Available online at www.eccomasproceedia.org
Eccomas Proceedia SMART (2023) 1173-1184

ECCOMAS
Proceedia

X ECCOMAS Thematic Conference on Smart Structures and Materials
SMART 2023
D.A. Saravanos, A. Benjeddou, N. Chrysochoidis and T. Theodosiou (Eds)

ENHANCING DEFECT TRACEABILITY AND DATA INTEGRITY IN INDUSTRY 4.0 USING BLOCKCHAIN TECHNOLOGY

ANDREANA MITSIAKI[*], NIKOLAOS DIMITRIOU[*], GEORGE MARGETIS[†], KONSTANTINOS VOTIS[*], DIMITRIOS TZOVARAS[*]

[*] Information Technologies Institute, Centre for Research and Technology Hellas, 57001, Thessaloniki, Greece
e-mail: amitsiaki@iti.gr, nikdim@iti.gr, kvotis@iti.gr, Dimitrios.Tzovaras@iti.gr, www.certh.gr

[†] Institute of Computer Science, Foundation for Research and Technology Hellas, Heraklion, Crete, GR-70013, Greece
email: gmarget@ics.forth.gr, www.forth.gr

Abstract. With the transition to Industry 4.0 factories have achieved significant gains in production with respect to quality, reliability, flexibility, and utilization of resources. Nonetheless, there are open challenges that shall be addressed when it comes to traceability of defects and automation actions that are interrelated with both optimization and security aspects along the production chain. Current industrial solutions use a centralized client-server architecture; however, if the central authority is undermined, the system can fail. In this regard, one of the most promising technologies is blockchain, as it is a decentralized technology based on a peer-to-peer network instead of a client-server model. In this paper, we propose a blockchain framework based on Ethereum platform, which is applied in three different production lines namely for antenna manufacturing, microelectronics, and elevators. Initially, we have three different private Ethereum networks, for each factory, with the Proof-of-Authority consensus mechanism. We have developed smart contracts for defect detection as well as firmware update for production equipment. As shown through experiments, the developed smart contracts have certain advantages compared to existing practises in terms of traceability and cyber-security. Furthermore, we have developed a blockchain API that connects the proposed framework with an industrial middleware platform and the overall OPTIMAI Industry 4.0 ecosystem. Experiments for each production line showcase the potential of our approach and it gains in terms of security, storage, traceability, and transparency.

Key words: Blockchain, Smart Contracts, Industry 4.0, Private Ethereum, Defect Detection, Firmware Update

1 INTRODUCTION

Industry 4.0 revolutionizes manufacturing, enhancing flexibility, mass customization, quality, and productivity [7]. Numerous applications based on Industry 4.0 have been implemented worldwide as a result of the widespread use of the Internet and related

© 2023 The Authors. Published by Eccomas Proceedia.
Peer-review under responsibility of the organizing committee of SMART 2023.
doi: 10.7712/150123.9866.443273

technologies, in which sensors and actuators perceive, process, and transmit data for industrial automation. Data between multiple sites can flow through either an open channel, such as the Internet, or internal networks in Industry 4.0-based applications. Regardless of the type of channel used, security and privacy threats have multiplied, highlighting the need for robust security measures. Since these applications work with large amounts of data, it is important to consider data heterogeneity, data integrity, and data redundancy, in addition to security and privacy issues. In addition, various applications require datasets from various disciplines in various formats. As a result, standardization of the data format is also necessary to enable its use by various Industry 4.0-based applications [5]. A major difficulty in Industry 4.0 is ensuring privacy, confidentiality, and integrity due to the volume of data exchanges over the Internet [6].

The Internet of Things (IoT), especially the Industrial IoT (IIoT), has developed rapidly and attracted much attention in the academic and business worlds, yet IoT privacy threats and security vulnerabilities arise due to a lack of basic security technology. Due to its decentralization and disclosure of information, the blockchain technique has been presented as a decentralized and distributed approach to meet the security needs and stimulate the development of IoT and IIoT. There are still significant technical issues that need to be resolved in order for the IoT and IIoT to develop rapidly, including interoperability challenges, security vulnerabilities, and a lack of data analysis and transmission. The need to address security vulnerabilities is one of the most pressing issues [8]. Addressing security vulnerabilities is an urgent concern in the realm of Industrial Internet of Things (IIoT). Blockchain technology offers several significant advantages in this regard, which we will elaborate on below.

