Asset Tracking System using Blockchain

Darsh Mehta
Department Of Computer Science
Sardar Patel Institute Of Technology
Mumbai, India
darsh.mehta@spit.ac.in

Jay Mehta
Department Of Computer Science
Sardar Patel Institute Of Technology
Mumbai, India
jay.mehta@spit.ac.in

Jainam Jain
Department Of Computer Science
Sardar Patel Institute Of Technology
Mumbai, India
jainam.jain@spit.ac.in

Surekha Dholay
Department Of Computer Science
Sardar Patel Institute Of Technology
Mumbai, India
surekha_dholay@spit.ac.in

***Abstract*—Asset tracking here refers to the process of tracking a product or a batch of products throughout the supply chain to make sure that the products that reach the customers are tamper-proof and original. Depending upon the type of products, every product has either some kind of packaging or a body of it's own which can be used for the purpose of tracking it throughout the supply chain. The ultimate goal of asset tracking system is to trace the route of the products in the supply chain, making sure it is not tampered and if it is, then point out exactly where the tampering took place in the supply chain.**

***Index Terms*—Blockchain, Keccak-256, Supply Chain, Truffle Framework, Ethereum, Smart Contracts, Decentralized System**

I. Introduction

The practice of tampering with the products has been on a constant rise since the last decade. Supposedly a product is being transferred from the one stage of supply chain say manufacturer to the next person in supply chain say distributor, someone sees this an opportunity and replaces the original product with a duplicate one in order to sell the original product someplace else for maximum profit. The statistics have shown that people using counterfeiting products have increased by 23% over the last three years. With customers not being satisfied with products they blame the company for it hence it damages the reputation of the company. Also, if products like medicines get tampered, it will result in a disaster.

Current system is not all transparent and definitely its not reliable. There are numerous ways in which the product can be tampered at any stage in supply chain. It becomes impossible to pinpoint the exact location and time at which product is being tampered with. The current system uses technologies like Bluetooth Low Energy Beacon(BLE) [1] which works fine only within a certain range, Geo Positioning Satellite(GPS) [2] tracker which ensures no security of the user and there are high chances of data theft, then there is Radio Frequency Identification(RFID) [3] which has its own disadvantages like its costly and it works only when it catches waves of specific frequency and if there are other waves of similar frequency it becomes impossible to predict the location of the asset. The biggest drawback of these systems is that it is completely centralized hence have a single point of failure and are subjected to hacking, privacy evasion and compromise. Thus there is a need for decentralized solution.

Blockchain is an immutable, tamper-proof, decentralized distributed ledger [4] [5] with ample security features [6] that make it possible to create an asset tracker to track our product throughout the supply chain. Blockchain uses ordered logs and events which are used to achieve traceability and auditability. In addition, using Ethereum makes blockchain programmable. Ethereum smart contracts empowered blockchain by allowing the execution of code. [7]

Our system will take care of all the general requirements: **Accountability,** which means everyone in the supply chain will be given their roles and responsibilities and will be held accountable for their actions. **Integrity,** where all transactions, logs and events are timestamped and tamper-proof. **Authentication and Authorization** which ensure that certain functions and operations can only be carried out by specific group of people.

II. Literature Survey

In this section, we review and summarize work related to asset tracking system. We also survey decentralized blockchain-based marketplaces.

The authors in [8] proposed a simple scheme based on Ethereum blockchain which involves transporting a product between two parties. The scheme depends on a single key that is given to the transporter by the seller [9]. The key is transported along with the item and is handed over to the receiver who is the buyer. The buyer then needs to enter the key for verification. The key hash would already be available in the smart contract which acts as an escrow and holds the buyer's Ether. The Ether would only be placed in the seller's account if the hash of the key entered by the buyer matches the existing hash in the smart contract. A successful verification leads to the transfer of the Ether to the seller [9]. This solution is easy to implement as it is simple and depends on only one key. The method however, depends on trusting the transporter completely that no manipulation of the key would take place before reaching the buyer. Although this approach fails with a malicious act from any participating entity especially the transporter.

