agroString 2.0: A Distributed-Ledger based Smart Agriculture Framework to Ensure Transparency in Food Delivery

Sukrutha L. T. Vangipuram
Computer Science and Engineering
University of North Texas, USA
Email: lt0264@unt.edu

Saraju P. Mohanty
Computer Science and Engineering
University of North Texas, USA
Email: saraju.mohanty@unt.edu

Elias Kougianos
Electrical Engineering
University of North Texas, USA
Email: elias.kougianos@unt.edu

***Abstract*—Consuming healthy food is one of the major concerns in the current and coming decades, as the agricultural food supply chain is confined to a lot of wastage during retail and unprepared storage acts. The conventional and zero tracking and communication systems towards their supply chain participants are some of the many reasons for damaged food delivery to consumers. With the aid of the Internet-of-Agro-Things (IoAT), the state of the food is being gathered at different supply chain stages to monitor and keep visibility of the agricultural product stored and transmitted. However, data tampering can be an issue with these IoAT devices as they are more prone to hacking and vulnerabilities, leading to data security and reliability problems. The current paper overcomes the traditional storage platform limitations in the supply chain. In this paper, we have collected the temperature and humidity data from the IoT-Edge device and sent the statistics directly to Distributed Ledger with Masked Authenticated Message (MaM) and called it agroString 2.0. Our previous work delivered the supply chain statistics temperature and humidity data through the private Blockchain Corda. Here in agroString 2.0, with the help of the distributed ledger, we bring aspects of data security into the supply chain domain with zero cost and faster transaction times for data.**

***Index terms*—** Smart Agriculture; Internet-of-Agro-Things (IoAT); Distributed Ledger; Blockchain; Food Supply Chain; Cybersecurity.

I. Introduction

Agriculture is the primary means of income for farmers and is an essential food source for the global population, that include crops, livestock, poultry, and forestry. The current global population remains at 7.6 billion and is estimated to grow to 9.8 billion in 2050 and 11.2 billion in 2100 [1]. While delivering food to the consumers from agricultural fields, many stakeholders reside in the center for maintaining storage, retail, and transport of agricultural produce. About 40% of the food is wasted and damaged during the retail and storage process [2]. Modern types of equipment and intelligent systems, such as the Internet-of-Agro-Things (IoAT), are used in smart agriculture to keep track of the crop or food's condition to avoid wastage [3]. However, these devices are more vulnerable to security and privacy attacks, resulting in tampered data leading to challenges in smart agriculture. Conventional platforms with central storage systems for storage can have latency and bottleneck limitations in storing the agricultural supply-chain data. Multiple platform equipment gives diverse data with origin and integrity questions, leading to authenticity issues; bringing this heterogeneous information under one system is a more significant challenge under food supplychain. Other issues in data storage and transmissions include latency, access control, privacy breach, single-point failure, data flow, and internet connectivity.

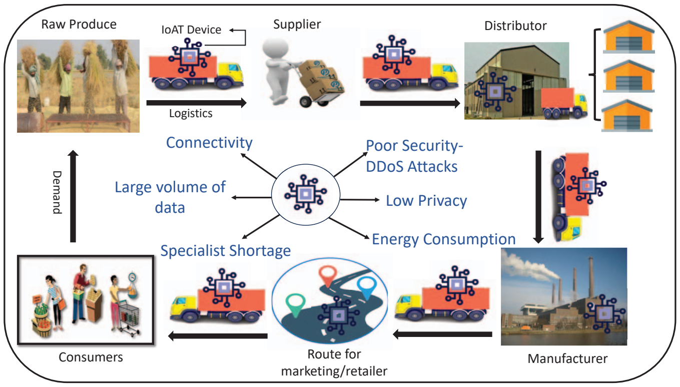


Fig. 1: Challenges of IoAT in Supply Chain.

In Fig. 1, we show some of the challenges IoAT possesses in the supply chain. Various applications are being introduced in healthcare and agriculture to detect and share patient readings, crop diseases, and livestock tracking. Decentralized storage techniques and cryptographic methods are utilized to secure the data coming from sensing devices to make healthcare and agricultural statistics more reliable and robust for research and study purposes and maintain privacy of the individual [4], [5]. One of the frameworks for data security is the distributed ledger (e.g. IOTA Tangle) that follows the Markle tree design for generating unique roots and hashes [6]. Cybersecurity methods help in protecting data from various attacks and can be helpful in multiple fields, including agriculture. Unlike Blockchain, which takes high fees and time for data transactions, distributed ledger (e.g. IOTA Tangle) charges zero fees and lesser time for information exchanges.

