Performance Evaluation and Optimization of MAM for Data Marketplace in IoT

Shubham Rahangdale
shubham00099@gmail.com

***Abstract*—This paper presents a secure and verifiable data storage architecture utilizing Masked Authenticated Messaging (MAM) within the IOTA Tangle, aimed at automating the trading of digital assets and services in an IoT ecosystem. MAM operations, such as channel and endpoint creation and data attachment, were evaluated on different hardware platforms, including a PC and Raspberry Pi 3. Results indicate that while MAM operations are computationally feasible on higher-powered machines, they become significantly time-consuming on lower-powered devices, such as the Raspberry Pi 3, especially as the Merkle Signature Scheme (MSS) height increases. The performance of MAM on low-level IoT devices poses challenges due to limited computational power and unstable internet connections. To address these issues, the paper proposes offloading MAM operations to brokers equipped with powerful machines and enhanced processing capabilities using Tangle-accelerator and Ethereum clients. This approach aims to lower the threshold for IoT device participation while ensuring data privacy and operational efficiency. The findings highlight the importance of improving MAM performance to make it a viable solution for data marketplaces in IoT environments.**

***Index Terms*—privacy computing, crowd sensing, publish/subscribe, decentralization**

I. Introduction

The rapid expansion of the Internet of Things (IoT) has triggered the emergence of a new digital economy centered around the value of data. As physical entities—ranging from transportation systems and industrial plants to residential buildings and agricultural fields—become equipped with sensors and interconnected devices, their digital representations, known as digital twins, are becoming crucial intermediaries for secure data exchange [1], [2]. This transformation enables real-time data flow across traditionally siloed infrastructures, reshaping linear value chains into dynamic, interconnected ecosystems. These evolving systems demand novel strategies from organizations, as traditional competitive paradigms become increasingly insufficient in the context of collaborative, data-driven environments.

One significant outcome of this shift is the rise of data marketplaces—platforms where data can be exchanged, monetized, and integrated into innovative business models. This ecosystem, which we term the "Economy of Things," enables machines and systems to transact autonomously, facilitating value creation from interconnected devices and data assets. However, the establishment of such marketplaces faces critical challenges:

1) Data owners often lack full control over their data, which is commonly confined within proprietary platforms.
2) Individual data holdings are typically of limited analytical value without access to external or aggregated datasets.
3) There is a significant knowledge gap among data owners regarding how to extract actionable insights from raw data streams.

To address these barriers, we present a lightweight and decentralized reference architecture for an IoT-enabled data marketplace. This system is optimized for deployment on constrained embedded devices and utilizes Tangle-based operations to ensure secure communication and trustless transactions. It leverages decentralized infrastructure to eliminate reliance on single points of failure, promoting robustness and scalability.

The proposed architecture identifies three primary roles within the marketplace:

1) **Data Sellers** are entities operating IoT infrastructures, such as smart energy meters or environmental sensors, and are interested in monetizing the data they generate.
2) **Managed Data Lakes** serve as repositories for large-scale data storage and metadata indexing, enabling efficient discovery and retrieval of datasets.
3) **Data Buyers** are stakeholders who seek to enhance their existing datasets with externally sourced data to derive new value or gain deeper insights.

A practical example of this model can be seen in Airbox [3], a distributed air quality monitoring initiative. Each household equipped with an Airbox device autonomously collects and shares environmental data without relying solely on centralized authorities. To maintain data privacy, sensitive information is encrypted before being stored on the blockchain. Additionally, the underlying system incorporates a smart-contract-based architecture that records each transaction. Data items marked as "tradeable" trigger verifiable requests, allowing buyers to review metadata and negotiate pricing before completing a transaction. Payments are held in escrow via smart contracts until the exchange is successfully verified. An overview of this process is illustrated in Fig. 1.