In this paper [9] the authors have proposed a system to track the assets using a transmitter and a receiver, a transmitter attached to items to be tracked and a receiver carried by the user. The circuit has five modules: transmitter, receiver, coding and decoding, timer, and alarm.

In this paper [10] authors have suggested a cellular based asset tracking system as an reliable alternative for satellite based vehicle tracking system. Its two way data connectivity offers an advantage to simple GPS position reporting systems.

The authors of this paper [11] have proposed a solution that was designed using in-lining approach to deliver a middle-ware that acts as an insulator for hiding the internal workings of the asset tracking system prototype by providing homogeneous and abstract environment to the highest layers.he middle-ware supports an array of services such as: (1) extraction of locus data from windows location sensor; (2) laptop monitoring using RFID reader and passive tags; (3) bi-direction SMS communication; and (4) utilisation of database services to facilitate data management through SQL commands.

In this paper [12] an asset tracking system was developed that exploits both traditional asset tracking techniques and services offered by new Android devices. This paper discusses about a heterogeneous system consisting of Bluetooth Low Energy(BLE), RFID tags and smartphones for locating our asset. The two main functions of the proposed solution are: i) the Asset Proximity Locator (APL) discovers the BLE tags present in the area and it estimates the distance between the smartphone and the tags and ii) the Wandering Object Location Finder (WOLF) which is the user interface. The paper demonstrates that a Energy/Position Accuracy trade-off can be found to allow saving enough energy to ensure smartphones' battery lifetime equal to (or greater than) an entire working shift.

This paper [13] discusses a system of asset tracking system using ZigBee for a hospital. The authors outlined the tracking service model, suggested a highly cost effective method to track the assets in a hospital, analyzed the value of the application, and conducted the customer satisfaction survey to measure the results. The results show that Zigbee based low power asset tracking system is extremely effective in monitoring mobile assets.

The authors of this paper [14] have made a significant contribution to the research topic application of wireless communications in healthcare systems. They have deployed a real-time mobile asset tracking system in a local hospital and evaluated the accuracy of the system under a number of different Wi-Fi network configurations. The positioning scheme is based on statistical modeling of strength of signals received and provides accuracy of up to 1 meter on average. Although this system was used for tracking wheelchairs it can be extended to track any personal item like IV pumps, vital signs monitors, patients and medical staff that have 802.11- based tags.

This paper [15] published in 2005 stated that the asset tracking system proved to be a feasible technique to prevent future equipment loss as well as tracking employees in the hospital.

In this paper [16], an agri-food supply chain traceability system is established, based on RFlD and blockchain technology.This system covers the whole process of data gathering and information management of every links in agri-food supply chain, which realizes the monitoring, tracing and traceability management for the quality and safety of the agri-food ”from farm to fork”.This paper has highlighted the advantages and disadvantages of using RFID

In this article [17], the author conceptualizes the emerging BT and identify specific application areas of BT to supply chain operations from risk management/security perspectives. It is one of the first attempts to design an architecture of BT for revolutionizing supply chain practices and provide some rationales for its usefulness in enhancing supply chain resilience.

In this paper [18], the author proposed and discussed the adoption of blockchain technology in supply chain networks. The evolution of blockchain-based supply chain management is presented which enables the creation of shared, secure, decentralised ledgers, autonomous digital contracts (smart contracts), and trustworthy and secure networks. In addition, it supports transaction between partners (peer-to-peer) by reducing the role of middlemen/intermediaries in the network

In this paper [19], AgriBlockIoT enables the integration of IoT and Blockchain technologies, creating transparent, fault-tolerance, immutable and auditable records which can be used for an Agri-Food traceability system. Regarding the preliminary, very practical test: even if the Hyperledger Sawtooth-based implementation had better results in terms of measured metrics with respect to the Ethereum one, both implementations have different properties and capabilities that need to be considered before choosing one over the other.