979-8-3503-5823-0/23/$31.00 ©2023 IEEE

The paper has the following order: Some of the traditional central storage systems and blockchains have limitations such as data latency, energy consumption and implementing high fees that are discussed in Section II and new provisions to the traditional systems. Section III explains some of the previous works of data storage in the supply chain and how our current system adds new functionalities overcoming them is detailed. We present a novel architecture for the current agroString 2.0 application with a Distributed Ledger technology using MaM authentication service in Section IV, explain the application's data flow and propose algorithms for the current system. The overall results, performance, and implementation of the system are shown and explained in Section V and ending with conclusions and future directions in Section VI.

II. Sate-of-the-Art Contributions

A. Problems discussed

- Food Hygiene Issues resulting from Food distribution.
- Security of Agricultural data in IoAT systems.
- Conventional storage system limitations.
- Blockchain Decentralized data storage with High fees and Time.
- Authenticity of data.

B. Proposed Ideas in the Current Paper

- Distributed Ledger security for Internet-of-Agro-Things (IoAT) data in supply chain.
- Authenticity of data through Distributed Ledger System.
- Evading central storage in supply-chain.
- Proposing visibility and provenance design to end consumers in the food supply string.

C. State-of-the-art Solutions

- Proposed a ledger storage architecture with IoAT Edge device to avoid central and cloud limitations.
- Implementing the IOTA Tangle with Masked Authenticated Messaging (MaM) for data integrity and validity.
- Showing results with IoAT Edge device and Distributed Ledger-IOTA Tangle with MaM.
- Comparing the results with Public and private distributed ledger systems to the current System.

III. Related Prior Works

Supply chain traceability systems give the ability to access all the data regarding a product from the beginning to the end until it reaches the hands of the consumer. Traceability is also helpful in tracking and tracing the food products in the supply chain to ensure the safety of agricultural produce. The Internet-of-Agro-Things (IoAT) is proving beneficial in making the traceability and visibility of food products practical. But, securely moving this IoAT data is becoming a more significant challenge. Some studies include sending the sensor data to central storage and public and private blockchains, as discussed in this Section.

The work [7] uses Radio-frequency technology- RFID for the entire supply chain system for collecting product data and sending the data to the central database platform. The total information regarding food products can be retrieved using the design of a dynamic query platform and mobile terminal. The work [8] uses blockchain technology to add confidentiality, authentication, and integrity to the narrow band of Internet of Things data collected while transporting food produce. The system takes advantage of the blockchain platform and implements the sha256 secure algorithm on the supply chain information to make the system more reliable and robust. In the system [9], the authors talk about the traceability system based on blockchain technology for storing information about food products. It engages a dual storage system for on-chain and off-chain data to reduce the information load and prevent blockchain's latency and cost drawbacks. It also provides tamper-proof and decentralization characteristics in the supply chain system. The supply chain in [10] designs a corDapp application with a Corda private blockchain that uses an IoAT edge device for collecting information regarding the food product for traceability and visibility toward end customers. The system implements food traceability with integrity and security to increase data quality and reduce food waste.

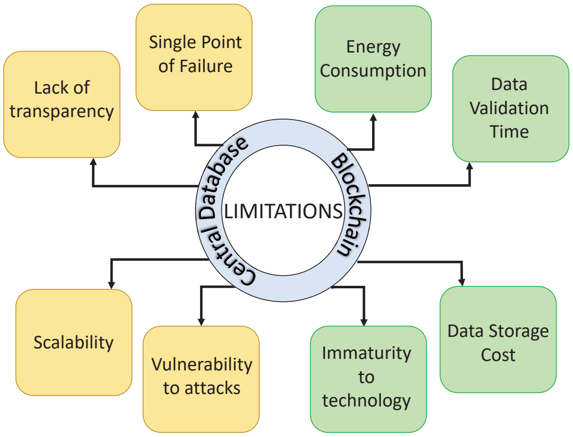


Fig. 2: Central and Blockchain Storage limitations.

The above approaches use centralized storage in some form or other that can lead to opaque information, untrustworthy data, and easy creation of information islands. Although the application built with private Corda masters the limitations of central and public blockchains, as shown in Fig. 2, it has its own disadvantages in the areas of familiarity with the platform, setting up the network, and expensive software. In the current paper, we overcome the above-discussed drawbacks and increase the level of security and integrity while implementing traceability and visibility in the agricultural food supply chain with cost free data transactions. The Table I compares prior works to the current application.

IV. Proposed State-of-the-Art Architecture and Proposed Algorithms

Each supply chain logistics receives the food products from the farmers through air, land, or sea transportation. Traceability and visibility can be achieved using IoAT devices at the edge

TABLE I: Previous works of supplychain storage in smart farming.