The immutability of Blockchain technology, which fosters transparency and reliability, is one of its main advantages. Once data is recorded in the chain, it cannot be modified or removed, giving stakeholders access to it. As a result, the immutability of Blockchain can be used to document all the processes that must be taken to get at a specific result, such as a transaction or even a piece of regulation, ultimately giving all parties involved an unquestionable audit trail.

The traceability afforded by blockchain technology's framework embodies the potential for regulation and responsibility. Through its guarantee of correctness and immutability, distributed ledgers provide an ideal of knowing the beginnings and destinations of people and things, accurately and securely recording historical records of transactions. This enables investigations and promotes accountability and regulation [13].

In this paper, we want to highlight the blockchain possibilities in the OPTIMAI project, which is referred to throughout the paper [14]. The blockchain framework is being used in three factories to follow the benefits of Industry 4.0 while attempting to solve problems related to traceability and cyber-security.

- Data quality and reliability: Blockchain provides a tamper-proof record of all transactions, ensuring data accuracy and reliability over time.
- Blockchain technology can increase accountability for firmware authenticity by providing a tamper-proof and transparent record of all firmware changes and updates. This can ensure that the firmware has not been modified, providing assurance to users and mitigations potential security risks.

- Traceability of a product: With blockchain, every transaction is recorded in a block, which is then cryptographically linked to the previous block, forming a chain of blocks. This creates a permanent and transparent record of all transactions that cannot be altered or deleted. This enables stakeholders in the supply chain, such as manufacturers, suppliers, distributors, and customers, to have complete visibility into the journey of the asset, providing them with the confidence that the asset has been handled properly and that its quality and authenticity have been maintained.

2 BLOCKCHAIN IN INDUSTRY 4.0

Blockchain technology is a decentralized database that, compared to traditional database models, is P2P, not based on the client-server scheme; there is no central authority that has all the control, but it is made up of peer nodes that take the network decisions. The blockchain can be divided based on who has the right to participate in the network. If everyone can, then it is public (permissionless), while if only specific users can participate, then it is private (permissioned). A very important feature of the blockchain is that all participating nodes must agree on any transaction being added to the blockchain. This is achieved by the consensus mechanism. PoW is one of the most well-known consensus mechanisms, but it requires a massive amount of computing power and has environmental consequences in terms of energy consumption. Another consensus mechanism, which we use in this paper, is PoA. It can be used in private networks (e.g., private Ethereum), and the participants are known in this network.

Industry 4.0 technology can benefit the manufacturing industry in many ways, but there are several obstacles that must be overcome in order to fully reap these rewards. Security, trust, dependability, traceability, and greater value chain integration are a few of these difficulties, which can be addressed with the blockchain technology [2].

Blockchain technology records exchanges and transactions in a secure and transparent decentralized ledger, enabling traceability and reducing human error and delays. This improves the supply chain industry's transparency, dependability, and efficiency through data accessibility and immutability [4].

The research in [9,10] suggested a blockchain platform for IIoT. Using a decentralized, trusted, peer-to-peer network for IIoT applications, this platform, which has smart contracts deployed, enables creation of various distributed industrial applications. Additionally, the massive computational and storage capabilities of Industry 4.0 technologies are recognized, verified, and connected via a cloud server. However, these IoT solutions are quite expensive due to integrated cloud providers' high storage and operating expenses, and these expenditures are rising significantly as IoT, and wired computers become more prevalent. A transparent technology like blockchain can quickly resolve these issues. Putting in place a structured peer-to-peer connectivity paradigm would save resources and decrease costs by eliminating the need to build and maintain large, consolidated data centres, which are currently required to handle the enormous amounts of transactions between linked devices. Due to the transparency of the blockchain, all parties involved will be able to access and manage the information relevant to all production phases [1].