In this paper [20] the author has proposed a method to track personal devices such as mobiles,jewellery etc using RFID. The items will have RFID tags and it will communicate with RFID readers which owners will carry. When of these items is out of range the owner will be alerted helping him to locate the lost object. The timestamp will also be recorded so that owner will know the exact time at which asset went missing.The authors have used Gumstix mini-computer and the WJM3000RFID to accommodate all of the needs for the project.

The authors of this paper [21] have proposed a new filter, Kalman-LULU filter, in signal processing stage, helps increase the accuracy of the system and reduce noise in fast pace. The proposed system is first introduced and equipped with enhanced data filtering and processing techniques to minimize the noise in Bluetooth signal for a more accurate distance estimation.The paper could provide useful insights in manufacturing industry to facilitate the application of cost-efficient Bluetooth Location-based indoor positioning system.

This paper [22] proposes a hybrid positioning system which is based on Wi-Fi fingerprinting and BLE trilateration is proposed for tracking assets. It can guide users who use mobile devices to find a particular asset. The Wi-Fi fingerprinting method can guide the user to move to the nearby area of the asset. Then, the asset can be positioned by the accurate positioning in a smaller area, which adopts BLE trilateration method.

III. Proposed Methodology

In this section, we describe our completely decentralised Ethereum blockchain solution based on the security features of blockchain technology. Our main goal is to make a system that tracks an asset throughout the supply chain and if there is any tampering or replacement, then find out exactly at which stage it took place. Our solution will work irrespective of the size and extent of the supply chain. Our solution will be extended to as many manufacturers or supply chain administrators as required and will work in an adequate, reliable and secure way similar to working with only one manufacturer or supply chain. The system will ensure that every entity in the supply chain is honest and there is no chance of tampered products reaching the customer. Also, unlike other solutions, there is no use of hardware parts or comparatively big devices which can be detached from the original product and attached to the duplicate ones.

*1) **System Overview**:* The proposed blockchain solution focuses on the tracking of assets between every two parties in a supply chain and validation of the assets at every stage. The main participating entities of the system, the manufacturer, supply chain entities, customer and arbitrator. Each of the entities has an Ethereum address and interacts with the smart contracts created throughout the process based on permissions. As the item gets handed over between two entities in a supply chain and a transaction takes place while verifying the asset. The roles of the participants can be summarized as follows:

- **Manufacturer:** The manufacturer is the creator or the first bearer of the product in the supply chain. He/She has the power to create or discard a product and can generate keys for products or product batches.
- **Supply chain entities:** The supply chain entities are various people involved in the supply chain of the product. These people can get the assets and verify it using the system and then can sell the genuine assets further.
- **Customer:** The customer is the end of the supply chain and doesn't own any contract or doesn't need to spend any ether. He/She consumes the product.
- **Arbitrator:** The arbitrator is the trusted entity by everyone else and has the right to view the logs of transactions. Also, in case of any dispute amongst other entities, they can approach the arbitrator in order to view the logs and resolve the dispute.

The main benefit of blockchain technology is transparency and the ability to log every event and transactions on a public ledger and review it everytime. The smart contracts used in the system utilize this property to create events and logs that help in tracking the item as it gets packaged, delivered and passed from one entity to another. Furthermore, a key hash is used to confirm the receiving of an item by the supply chain entities. Hence, the key is given with the asset to the next entity or finally to the customer and then its hash is created and compared to the hash already decentralised. This verification is done to ensure that the item that is received is the original one dispatched at the manufacturer's end and not an altered or fake one. In order to achieve the needed functionality with transparency and tracing the item as it moves through the chain of contracts, the smart contracts contain the following:

- **Methods:** Methods in a smart contract are used to create function calls. These functions execute and implement actions on certain parameters when called. These methods make repeating actions easier as we don't have to write the entire code again.
- **Modifiers:** Modifiers are used to make sure that required conditions are achieved before a function is executed. These play an important role in limiting access and strictly following predetermined rules. For instance, if the logs are to be viewed it is required that the address belongs to an arbitrator or the keys match to log a product received as genuine.
- **Events:** Events are the notifications that are logged into a public ledger so that they can be referred to in future in case it is needed. Any event logged creates a status update of all the parameters involved and logs them in the ledger.
- **Variables:** Variables are used for storage of data and recalling them after they have been declared to either update or refer the value. These variables keep on changing as transactions take place in a smart contract. The main variable used in a smart contract is to store Ethereum address of participating entities, their keys, the items they own and an IPFS hash.

*2) **System Design**:* Every entity in a supply chain will have it's own Ethereum address and a set of rules asserted to that address that they need to follow. Also the address of the entity decides what power or permissions the entity has. The Ethereum address will also be used to have ether that is spent when transactions take place. The amount of ether spent per transaction will be as minimal as possible in order to optimise the cost incurred to run the system. These transactions include creation of products, creation of keys for products and checking the product is genuine or not. Only the manufacturer has the rights to create a product or discard a product, rest all supply chain entities along with the manufacturer can create keys and check the genuinity of the product.

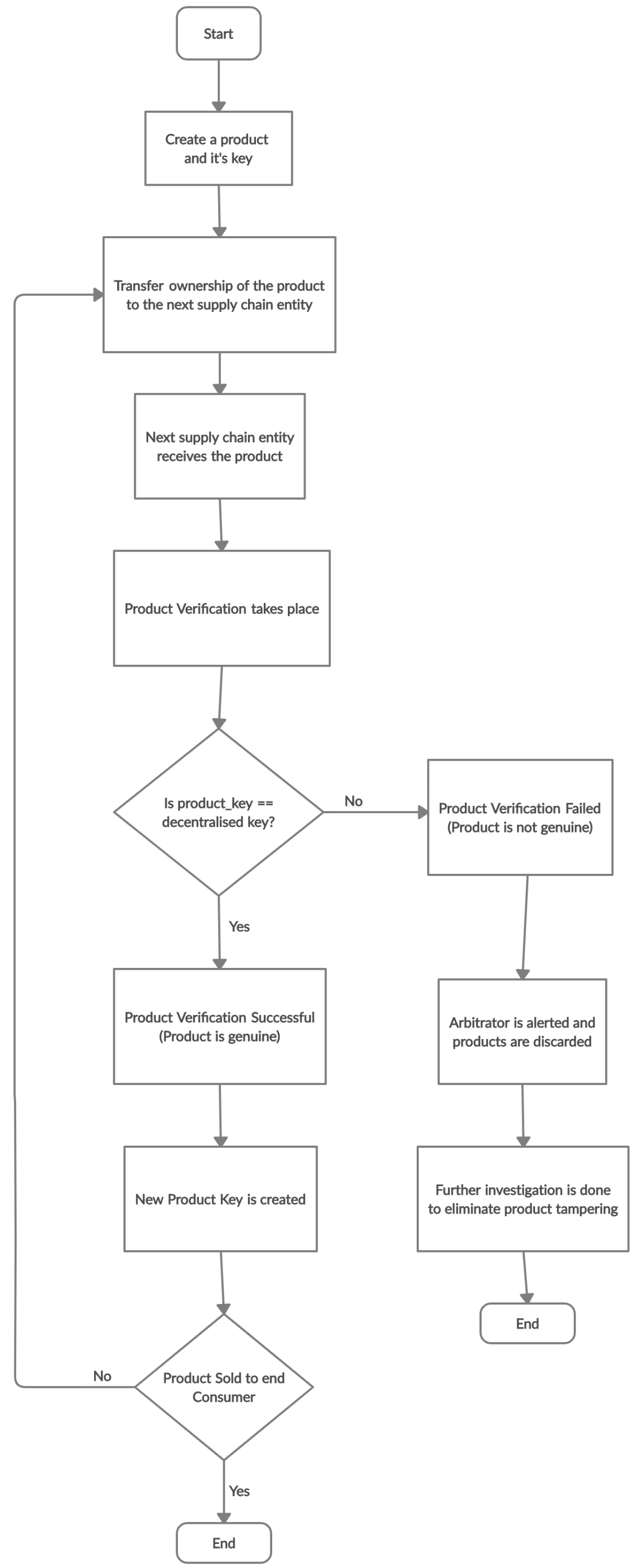
In order to assure that the product is genuine, at every stage in the supply chain the transfer of product ownership is governed by a contract. Now, every product or product batch has a decentralised key asserted to it and also has a QR code or NFC tag on it that has the key hashed using keccak-256 encryption which is in-built in Solidity. When the product reaches an entity he/she scans the product to obtain the hashed key. This key is then matched with the key decentralised by the original owner of the product. The event of key matching (or not matching) will be logged on a public ledger and can be brought up in future for reference. If the key matches, the product is genuine and the owner can make another key and sell the product or product batch ahead in the supply chain.