Paper	Storage Technology	Security Level	Computation
Zheng et al. [7]	Centralized	High Privacy Breach	Very High near client
Mohammed and Chopra. [8]	Decentralized-Blockchain	High	Very High near client
Yang et al. [9]	Partially Centralized	High	High near client
Vangipuram et al. [10] (agroString)	Decentralized-Corda	High	High near client
agroString 2.0 [Current-Paper]	**Distributed Ledger-IOTA Tangle**	**High**	**Very Low**

points between the supply chain stakeholders. In Fig. 3, a novel architectural design is presented where an information center maintains each participant's information and is communicated through the Internet. An Internet-of-Agro-Things (IoAT) resides on edge programmed with Distributed Ledger-IOTA Tanglealong with Masked authenticated Messaging. The sensor data from the IoAT moves toward the Ledger and is received by the end customer without being tampered with. Inside the edge layer, the Ledger acts as a gateway between the IoAT sensors and servers. The Distributed Ledger avoids double spending and other attacks that are more vulnerable in central storage systems. The supply chain sensor data is encrypted through Masked Authenticated Messaging in Ledger to secure the data and provide traceability and provenance of the food products. Fig. 3 shows that the tangle hashes are created using the Merkle tree that calculates the root hash to increase IoAT data security.

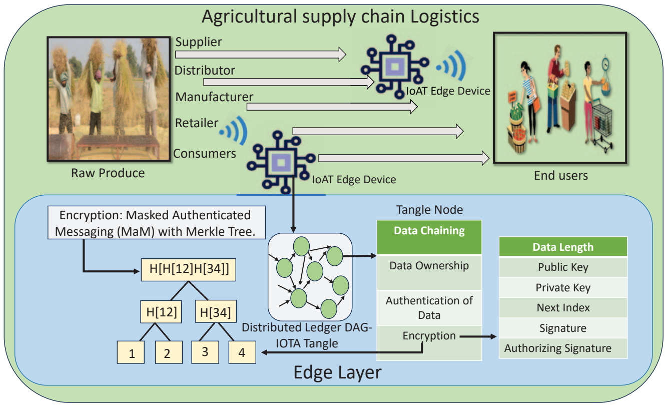


Fig. 3: Novel Architecture of agroString 2.0 through Distributed Ledger in supply chain.

A. Distributed Ledger-DAG (IOTA Tangle)

A distributed ledger comprises unique nodes that record, share, and synchronize the transactions in their own respective electronic ledgers rather than keeping the data in the centralized storage servers. Some of the examples of Distributed Ledger Systems include Blockchain, DAG, Holochain, Tempo, Hyperledger Fabric. Distributed Ledger, Directed Acyclic Graph(DAG) is called as Tangle where each point is a single transaction, and the arrows represent the authorizations. Individual data transactions are pointed toward the previous transactions named as parents and are responsible for approving the child data transaction. The approval of the transactions can be both direct and indirect. The starting transaction point is the genesis, and all the IOTA cryptocurrencies are designed one and all in the first genesis point, and no other new ones get created, and so on; all the other transactions have to approve the genesis transaction directly or indirectly [11]. A transaction gets accepted by a large number of newer transactions and gets added to a consensus to make it immutable. Every transaction goes through a small PoW-Power of work and achieves the consensus, making it impossible for the hacker to attack the Tangle. A Random Walk Monte Carlo algorithm is employed to reach a consensus with an authentic transaction. The Tangle grows and moves fastest with increasing weight to reach the consensus, and those with light weight are orphaned and are not included in the new consensus. Tangle is more advantageous than blockchain in avoiding high fees in micropayments for transactions, limiting the role separation among the individual participants, and evading double spending vulnerabilities [12].

B. Masked Authenticated Messaging (MaM)

A Masked authenticated (MaM) fulfills the industry's need wherever there is a requirement for data privacy and integrity inside Tangle. MaM data streaming in Tangle acts like another layer of communication protocol irrespective of the size and cost of the device once the consensus mechanism in IOTA performs a minor proof of work to flow the data through the network. A Merkle tree signature-based scheme is used in MaM to sign the encrypted message of the cipher digest. The channel ID is the root of the Merkle tree and is available for a short period; therefore, the data is attached to the root of the next Merkle tree to maintain secrecy in the channel. The visibility and access of the messages in Tangle can be controlled easily through MaM in multiple ways. In the public mode, the message that is being published directly contains the root of the Merkle tree as the transaction address. Any random user that comes across the message can retreive it by the address of the message. In the private mode, the transaction address is the hash of the Merkle root, and if any random user comes across the message, they can only be able to read but not able to decrypt or derive the hash of the root. In the restricted mode, the hash of the root is attached to an authorization key, and the address linked to the transaction message is the hash of the authorization key along with the Merkle root [13]. Whenever there is a change of key, the new key needs to be distributed to the relevant parties in order to retrieve the data or follow the order of the stream. One of the additional features of MaM is forward secrecy combined with channel splitting.