Another compelling use case involves smart electricity meters. These devices collect high-frequency energy consumption data, which holds significant potential for driving energy efficiency and enabling the transition to more sustainable energy systems. However, such data is inherently privacy-sensitive. Research has demonstrated that high-resolution consumption data—collected, for example, every 0.5 seconds—can reveal

detailed information about household behavior, including the types of appliances in use or even the specific television channel being watched [4]. By analyzing this granular data, it is possible to infer:

1) The number of residents in a household, based on patterns in water heater usage;
2) Daily routines or availability patterns by analyzing television usage cycles;
3) Specific lifestyle habits from the energy signatures of kitchen appliances like toasters or coffee machines;

As shown in Fig. 2, this level of granularity raises critical concerns regarding surveillance, profiling, and behavioral tracking. The emergence of such capabilities necessitates a fundamental rethinking of data privacy, ownership, and governance in digital infrastructures.

The European Union's General Data Protection Regulation (GDPR), enacted on May 25, 2018, addresses some of these issues by mandating that service providers:

1) Clearly specify the purpose of data collection and restrict use to that stated purpose;
2) Obtain explicit consent from consumers before accessing their personal data;
3) Cease data collection upon withdrawal of consent;

Therefore, any system that aims to enable open exchange of personal data—such as smart meter readings—must incorporate mechanisms that comply with these legal obligations. Our approach proposes two key measures: (1) Ensuring that energy consumption data is transmitted directly from the meter to the authorized service provider, and (2) Storing user consent on decentralized infrastructure. Avoiding centralized data or consent repositories mitigates risks such as irrevocable access due to server failures or vendor lock-in scenarios where service providers are predetermined by data platform owners.

The interconnected value networks emerging from these technologies will require unprecedented levels of cooperation among businesses. This will be made possible only through a common, decentralized infrastructure that supports economic transactions between machines. By providing a platform that is vendor-neutral and based on open standards, our proposed decentralized data marketplace fosters industrial automation, drives open innovation, and enables equitable participation in the digital economy.

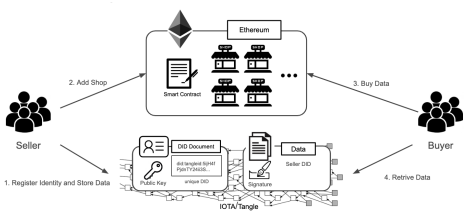

Fig. 1. IoT-enabled personal air quality assistant.

II. Related Work

The publish/subscribe service model consists of publishers, subscribers and brokers, which has been proven [5], [6] to be an efficient and flexible solution for a large number of diverse entities like IoT applications. A lot of work in publish/-subscribe system focused on the scalability and the different security issues such as, encrypted data communication, privacy preserving data subscription and access control of digital asset. A few work targets on the storage which is a vital considerations for IoT and mobile computing and the incentive for data economics.

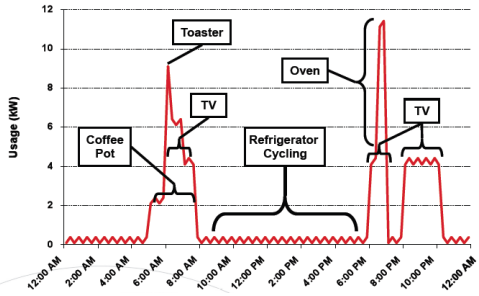

Fig. 2. Power consumption peaks can be associated with individual appliances. (Source: Privacy Issues Related to Smart Grid Technologies, Megan J. Hertzler)

M. B. Abdullahi and G. Wang [7] presented a secure publish/subscribe data storage service in Wireless sensor networks (WSNs) which ensures several security issues. Each user has an identity for authentication, whereas subscribers' interests are encoded before matching to protect users' interests. Additionally, the proposed encryption scheme can prevent adversary to access published data if the sensor node is compromised. However, the access control and encryption keys of data is enforced by the network controllers (NCs) and Cirtificate Authorities (CAs), which may be a potential security risk of the system. Also the storage is not well illustrated in the work. G. S. Ramachandran et al. [8] pointed out the security risk of centralized brokers, and applied DLTs to build a distributed pub/sub system which promotes the transparency of interactions of participants and the status of data. With the help of Smart Contract, usesrs can perform data validation easily, and brokers can keep track of data status. But data is plaintext on blockchain, which the privacy problem of sensitive data need to be considered carefully. The economics incentives that can encourage the publishers to participate the system and pay more attention on the quilatiy control are not included.