If this key doesn't match, then the product is not genuine and is tampered at that particular stage of supply chain. In this case the arbitrator is alerted and he/she can contact the manufacturer and the related entities. Then further investigation can be carried out to get to the exact cause of product tampering. Also these tampered products are discarded and hence don't reach the market. In this way it is guaranteed that the products that reach the consumers are the original ones.

Now, any attempt to brute force the key will be fruitless as the key on one side is decentralised using blockchain and smart contracts and is keccak-256 encrypted on the other side making it impossible to crack. Hence, this key is unavailable to anyone. Also any attempt to obtain the key from the system's side will be fruitless due to blockchain's secure decentralisation feature. If any of the supply chain entities help tampering the products then it'll be stopped at the later stage and hence stopped from reaching the market. The end consumer can also himself get the product verified by the retail store before buying.

*3) **Flow of the applications**:* Our system will follow a zero tolerance policy with unverified or tampered products. To obtain this we need to set a predefined actions for every entity regarding product verification and our system should follow a similar flow at every stage of the supply chain. In this section, we describe the flow of the system and the flow of actions of every entity involved in the system. In our system, the flow will consist of events and logs based on the actions of the entities and the result of product check.The flow goes as follows:

- **Creation of a product:** A product or a product batch is created by the manufacturer as only he/she has the power/permission to create a product or a product batch in the system. The creation of a product takes place via a method that first checks if the request has come from the Ethereum address that belongs to the manufacturer of the product. Along with the creation of the product a key is also



Flow chart illustrating the flow of the system.

created and decentralised, this key is also hashed using keccak-256 and is given to the manufacturer to put it on the product or batch. The method results in two events. One being the creation of the product and other the creation and decentralisation of the key. Both of these events are logged into the ledger which can only be seen by the arbitrator.
- **Transfer of physical ownership:** The product when sent to the next entity in the supply chain it's ownership is transferred to that entity using a method which results in another event of physical ownership transfer which is also logged. The method first checks that the request for ownership transfer is from the Ethereum address that is the owner of the product or product batch.
- **Verification of the product:** The product when received by a supply chain entity is verified by matching the keys. This matching is done via a method which first checks that the entity verifying is the owner of the product. This method checks if the keys match. If they match two events are emitted and logged, one being successful verification of the product and another one being creation of a decentralised key for further continuation of the product in the supply chain. If the keys don't match then one event is emitted and logged regarding tampered products received and the arbitrator is alerted. In this case, no key is produced hence the products are discarded for being sent further.

The last two flow components are repeated until the product reaches the end consumer. The end consumer may also get the product verified at the retailer's end before buying it.