Each MaM publisher can split the channel at any time to give the future data with a new Merkle root which is not revealed earlier. This feature enables users with two types of visibility: public and restricted. Not knowing earlier Merkle roots, the user can only move the data stream forward [14].

C. Proposed algorithm for Current System

The data on temperature and humidity is collected from the edge devices residing between the supply chain logistics. The statistics from the IoAT sensing device are moved toward the Distributed Ledger tangle with Masked authenticated messaging (MaM) ability. The sensor device data attached to MaM is more secure and reliable through a permissionless distributed ledger system. Algorithm 1 explains the data flow from the sensor things to the IOTA tangle inside the edge device and elaborates on how the data gets validated and authenticated. The sensor data residing in Tangle is allocated to the following fields sensor data length, sensor data, public and private key, index number, nextindex number, signature, and authentication signature. An edwards25519 curve algorithm is used to generate a key pair for the input sensor data and creates a seed ($\mathrm{S}_{\mathrm{seed}}$) from the random source. The index number and nextindex numbers are calculated from the input sensor data with the help of public-private keys. The index number value is the hash of the public key, and the nextindex number value is the hash of the subsequent input sensor data. The same approach is followed for all the input data for creating seeds and calculating indexes. The algorithm BLAKE2b is applied for computing hashes in IOTA. The most critical step is to add indexes for continuous data streaming, authentication, and valid verification. The hash of the sensor data, sensor data length, public key, and nextindex are combined to give a digest D and sign the digest along with the private key to provide a sign field for verification of the data. The user can verify the input data by comparing both the hash and sign fields with the public key of the IoAT data, and if both are the same, the data is verified precisely. For authentication, the sign field is not helpful, but the auth-signature field. It is calculated using the IoAT device's key pair and the public key certificate. A third-party certificate authority helps in authenticating and validating the sensor data. The input data from the IoAT device is represented as $\mathrm{IoAT}_{\mathrm{input}}$, length of the input data as $\mathrm{IoAT}_{\mathrm{input\text{-}len}}$, the private key as $\mathrm{IoAT}_{\mathrm{input\text{-}Pr}}$, public key as $\mathrm{IoAT}_{\mathrm{input\text{-}Pu}}$, index value as $\mathrm{I}_{\mathrm{value}}$ and nextindex value as $\mathrm{NI}_{\mathrm{value}}$, a signature field with Si and the last authorized signature field as $\mathrm{Si}_{\mathrm{auth}}$. We denote H and d_{t} for Hashing and digest, respectively. Random source for generating the seed as $\mathrm{S}_{\mathrm{random}}$.

Algorithm 1 IoAT Sensor Data File in IOTA Tangle.

```
1:  $\mathrm{IoAT}_{\mathrm{input}} \rightarrow \mathrm{IoAT}_{\mathrm{input}}, \mathrm{IoAT}_{\mathrm{input\text{-}len}}, \mathrm{IoAT}_{\mathrm{input\text{-}Pr}}, \mathrm{IoAT}_{\mathrm{input\text{-}Pu}}, \mathrm{I}_{\mathrm{value}}, \mathrm{NI}_{\mathrm{value}}, \mathrm{Si}, \mathrm{Si}_{\mathrm{auth}}$.
2:  $S_{\mathrm{random}} \rightarrow S_{\mathrm{seed}}$.
3:  $S_{\mathrm{seed}} \rightarrow IoAT_{\mathrm{input\text{-}Pr}}, IoAT_{\mathrm{input\text{-}Pu}}$.
4:  $H(IoAT_{\mathrm{input\text{-}Pu}}) \rightarrow I$.
5:  Another key pair is generated for the next input data ($\mathrm{IoAT}_{\mathrm{nextinput}}$) from another random source ($\mathrm{S}_{\mathrm{random}}$).
6:  The key pairs from the next input data would be ($\mathrm{IoAT}_{\mathrm{nextinput\text{-}Pr}}$) and ($\mathrm{IoAT}_{\mathrm{nextinput\text{-}Pu}}$).
7:  $H(IoAT_{\mathrm{nextinput\text{-}Pu}}) \rightarrow NI_{\mathrm{value}}$.
8:  A digest D is calculated for signature.
9:  $D = H((IoAT_{\mathrm{input}}) + (IoAT_{\mathrm{input\text{-}len}}) + (IoAT_{\mathrm{input\text{-}Pu}}) + (NI_{\mathrm{value}}))$.
10: $Si = signature(D + IoAT_{\mathrm{input\text{-}Pr}})$
11: if H($\mathrm{IoAT}_{\mathrm{input}}$)==Si+$\mathrm{IoAT}_{\mathrm{input\text{-}Pu}}$ then
12:     successful verification.
13: else
14:     end the process.
15:     $\mathrm{Si}_{\mathrm{auth}}$ = signature($\mathrm{IoAT}_{\mathrm{Prkey}}$)
16:     if $\mathrm{Si}_{\mathrm{auth}}$ == signature($\mathrm{IoAT}_{\mathrm{Pukey}}$) then
17:         successful authentication.
18:     else
19:         end the process.
20:     end if
21: end if
22: Repeat the steps from 1 through 21 in each logistic step of the supply chain
```