The authors in [3] suggest that the blockchain technology can be utilized to manage and coordinate product development strategies, capabilities and information securely, across the entire product lifecycle, from ideation to disposal. This would provide stakeholders with real-time visibility into how the product is being utilized, without having to wait for consumer feedback. Blockchain can also help address issues related to transparency in supply management, by enabling tracking, trade, contracts and payment.

The [11] paper explores the use of blockchain technology for quality assurance in smart manufacturing. The paper elaborates on how blockchain can facilitate equipment management, improve transaction efficiency, and retrieve material provenance.

The authors in [12] highlights the importance of traceability in multi-tiered manufacturing to promote transparency and quality guarantees. It is proposed a blockchain-based traceability framework to address challenges such as data tampering and lack of information exchange. The framework includes interaction and network architecture at the organization level, as well as smart contracts and transaction validation standards at the operation level. The proposed solution utilizes distributed ledger technology to store and authenticate supply chain transaction.

3 BLOCKCHAIN IN THE OPTIMAI ECOSYSTEM

3.1 Smart contract for defect detection

This paper presents a proposal for utilizing blockchain technology in three different industries, namely in microelectronics, communications and elevators manufacturing,) under the OPTIMAI ecosystem. The aim is to store and retrieve data collected on the production line using Ethereum-based private blockchains, each of which is dedicated to a specific factory. As it is described in [14], each factory's products go through quality control, where sensors and AI methods analyze the objects to detect defects. If a defect is detected, the data is sent to the middleware, which sends a JSON file to the appropriate endpoint IP of the Blockchain API. The Blockchain API then sends the data from the JSON file to the smart contract, which is deployed on the private Ethereum blockchain. We utilized Geth, from “Go Ethereum” a tool used by Ethereum, to set up the Ethereum nodes and created three privates Ethereum blockchains, one for each factory, as each factory's data is private and cannot be shared with others.

In private Ethereum, real cryptocurrencies are not needed; however, in order to be able to deploy a smart contract, funds are needed. For the sake of acquiring a node’s funds, it will either have to mine or, as preferred in this study, to pre-fund the accounts using Puppeth’s relevant option. Puppeth is an Ethereum command-line tool for creating private Ethereum. With Puppeth still running on top of Geth, we have created the genesis file and all the essentials for our network, such as the network ID.

As we mentioned, the middleware will send data to the blockchain, in JSON format. In the Blockchain, it is needed to be a third-party service that will provide the data to the smart contract on a blockchain. That third-party is called an Oracle, as a blockchain operates in a decentralized, trustless environment, and therefore cannot access data from the outside world without an

oracle. In our project, we use a Node.js back-end server as an oracle. The Node.js back-end server can retrieve data from external sources, process it, and then pass it on to smart contract as input. So, for the middleware to be able to send data to the blockchain, we have developed an API, using Node.js as a backend server. Node.js server, receive and respond to HTTP requests from the Middleware, and interact with the deployed Smart Contract through web3.js library, using the address and ABI (Application Binary Interface) of the smart contract. ABI is a specification that defines the interface of a smart contract and provides a way for external entities to interact with a smart contract. The smart contract is linked to a private Ethereum instance. We have created an account with a unique address in the private Ethereum from Geth, which we have pre-funded, so that we can deploy smart contracts from this address.

As can be seen in the image below, the objects of interest in each factory went through quality control. Specifically, sensors (e.g., cameras) are placed on the production line, and AI methods are used to analyze the object to see if there is any defect (e.g., a crack in the antenna). The data is then sent to the middleware, which will send a JSON file to the Blockchain API at the appropriate endpoint IP if a defect is detected. The Blockchain API with the Web3.js library will send the data from the received JSON to the smart contract. We've also connected the private Ethereum blockchain we've built to the Blockchain API, with Web3.js. When data is sent to the smart contract that has been deployed from the private Ethereum blockchain, the user (in the case of OPTIMAI the middleware) receives the hash of the newly created block as a response along with other information such as its number block.