4) Implementation Details and Algorithms: In this section, we describe the algorithms used in the smart contracts and the structural and functional aspects of the system in detail. The language used for writing the smart contracts is Solidity. All the functions, events, modifiers and variables are declared inside the smart contract. This section discusses the details of important algorithms and tools used.

- **Product Creation:** Creation of product is the first task to take place in a supply chain. It can only be done by a manufacturer. The algorithm will first check if the request is from the address of the manufacturer. If it is from manufacturer's address then a new product along with it's key will be created and two corresponding events will be emitted and logged. If it is not from the manufacturer's address then the access will be denied and no event will be emitted.

Algorithm 1: Creation of product

Input: e, E, batch_quantity, product_details.

E: Ethereum address that belongs to the manufacturer.

e: Ethereum address from which the request of creating new product has been made.

batch_quantity: The amount of products in the batch (can also be 1).

Product_details_valid: The function that returns if product details are valid or not.

Random_string: The function that returns a random string.

keccak265: The function returns keccak265 encryption of a string.

Product_Creation_Event: Event of product creation.

Key_Creation_Event: Event of key creation.

product_details: The details of the product as a struct (example name, expiry_date etc.)

Decentralise: Decentralises data into our blockchain.

Pseudo Code:
if e = E **and** batch_quanity $>$ 0 **then**
— **if** Product_details_valid(product_details) **then**
—— Decentralise(product_details)
—— key_hash = keccak256(product_details, Random_string())
—— emit Product_Creation_Event
— **end**
— **else**
—— Show an error - ”Invalid Product details”
— **end**
end
else
— Show an error - ”Permission not granted”
end

- **Key Creation:** Key creation can be done in multiple ways . Some of them are generating a completely random key, generating the key on the basis of current timestamp or generating it on the basis of the product details. Once it is generated we are going to encrypt it using Keccak-256 encryption. Here our key will be generated in a hybrid way based on the product details, a randomly generated string and current timestamp in milliseconds since epoch. This will make it impossible to regenerate. Also this key will be in a bar code or a QR code or an NFC tag so that the receiver of the product doesn't have to put it manually for every entry.

Algorithm 2: Creation of key_hash

Input: e, E, block_number, product_details.

E: The Ethereum address of the current owner of the product.

e: Ethereum address from which the request of creating new key has been made.

block_number: The block number of current block (block.number).

keccak265: The function returns keccak265 encryption of a string.

product_details: The details of the product as a struct (example name, expiry_date etc.)

Product_details_valid: The function that returns if product details are valid or not.

Random_string: The function that returns a random string.

Pseudo Code:
if e = E **and** Product_details_valid(product_details) **then**
— key_hash = Random_string() + product_details + block_number
— key_hash = keccak256(key_hash)
— return key_hash
end
else
— Show an error - "Invalid Product details or Insufficient Permissions"
end

- **Key matching and product verification:** Key matching is done to verify product and to generate a new key for the next stage of the supply chain. Two strings are compared by encoding them using ABI (The Contract Application Binary Interface) "abi.encodePacked()" function and then again keccak256 encryption. So for example if we want to compare string **s1** with string **s2** then it'll be **"keccak256(abi. encodePacked(s1)) == keccak256(abi. encodePacked(s2))"** this piece of solidity code will return a boolean depending on if the strings are equal or not.

Algoritm 3: Key matching and product verification

Input: arb, e, E, physical_key, product_details.

arb: The Ethereum address of the arbitrator. **E:** The set of Ethereum address of the owner of the product.

e: Ethereum address from which the request of matching the key and verifying the product has been made.

physical_key: The key_hash obtained from the physical product.

product_details: The details of the product as a struct (example name, expiry_date etc.)

Product_details_valid: The function that returns if product details are valid or not.

Get_product_key: The function that returns the decentralised key of the specific product.

Product_Verification_Successful_Event: Event of product verification.

Product_Verification_Failed_Event: Event of product verification failure.

