencies. IEEE Commun. Surv. Tutorials 18(3), 2084–2123 (2016)

[11]. Mohanta, B.K., Panda, S.S., Jena, D.: An overview of smart contract and use cases in blockchain technology. In: 2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT) (2018)

[12]. Sato, Tatsuya, and Yosuke Himura. “Smart-Contract Based System Operations for Permissioned Blockchain.” New Technologies, Mobility and Security (NTMS), 20189th IFIP International Conference on. IEEE, 2018.

[13]. Tamper proof Birth certificate using Blockchain Technology, Maharshi Shah and Priyanka Kumar, International Journal of Recent Technology and Engineering (IJRTE), E-ISSN:2277-3878, Volume-7, Issue-5S3, February 2019.

[14]. Cred utilization in the product authentication process: a framework and research agenda for product counterfeit preventionZoltánLeventeFejes& Jeremy M. Wilson, Anti-Counterfeiting and Product Protection Program, School of Criminal Justice, Michigan State University, East Lansing, MI, USA Version of record first published: 19 Nov 2012

[15]. Face recognition for security efficiency in managing and monitoring visitors of an organization2014 International Symposium on Biometrics and Security Technologies (ISBAST)Published: 2014.

[16]. BrunFalaPogg+95.Brunelli, R., Falavigna, D., Poggio, T., and Stringa, L. (1995). Automatic Person Recognition by Using Acoustic and Geometric Features, Machine Vision and Applications, 8(5):317-325.

[17]. To build Scalable and Portable Blockchain Application using Docker, Maharshi Shah and Priyanka Kumar at SoCTA 2019, 27th Dec-29th Dec 2019, NIT Patna, International Journal of Computer Networks & Communications (IJCN)ISSN: 0975 - 2293.

[18]. An Efficient and Novel Buyer and Seller’s Distributed Ledger based protocol using smart contracts, Dhanush G A, Srivatsa D, Nithin Aakash, Sahishnu S and Priyanka Kumar, 9th-12th January 2020, ICDCIT 2019, LNCS, volume 11969, pp-349-363, DOI https://doi.org/10.1007/978-3-030-36987-3_23, Online ISBN 978-3-030-36987-3.

[19]. An Autonomous Food Wastage Control Warehouse: Distributed Ledger and Machine Learning based Approach, Nikish Kumar S V, Saiharsha B, Sanjay Tharagesh, Janavi and Priyanka Kumar, 9th-12th January 2020, ICCCNNT, 1-3 July 2019, IIT Kharagpur, India.