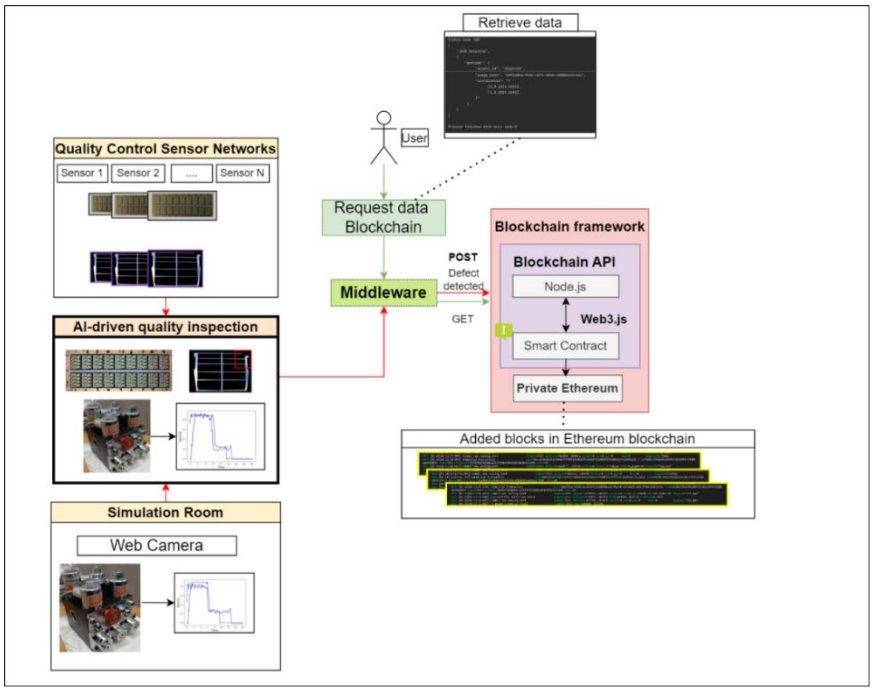

Figure 1: The data of the AI-driven quality inspection of a defective product are sent to the blockchain API to the smart contract and stored to the private Ethereum

3.2 Smart contract for firmware update

As the use of technology and the Internet of Things (IoT) continues to expand, it is increasingly important to develop secure systems for releasing and updating firmware. In this paper, we propose a method for ensuring the authenticity of firmware updates by using blockchain technology to securely store hashes of the updates. We use Blockchain to verify the authenticity of firmware by securely storing the hashes of firmware updates, which we can use to verify that the firmware has not been altered or tampered with.

Our approach involves several steps:

1. Generation of hashes: The smart contract generates a hash of the firmware using a cryptographic (keccak256) hash function. Keccak256 is a cryptographic hash function in the Ethereum Virtual Machine (EVM). It is used to generate a hash of an input data to be stored on the blockchain. The purpose of using keccak256 is to provide data integrity, security and consistency by ensuring that the output hash of the function is unique for given input, making it infeasible for any malicious user to modify the input data without being detected.
2. Authorization for firmware updates: In the Smart Contract only authorized users can add a new firmware update (e.g., the firmware developer, the smart contract owner) by giving the version of the Firmware, a hash (in a string format and the Smart Contract will cryptography it with keccak256), and the Ethereum address.
3. Secure hash storage: The hash is securely stored in the blockchain, through a Smart Contract, with a timestamp and any other relevant metadata.
4. Verification during firmware update: The Smart Contract, compares the hashes of the firmware updates. If the hashes match, the firmware is verified as authentic and can be installed. In the event that the firmware has been altered or tampered with, the hash of the firmware would no longer match the hash stored in the blockchain ledger. This would indicate that the firmware is not authentic and should not be installed.
5. Transparent hash comparison: The hash of the firmware update is stored in the blockchain and can be queried by any device that needs to verify the authenticity of a firmware update. The smart contract provides a secure and transparent way to compare the hash of the firmware update with the hash stored in the blockchain, ensuring that only authentic firmware updates are installed.
6. Querying the blockchain: The device or the user, queries the blockchain to retrieve the hash of the firmware stored in the blockchain.