Create_key: Function that creates new key using Algorithm 2 and returns it.

Alert: Functions that alerts an address with some details.

keccak265: The function returns keccak265 encryption of a string.

abi.encodePacked: Function that encodes a string according to the Contract Application Binary Interface (ABI).

Compare_keys: Returns result of solidity code "keccak256(abi.encodePacked(s1)) == keccak256(abi.encodePacked(s2))"

Pseudo Code:
if e = E **and** Product_details_valid(product_details) **then**
— key_hash = Get_product_key(product_details)
— **if** Compare_keys(key_hash, physical_key) **then**
—— emit Product_Verification_Successful_Event
—— new_key_hash = Create_key(product_details)
—— Show success message - "Product successfully verified"
—— return new_key_hash
— end
— else
—— emit Product_Verification_Failed_Event
—— Alert(arb, product_details)
—— Show an error - "Product verification failed, please discard it and contact arbitrator!"
— end
end
else
— Show an error - "Invalid Product details or Insufficient Permissions"

end

- **Keccak-256 encryption:** It is a hashing technique that takes in bytes memory and returns bytes 32 hashed output. It is majorly used in cryptography and is a trusted encryption technique used worldwide.

All the algorithms defined can be implemented in solidity and provide the desired results 100 percent of the time. These algorithms are full-proof and have no faults or cases missed.

IV. Conclusion

Our proposed system for asset tracking system using blockchain will definitely prove to be a reliable and transparent cost effective compared to the current systems in the market. The decentralized system will ensure complete privacy and security of our users as well as their items. All the actions of all the people in the entire supply chain will be accounted for and all the activities will be logged in with the timestamp which will help us identify the exact time and place at which the product was tampered. Even the customers can trace the product right back to the manufacturer and ensure that they have received 100% original product. This is our attempt to eliminate the discrepancies in the current system and ensure customer satisfaction and create a tamper proof system.

References

[1] Wikipedia ”Bluetooth Low Energy”,September 2020 ,Available[Online],https://en.wikipedia.org/wiki/Bluetooth_Low_Energy

[2] Wikipedia ”Global Positioning System”,September 2020 ,Available[Online],https://en.wikipedia.org/wiki/Global_Positioning_System

[3] Wikipedia ”Radio-frequency identification”,September 2020 ,Available[Online],https://en.wikipedia.org/wiki/Radio-frequency_identification

[4] K. Toyoda, P. T. Mathiopoulos, I. Sasase, and T. Ohtsuki, “A novel blockchain-based product ownership management system (poms) for anti-counterfeits in the post supply chain,” IEEE Access, vol. 5, pp. 17 465–17 477, 2017.

[5] K. Biswas and V. Muthukkumarasamy, “Securing smart cities using blockchain technology,” in 2016 IEEE 18th International Conference on High Performance Computing and Communications; IEEE 14th International Conference on Smart City; IEEE 2nd International Conference on Data Science and Systems (HPCC/SmartCity/DSS), Dec 2016, pp. 1392–1393

[6] S. Singh and N. Singh, “Blockchain: Future of financial and cyber security,” in 2016 2nd International Conference on Contemporary Computing and Informatics (IC3I), Dec 2016, pp. 463–467.

[7] K. Christidis and M. Devetsikiotis, “Blockchains and smart contracts for the internet of things,” IEEE Access, vol. 4, pp. 2292–2303, 2016.

[8] Two party contracts. [Online]. Available: https://dappsforbeginners.wordpress.com/tutorials/two-party-contracts/

[9] W. Zheng, X. Wang and R. Kamoua, ”Personal asset tracking,” 2013 IEEE Long Island Systems, Applications and Technology Conference (LISAT), Farmingdale, NY, 2013, pp. 1-6, doi: 10.1109/LISAT.2013.6578230.