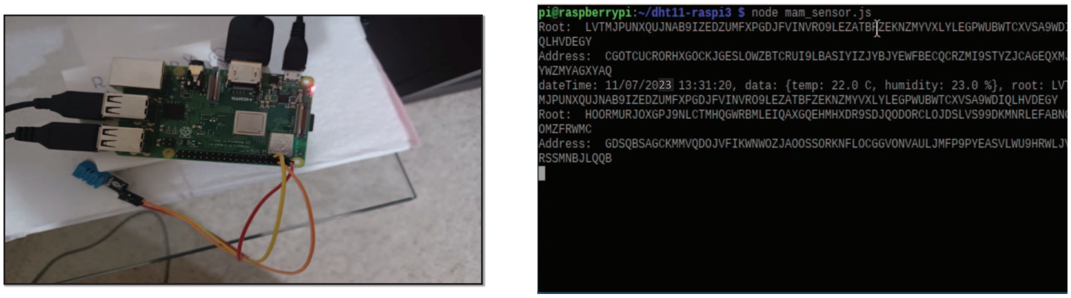
V. Experimental Results

A. Implementation

For implementing the application, we used Raspberry Pi as the IoAT edge device, as given in Fig. 4(a). We developed the application inside the Edge machine for publishing and retrieving the data to and from the Tangle. A node is installed on edge from the NodeJS.org provider for all the required modules for the execution of the application. A DHT11 sensor is connected to the raspberry pi edge device to record the temperature and humidity reading. The data from the sensor is sent in real-time to the agroString 2.0 application to publish the data onto the IOTA Tangle node and retrieve the same data with integrity and authenticity using MaM data streams. The Tangle application inside the edge is developed with javascript language. The project is developed with four main files that are executed in Raspberry Pi.

B. Experiments

Fig. 4(b) displays the readings of a sensor DHT11 temperature and humidity data through sensor.js file. The second javascript mam-sensor.js file is used to read and publish the sensor data to the Tangle using MaM. To publish random numbers to the Tangle from the sensor using MaM, a mam-publish.js file, and to extract the stored data from the Tangle using MaM and display the data, a mam-receive.js file is used. The sensor data is published to the Tangle using MaM, as shown in Fig. 5(a), and that root hash is saved and given as input in mam-recieve.js to retrieve the sensor data using MaM. The data retrieval is shown in Fig. 5(c). The application executes successfully. The node modules of IOTA and http client packages are called to run the code, and we show the results for publishing and receiving DHT11 sensor data in Fig. 5(b) and Fig. 5(d), respectively.

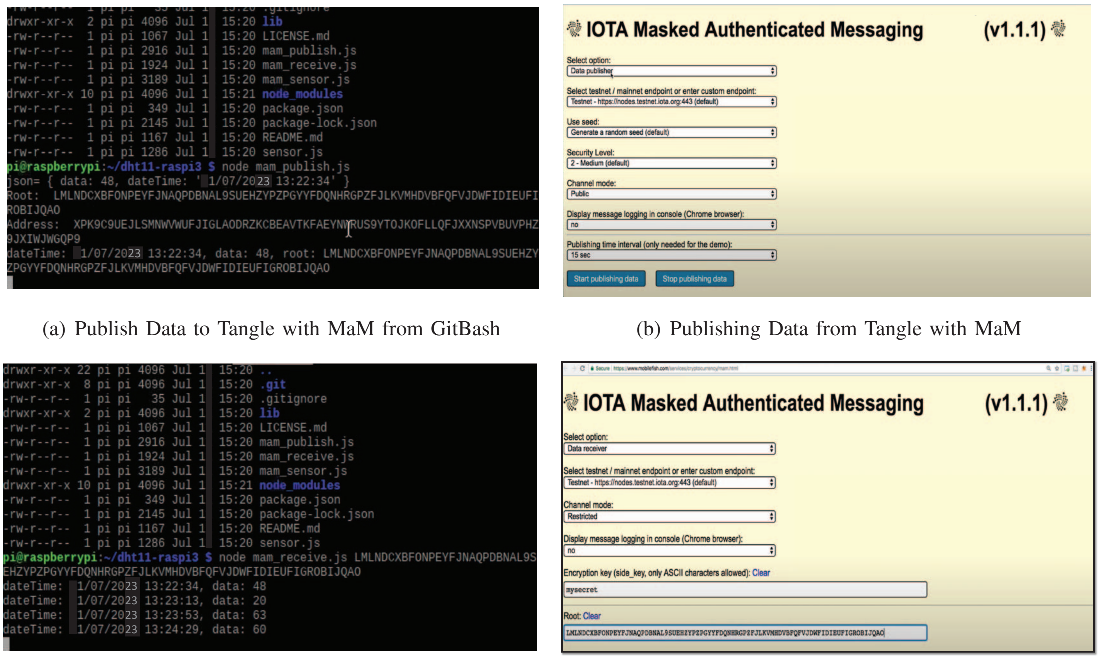


```
pi@raspberrypi:~/dht11-raspi3 $ node mam_sensor.js
Root:  LVTMJPUNXQUJNAB9IZEDZUMFXPGDJFVINVRO9LEZATBFZEKNZMYVXLYLEGPWUBWTCXVSA9WDI
QLHVDEGY
Address:  CGOTCUCRORHXGOCKJGESLOWZBTCRUI9LBASIYIZJYBJYEWFBECQCRZMI9STYZJCAGEQXMJ
YWZMYAGXYAQ
dateTime: 11/07/2023 13:31:20, data: {temp: 22.0 C, humidity: 23.0 %}, root: LV
MJPUNXQUJNAB9IZEDZUMFXPGDJFVINVRO9LEZATBFZEKNZMYVXLYLEGPWUBWTCXVSA9WDIQLHVDEGY
Root:  HOORMURJOXGPJ9NLCTMHQGWRBMLEIQAXGQEHMHXDR9SDJQODORCLOJDSLVS99DKMNRLEFABN
OMZFRWMC
Address:  GDSQBSAGCKMMVQDOJVFIKWNWOZJAOOSSORKNFLOCGGVONVAULJMFP9PYEASVLWU9HRWLJ
RSSMNBJLQQB
```