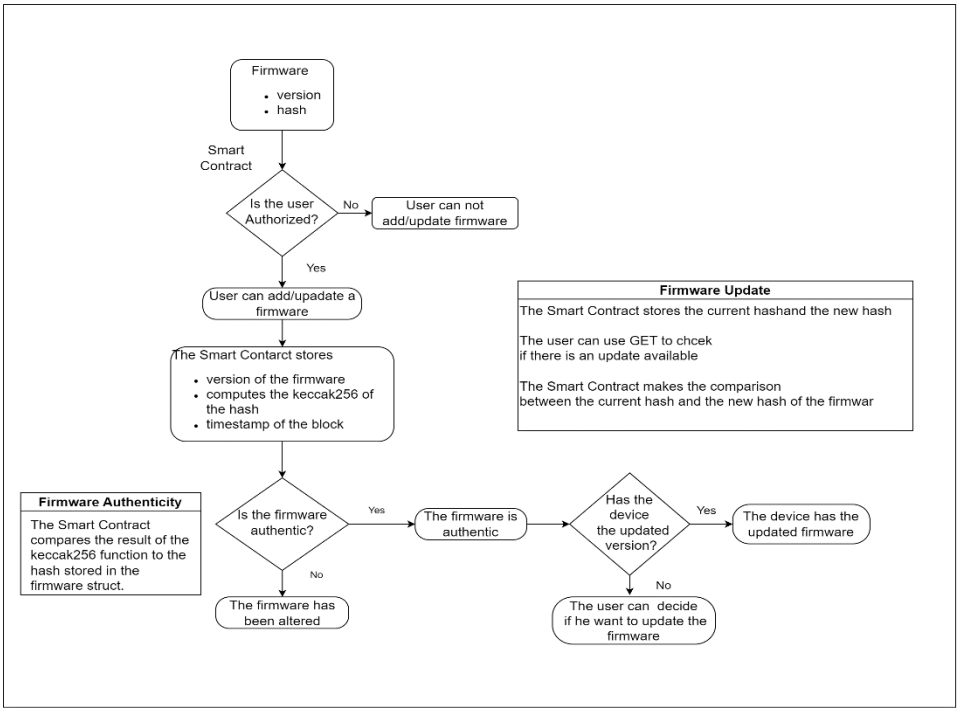

Figure 2: The data flow of the Firmware authenticity and the Firmware updates

By using blockchain technology to verify the authenticity of firmware updates, organizations can reduce the risk of security vulnerabilities and improve the overall security of the devices and the network. This approach provides a transparent and secure way to ensure that only verified and secure firmware is installed, which is essential for protecting against potential cyber threats.

4 EXPERIMENTAL RESULTS

In this paper, we conducted experiments to evaluate the performance of a smart contract system form storing defects and calibration data in the manufacturing process. We developed three different smart contracts for PCB defect detection, antenna defect detection, and elevators for calibration, and tested the system by sending POST requests with different sizes of JSON to a smart contract on a private Ethereum network. The network was configured to create a new block every 15 seconds. We have sent 3 different JSON file sizes, namely 1,57KB, 396B, and 150B, and recorded the response times for each request. It's worth noting that the experiments were conducted under similar conditions for all three smart contracts, and the results were similar across the board. The results of the experiments showed that the response times were similar for all three JSON file sizes, with an average response time of 15-20 seconds. These results suggest that the block time of the private Ethereum network may have had some impact on the response times of the smart contract, but it did not have a significant impact. However,

it's worth noting that other factors, such as the complexity of the smart contract and the computing power of the network nodes, may still impact the performance of the smart contract. When a POST request is sent to the smart contract, the smart contract needed to execute the code and update the state of the blockchain. This process takes time, and the time it takes can be impacted by the above various factors. So while the block time may have had some impact on the response times we observed, it may not have had a significant impact because the smart contract was able to execute and update the blockchain state within the 15-second block time. Overall, these results indicate that sending POST requests to a smart contract on a private Ethereum network can be a reliable and efficient way to interact with the blockchain, even with a block time of 15 seconds.

Overall, our experimental results demonstrate that our proposed smart contract system is a viable solution for storing defects and calibration data in the manufacturing process. The system is secure, efficient, and able to handle large amounts of data, making it suitable for use in various manufacturing settings.