[10] G. H. Truelove, M. A. Foster, V. K. Kohli and T. G. Raslear, ”Real-time asset tracking and monitoring using low-cost cellular networks,” Proceedings of the 2006 IEEE/ASME Joint Rail Conference, Atlanta, GA, 2006, pp. 315-318, doi: 10.1109/RRCON.2006.215323.

[11] A. Mhlaba and M. Masinde, ”Implementation of middleware for Internet of Things in asset tracking applications: In-lining approach,” 2015 IEEE 13th International Conference on Industrial Informatics (INDIN), Cambridge, 2015, pp. 460-469, doi: 10.1109/INDIN.2015.7281778.

[12] I. Bisio, A. Sciarrone and S. Zappatore, ”Asset Tracking Solution with BLE and Smartphones: An Energy/Position Accuracy Trade-Off,” 2015 IEEE Global Communications Conference (GLOBECOM), San Diego, CA, 2015, pp. 1-6, doi: 10.1109/GLOCOM.2015.7417782.

[13] Sun-Jin Kim, Jung Hae Seo, Jonnalagadda Krishna and Sun-Joong Kim, ”Wireless sensor network based asset tracking service,” PICMET ’08 - 2008 Portland International Conference on Management of Engineering Technology, Cape Town, 2008, pp. 2643-2647, doi: 10.1109/PICMET.2008.4599893.

[14] J. Youn, H. Ali, H. Sharif, J. Deogun, J. Uher and S. H. Hinrichs, ”WLAN-Based Real-Time Asset Tracking System in Healthcare Environments,” Third IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob 2007), White Plains, NY, 2007, pp. 71-71, doi: 10.1109/WIMOB.2007.4390865.

[15] R. Kaur, V. Defrancesco and J. Enderle, ”Design, development and evaluation of an asset tracking system,” Proceedings of the IEEE 31st Annual Northeast Bioengineering Conference, 2005., Hoboken, NJ, 2005, pp. 110-111, doi: 10.1109/NEBC.2005.1431949.

[16] Feng Tian, ”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), Kunming, 2016, pp. 1-6, doi: 10.1109/ICSSSM.2016.7538424.

[17] Min, Hokey. (2018). Blockchain technology for enhancing supply chain resilience. Business Horizons. 62. 10.1016/j.bushor.2018.08.012.

[18] Saberi, Sara Kouhizadeh, Mahtab Sarkis, Joseph Shen, Lejia. (2018). Blockchain technology and its relationships to sustainable supply chain management. International Journal of Production Research. 1-19. 10.1080/00207543.2018.1533261.

[19] M. P. Caro, M. S. Ali, M. Vecchio and R. Giaffreda, ”Blockchain-based traceability in Agri-Food supply chain management: A practical implementation,” 2018 IoT Vertical and Topical Summit on Agriculture - Tuscany (IOT Tuscany), Tuscany, 2018, pp. 1-4, doi: 10.1109/IOT-TUSCANY.2018.8373021.

[20] S. Chan, A. Connell, E. Madrid, D. Park and R. Kamoua, ”RFID for personal asset tracking,” 2009 IEEE Long Island Systems, Applications and Technology Conference, Farmingdale, NY, 2009, pp. 1-7, doi: 10.1109/LISAT.2009.5031570.

[21] C. K. M. Lee, C. M. Ip, T. Park and S. Y. Chung, ”A Bluetooth Location-based Indoor Positioning System for Asset Tracking in Warehouse,” 2019 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM), Macao, Macao, 2019, pp. 1408-1412, doi: 10.1109/IEEM44572.2019.8978639.

[22] C. Kao, R. Hsiao, T. Chen, P. Chen and M. Pan, ”A hybrid indoor positioning for asset tracking using Bluetooth low energy and Wi-Fi,” 2017 IEEE International Conference on Consumer Electronics - Taiwan (ICCE-TW), Taipei, 2017, pp. 63-64, doi: 10.1109/ICCE-China.2017.7990996.