(a) IoAT Edge Device

(b) Sensor Data from Edge Device

Fig. 4: Collecting Edge Device Sensor Data.



```
-rw-r--r--  1 pi pi   35 Jul 1  15:20 .gitignore
drwxr-xr-x  2 pi pi 4096 Jul 1  15:20 lib
-rw-r--r--  1 pi pi 1067 Jul 1  15:20 LICENSE.md
-rw-r--r--  1 pi pi 2916 Jul 1  15:20 mam_publish.js
-rw-r--r--  1 pi pi 1924 Jul 1  15:20 mam_receive.js
-rw-r--r--  1 pi pi 3189 Jul 1  15:20 mam_sensor.js
drwxr-xr-x 10 pi pi 4096 Jul 1  15:21 node_modules
-rw-r--r--  1 pi pi  349 Jul 1  15:20 package.json
-rw-r--r--  1 pi pi 2145 Jul 1  15:20 package-lock.json
-rw-r--r--  1 pi pi 1167 Jul 1  15:20 README.md
-rw-r--r--  1 pi pi 1286 Jul 1  15:20 sensor.js
pi@raspberrypi:~/dht11-raspi3 $ node mam_publish.js
json= { data: 48, dateTime: ' 1/07/2023 13:22:34' }
Root:  LMLNDCXBFONPEYFJNAQPDBNAL9SUEHZYPZPGYYFDQNHRGPZFJLKVMHDVBFQFVJDWFIDIEUFI
ROBIJQAO
Address:  XPK9C9UEJLSMNWVWUFJIGLAODRZKCBEAVTKFAEYNWRUS9YTOJKOFLLQFJXXNSPVBUVPHZ
9JXIWJWGQP9
dateTime:  1/07/2023 13:22:34, data: 48, root: LMLNDCXBFONPEYFJNAQPDBNAL9SUEHZY
ZPGYYFDQNHRGPZFJLKVMHDVBFQFVJDWFIDIEUFIGROBIJQAO
```

IOTA Masked Authenticated Messaging (v1.1.1)

Select option: Data publisher

Select testnet / mainnet endpoint or enter custom endpoint: Testnet - https://nodes.testnet.iota.org:443 (default)

Use seed: Generate a random seed (default)

Security Level: 2 - Medium (default)