The following table presents simulation times for a Blockchain system that stores JSON data with three different sizes. The private Ethereum network used in the experiment created a new block every 15 seconds using the Clique client. The table includes results for different numbers of POST requests with and without delay. It is important to mention that the experiments yielded similar results across all three smart contracts, leading us to provide a general table that summarizes the results for all the experiments.

Table 1: Request and time of the response

	JSON 1.57KB				JSON 396B		JSON 150B	
Number of POST requests	Without delay	15s delay	25s delay	35s delay	Without delay	15s delay	Without delay	15s delay
5	1.28m	2.24m	2.41	3.31m	1.06m	2.15m	1.2m	2.15m
20	5.05m	9.51m	9.59m	14.56m	4.82m	9.22m	4.94m	9.35m
50	12.4m	24.53m	25.06m	37.34m	12.2m	24.3m	12.36m	24.42m
100	25.56m	49.88m	50.11m	74.9m	25.2m	49.02m	24.87m	48.8m

For 5 requests, the results showed that the times for each request corresponded to around 15 seconds, as expected. Each block that was created during this time went through the phases of committing new sealing work, submitting the transaction, successfully sealing the new block, and mining the potential block.

The results for 20, 50, and 100 requests also showed a similar pattern, with longer simulation times due to the increased number of requests. The delays introduced for each request resulted in longer simulation times as well.

The results demonstrate that the proposed Blockchain system can handle a range of JSON data sizes and numbers of requests with reasonable simulation times. The delay introduced for each request also had a significant impact on the simulation time, highlighting the importance of carefully considering the network parameters when designing and implementing Blockchain-based systems.

The use of the below smart contracts can provide transparency and traceability in various manufacturing settings. By storing important information related to defective products in an immutable smart contract, stakeholders can access and verify the information at any time, promoting transparency and building trust between manufacturers, consumers, and other supply chain members.

The smart contract can capture and store critical details such as the date of defect identification (block timestamp), the number of defects, and other relevant data. This information can be used to track the product's history and identify the initial manifestation of a defect, even in previous manufacturing stages. Additionally, the smart contract's ability to provide an immutable record of the product's history can be invaluable in situations where a product recall is necessary or where liability needs to be established, as it can aid in identifying the source and extent of the problem.

4.1 Smart contract for PCB defect detection

The data for the PCB (printed circuit board) defect detection smart contract, were collected from the manufacturing line where PCB was produced. Each PCB in the manufacturing line consisted of 18 identical circuit modules. AI methods, as descripted in the paper [15], were utilized to analyze the data and detect any defects present in the individual modules. In the smart contract, we provide the PCB ID, module IDs ranging from 0 to 17, and the numbers of defects identified for each module (e.g., {id: 0, defects: 3}, {id: 1, defects: 4}, etc.). The image below provides an overview of the process involved in detecting defects in PCBs. The PCB is subjected to sensors such as cameras on the production line, and AI methods are utilized to analyze it for any defects. Modules with defects are identified by marking them with red boxes (e.g., excess glue), while modules without any defects are marked with green boxes. Middleware receives the data and sends a JSON file to the blockchain API containing the PCB ID and module IDs with the number of their defects via a POST request. The blockchain API, in turn, sends the data to the smart contract via Web3.js. The smart contract receives the data and stores it on the private Ethereum blockchain. Upon successful storage of the information, the Middleware receives a message containing the hash of the block, block number, etc. Users can retrieve information about the defective PCBs by making a GET request with the PCB ID.

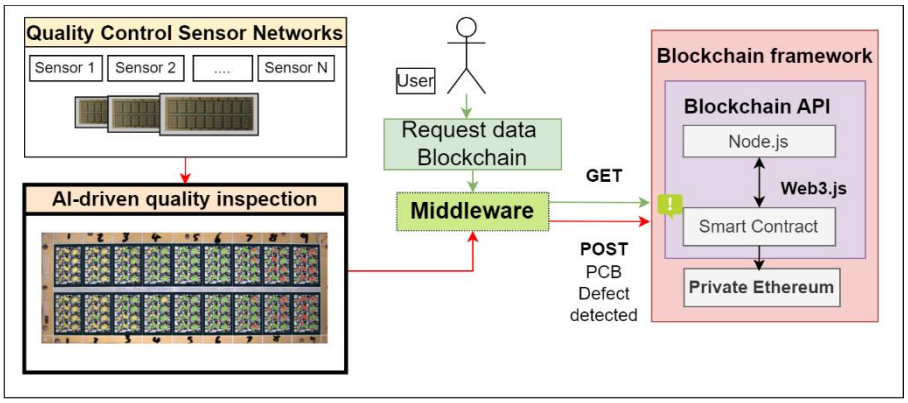

Figure 3: The data of a PCB with defects are stored in the private Ethereum.