Channel mode: Public

Display message logging in console (Chrome browser): no

Publishing time interval (only needed for the demo): 15 sec

(a) Publish Data to Tangle with MaM from GitBash

(b) Publishing Data from Tangle with MaM

```
drwxr-xr-x 22 pi pi 4096 Jul 1  15:20 ..
drwxr-xr-x  8 pi pi 4096 Jul 1  15:20 .git
-rw-r--r--  1 pi pi   35 Jul 1  15:20 .gitignore
drwxr-xr-x  2 pi pi 4096 Jul 1  15:20 lib
-rw-r--r--  1 pi pi 1067 Jul 1  15:20 LICENSE.md
-rw-r--r--  1 pi pi 2916 Jul 1  15:20 mam_publish.js
-rw-r--r--  1 pi pi 1924 Jul 1  15:20 mam_receive.js
-rw-r--r--  1 pi pi 3189 Jul 1  15:20 mam_sensor.js
drwxr-xr-x 10 pi pi 4096 Jul 1  15:21 node_modules
-rw-r--r--  1 pi pi  349 Jul 1  15:20 package.json
-rw-r--r--  1 pi pi 2145 Jul 1  15:20 package-lock.json
-rw-r--r--  1 pi pi 1167 Jul 1  15:20 README.md
-rw-r--r--  1 pi pi 1286 Jul 1  15:20 sensor.js
pi@raspberrypi:~/dht11-raspi3 $ node mam_receive.js LMLNDCXBFONPEYFJNAQPDBNAL9S
EHZYPZPGYYFDQNHRGPZFJLKVMHDVBFQFVJDWFIDIEUFIGROBIJQAO
dateTime:  1/07/2023 13:22:34, data: 48
dateTime:  1/07/2023 13:23:13, data: 20
dateTime:  1/07/2023 13:23:53, data: 63
dateTime:  1/07/2023 13:24:29, data: 60
```

Secure https://www.mobilefish.com/services/cryptocurrency/mam.html

IOTA Masked Authenticated Messaging (v1.1.1)

Select option: Data receiver

Select testnet / mainnet endpoint or enter custom endpoint: Testnet - https://nodes.testnet.iota.org:443 (default)

Channel mode: Restricted

Display message logging in console (Chrome browser): no

Encryption key (side_key, only ASCII characters allowed): Clear

mysecret

Root: Clear

LMLNDCXBFONPEYFJNAQPDBNAL9SUEHZYPZPGYYFDQNHRGPZFJLKVMHDVBFQFVJDWFIDIEUFIGROBIJQAO

(c) Retrieve Data from Tangle with MaM from GitBash

(d) Extracting Data from Tangle with MaM

Fig. 5: Distributed Ledger Tangle with Masked authenticated Messaging.

C. Limitations and Challenges in current approach

Here in this current system, we send the data from the edge, collecting the real-time temperature and humidity statistics from the Raspberry Pi Device to the IOTA Tangle. The application is automated to read, store, and publish data. Approximately six hours of data have been collected to see the working ability of the application. The continuous data is about 279 kb in size, and we tested the loading and latency of the application. If the data size increases, there may be multiple issues in loading times and latency. The edge device used in the current paper is Raspberry Pi; leaving the power ’ON’ all the time leads to the rising temperature of the board. The main encounter was the need for knowledge in developing the current Tangle application and the lack of support for DApps with the Tangle. The Tangle network has no such support for these decentralized applications. The challenges faced were while programming javascript methods for the interconnections and transmissions of sensor data from the edge device Raspberry Pi DHT 11 to the back-end IOTA tangle and publishing and retrieving the data to and from the Tangle. Verifying the previous two transactions is enough to make a transaction valid in Tangle. Although Tangle has scalability and faster data loading times, in terms of Security, the blockchain is more reliable because a 51% attack is necessary for taking over a blockchain network, while only a 34% attack is enough for hacking the Tangle network.

D. Comparative Analysis

Table II compares the proposed work with other similar works in the current literature. Our proposed “agroString 2.0” doesn’t have any double-spending, and it has high data integrity and accuracy in authentication. The performance of

TABLE II: Comparison of previous IoAT data storage systems to current agroString 2.0.

Paper	Data Integrity	Authentication	Double-Spending	Energy Consumption	Cost
Zheng et al. [7]	No	Less	High	High	High
Mohammed and Chopra. [8]	Yes	2-factor Authentication	Less	Very High	Very High
Yang et al. [9]	Yes	Less	High	High	Less
Vangipuram et al. [10] (agroString)	Yes	High	No	High	Less
agroString 2.0 [Current-Paper]	**Yes**	**High**	**No**	**Very Less**	**Zero**

TABLE III: Performance Analysis of agroString 2.0 by evaluating data loading times and Latency.

Paper	Storage Technology	Edge	Time Taken	Latency
Zheng et al. [7]	Centralized	No	2.23s [15]	Very High
Mohammed and Chopra. [8]	Decentralized	No	8.72s [4]	Very High
Yang et al. [9]	Partially Centralized	No	10.2s [15], [4]	High
Vangipuram et al. [10] (agroString)	Decentralized	Yes-IoAT Data	1 ms [10]	Less
agroString 2.0 [Current-Paper]	**Distributed Ledger-Tangle**	**Yes-IoAT Data**	**Zero**	**Less**
Performance Calculated for Data Size of 279 Kb File.				

the current application is measured in terms of time taken to load data, cost, and latency with other works and evaluated with the present application as given in Table III.

VI. Conclusion and Future Direction

In this paper, we successfully design an application with the IoAT edge device raspberry pi to sense the data from the DHT11 sensor taking temperature and humidity and publish the data to the Distributed Ledger using a Masked authenticated Message service and successfully also retrieve the data securely. The application we design evades the limitations of the central and blockchain storage platforms and takes in real-time data from the sensor inside the edge layer. The application is successfully implemented, and the results are displayed to show the traceability and provenance of the food products inside the supply chain with the Distributed Ledger technology. The system increases the security of sensor data and provides integrity by providing quality food data to end consumers. In the future, the current system can be elaborated to other domains for the secure flow of sensitive information.

References

[1] U. Nations, “World population projected to reach 9.8 billion in 2050, and 11.2 billion in 2100,” *Department of Economic and Social Affairs*, 2017. [Online]. Available: https://www.un.org/development/desa/en/news/population/world-population-prospects-2017.html#:~:text=Worldpopulationprojectedtoreach9.8billionin,anewUnitedNationsreportbeinglaunchedtoday.

[2] NRDC, “Wasted: How America Is Losing up to 40 Percent of Its Food from Farm to Fork to Landfill,” https://www.nrdc.org/sites/default/files/wasted-2017-report.pdf, 2017, last Accessed on Jan 9th 2022.

[3] V. Udutalapally, S. P. Mohanty, V. Pallagani, and V. Khandelwal, “sCrop: A Novel Device for Sustainable Automatic Disease Prediction, Crop Selection, and Irrigation in Internet-of-Agro-Things for Smart Agriculture,” *IEEE Sensors Journal*, vol. 21, no. 16, pp. 17 525–17 538, August 2021.

[4] S. L. T. Vangipuram, S. P. Mohanty, and E. Kougianos, “CoviChain: A Blockchain Based Framework for Nonrepudiable Contact Tracing in Healthcare Cyber-Physical Systems During Pandemic Outbreaks.” *SN Computer Science*, 2021.

[5] A. K. Bapatla, S. P. Mohanty, and E. Kougianos, “sFarm: A Distributed Ledger Based Remote Crop Monitoring System for Smart Farming,” *SN Computer Science*, pp. 13–31, 2022.

[6] A. Alkhodair, S. P. Mohanty, and E. Kougianos, “FlexiChain 3.0: Distributed Ledger Technology-Based Intelligent Transportation for Vehicular Digital Asset Exchange in Smart Cities,” *Sensors*, vol. 23, no. 8, 2023. [Online]. Available: https://www.mdpi.com/1424-8220/23/8/4114

[7] M. Zheng, S. Zhang, Y. Zhang, and B. Hu, “Construct Food Safety Traceability System for People’s Health Under the Internet of Things and Big Data,” *IEEE Access*, vol. 9, pp. 70 571–70 583, 2021.

[8] C. P. Mohammed and S. R. Chopra, “Blockchain Security Implementation using Python with NB-IoT deployment in Food Supply Chain,” in *2023 International Conference on Emerging Smart Computing and Informatics (ESCI)*, 2023, pp. 1–5.

[9] X. Yang, M. Li, H. Yu, M. Wang, D. Xu, and C. Sun, “A Trusted Blockchain-Based Traceability System for Fruit and Vegetable Agricultural Products,” *IEEE Access*, vol. 9, pp. 36 282–36 293, 2021.

[10] S. L. T. Vangipuram, S. P. Mohanty, E. Kougianos, and C. Ray, “agroString: Visibility and Provenance through a Private Blockchain Platform for Agricultural Dispense towards Consumers,” *Sensors*, vol. 22, no. 21, 2022. [Online]. Available: https://www.mdpi.com/1424-8220/22/21/8227

[11] N. Sealey, A. Aijaz, and B. Holden, “IOTA Tangle 2.0: Toward a Scalable, Decentralized, Smart, and Autonomous IoT Ecosystem,” in *2022 International Conference on Smart Applications, Communications and Networking (SmartNets)*, 2022, pp. 01–08.

[12] N. Živi, E. Kadušić, and K. Kadušić, “Directed Acyclic Graph as Tangle: an IoT Alternative to Blockchains,” in *2019 27th Telecommunications Forum (TELFOR)*, 2019, pp. 1–3.

[13] P. Handy, “Introducing Masked Authenticated Messaging,” nov 2017. [Online]. Available: https://medium.com/iotatangle/introducing-masked-authenticated-messaging-e55c1822d50e

[14] M. Bhandary, M. Parmar, and D. Ambawade, “A Blockchain Solution based on Directed Acyclic Graph for IoT Data Security using IoTA Tangle,” in *2020 5th International Conference on Communication and Electronics Systems (ICCES)*, 2020, pp. 827–832.

[15] GCS, “How long does it take to upload gb data to the cloud?” 2014. [Online]. Available: https://www.goodcloudstorage.net/file-transfer-calculator/