4.2 Smart Contract for antenna defect detection

The smart contract for antenna defect detection utilizes data collected from the manufacturing line where antennas are produced. The antennas are subjected to sensors such as cameras on the production line, and AI methods are utilized to analyze them for any defects. The data is then transmitted to the smart contract through the Middleware and to the blockchain API, and stored on the private Ethereum blockchain. When submitting data to the smart contract, the user should provide the ID of the antenna, the hash of the antenna image (not the image itself), the number of defects identified, and a square of coordinates for each detected defect ([x1, y1, x2, y2]). Middleware receives the data and sends a JSON file to the blockchain API via a POST request, which in turn sends the data to the smart contract via Web3.js. Upon receipt, the smart contract stores the data on the private Ethereum blockchain, and the Middleware receives a message containing the information of the block that has been created. Users can retrieve the information by making a GET request with the antenna ID.

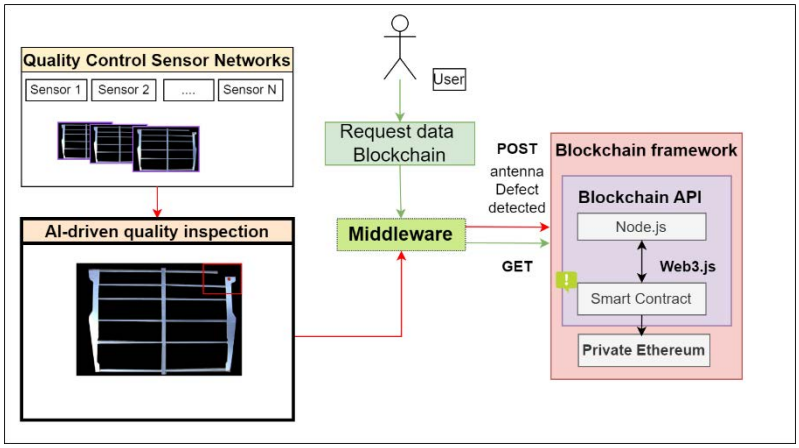

Figure 4: The data of a defective antenna are stored in the private Ethereum

4.3 Smart Contract for elevators for calibration

Calibrating the parameters of an industrial machine is important as it affects the quality of the production process. In the context of the factory that manufactures elevators, the calibration of hydraulic Elevator Control Valves (EVs) is a critical stage. Maladjustment of the EVs can cause unstable oration, sudden stops, and even damage to the machine. Manual calibration of the valve block is time-consuming and may lead to inaccurate results due to human factors. To address this, we conducted tests on the movements of the block valve in the elevator system. To analyze the performance of the valve, we divided the tests into multiple time segments of fixed duration. For each segment, we calculated the Normalized Root Mean Square Error (NRMSE) and the total movements made by the valve. To store the results of these tests on the private Ethereum blockchain, we divided the counterweights into individual sections of fixed time duration. For each section, we calculated the NRMSE and the total movements made by each valve. We stored this information on the blockchain, allowing for secure and tamper-proof storage of the test results. In the smart contract, we provide the id, the episode_id, (we may

have more than one test, each test has its own id), and the segments [valve, nrmse] In this case, because an id can have different episodes (tests), in order to retrieve the information, the user need to make a get request with the id and the episode_id.

5 FUTURE WORK

In future work, we plan to develop a decentralized application (dApp) for firmware updates and for the firmware authenticity. The dApp will be designed with a user-friendly interface to make it easy for users to initiate firmware updates and authenticity. Additionally, we aim to integrate the blockchain API with our smart contracts for defect detection to enable decision-making support. This integration will provide a more efficient and transparent system for detecting defects in the manufacturing process. We believe that this integration will result in a more reliable and secure system for quality assurance. Overall, these improvements will help to streamline the manufacturing process and enhance the quality of our products.

6 CONCLUSIONS

In conclusion, this paper proposes a blockchain-based framework for defect detection and firmware update in three different production lines. The proposed framework employs the Ethereum platform with a Proof-of-Authority consensus mechanism and smart contracts for defect detection and equipment firmware update. The experiments conducted in each production line have shown that the proposed framework has significant advantages over existing practices in terms of traceability, security, storage, and transparency. Furthermore, the developed blockchain API connects the framework with an industrial middleware platform and the OPTIMAI Industry 4.0 ecosystem. The results of this research suggest that the proposed blockchain-based framework has the potential to improve the current practices in the manufacturing industry, enhance the reliability of production equipment, and ensure the traceability of defective products throughout the supply chain. Therefore, the proposed framework can contribute significantly to the advancement of Industry 4.0 and its related technologies.

ACKNOWLEDGMENT

This work was supported by the European Commission through Project OPTIMAI funded by the European Union H2020 programme under Grant 958264. The opinions expressed in this article are those of the authors and do not necessarily reflect the views of the European Commission.

REFERENCES

1. Javaid, Mohd, et al. "Blockchain technology applications for Industry 4.0: A literature-based review." *Blockchain: Research and Applications* (2021): 100027.

2. Mohamed, Nader, and Jameela Al-Jaroodi. "Applying blockchain in industry 4.0 applications." *2019 IEEE 9th annual computing and communication workshop and conference (CCWC)*. IEEE, 2019.
3. Mushtaq, Anum, and Irfan Ul Haq. "Implications of blockchain in industry 4. o." *2019 International Conference on Engineering and Emerging Technologies (ICEET)*. IEEE, 2019.
4. Alladi, Tejasvi, et al. "Blockchain applications for industry 4.0 and industrial IoT: A review." *IEEE Access* 7 (2019): 176935-176951.
5. Bodkhe, Umesh, et al. "Blockchain for industry 4.0: A comprehensive review." *IEEE Access* 8 (2020): 79764-79800.
6. Fraga-Lamas, Paula, and Tiago M. Fernández-Caramés. "A review on blockchain technologies for an advanced and cyber-resilient automotive industry." *IEEE access* 7 (2019): 17578-17598.
7. Silvestri, Luca, et al. "Maintenance transformation through Industry 4.0 technologies: A systematic literature review." *Computers in industry* 123 (2020): 103335.
8. Wang, Qin, et al. "Blockchain for the IoT and industrial IoT: A review." *Internet of Things* 10 (2020): 100081.
9. Teslya, Nikolay, and Igor Ryabchikov. "Blockchain-based platform architecture for industrial IoT." *2017 21st Conference of Open Innovations Association (FRUCT)*. IEEE, 2017.
10. Bahga, Arshdeep, and Vijay K. Madisetti. "Blockchain platform for industrial internet of things." *Journal of Software Engineering and Applications* 9.10 (2016): 533-546.
11. Zhang, Yongping, et al. "Blockchain-based trust mechanism for IoT-based smart manufacturing system." *IEEE Transactions on Computational Social Systems* 6.6 (2019): 1386-1394.
12. Agrawal, Tarun Kumar, et al. "Blockchain-based framework for supply chain traceability: A case example of textile and clothing industry." *Computers & industrial engineering* 154 (2021): 107130.
13. Thylstrup, N. B. & Archer, M. & Ravn, L. (2022). Traceability. *Internet Policy Review*, 11(1). https://doi.org/10.14763/2022.1.1646
14. Margetis, George, et al. "Aligning Emerging Technologies onto I4. 0 principles: Towards a Novel Architecture for Zero-defect Manufacturing."
15. Evangelidis, Apostolos, et al. "A Deep Regression Framework Toward Laboratory Accuracy in the Shop Floor of Microelectronics." *IEEE Transactions on Industrial Informatics* 19.3 (2022): 2652-2661.