In [9], the publisher runs a node in blockchain that preserves all the history data of the ledger, therefore, they can publish and manage data without any third-parties. The subscribers request publisher directly and ask them to save a cache space for interested data. The main contribution of the system is to ensure data owners have full controls of produced data, but the data owners need to have well devices and environment to perform such functionalities and preserve data. Also, the rights for accessing digital assets is more compatible in the IoT scenarios instead of copying raw data. Secure Pub-Sub model [10], a brokerless of publish/subscribe model, is proposed to eliminate

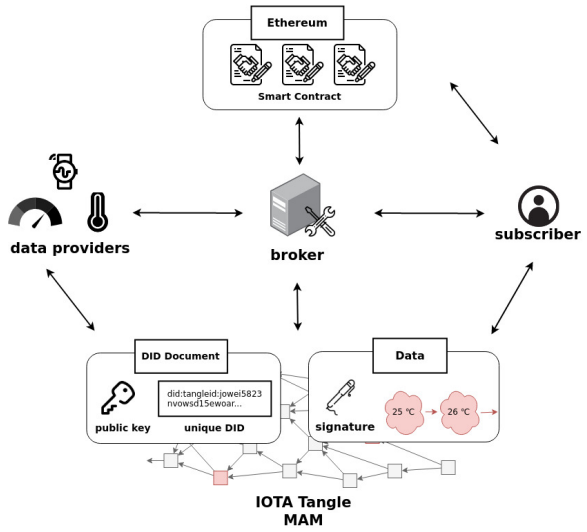

Fig. 3. The system design of a decentralized architecture which consists of data providers, subscribers and brokers.

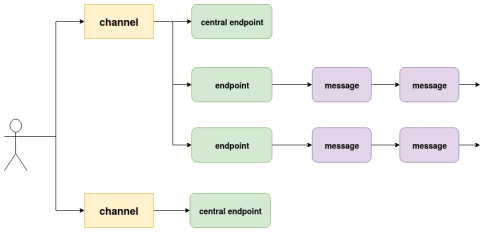

Fig. 4. The concept of MAM.

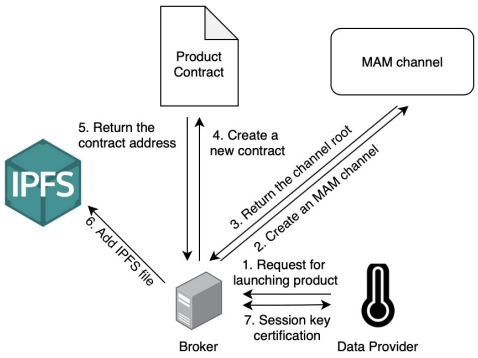

Fig. 5. The process of launching a product.

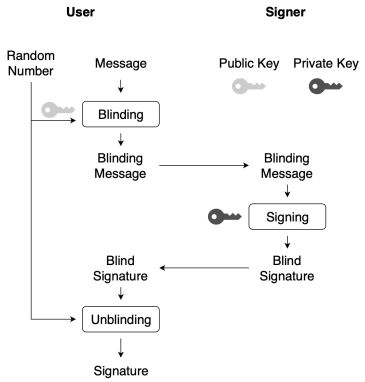

Fig. 6. The scheme of blind signature.

the security risk of middlewares in the model and to provide a reputation-based fairness payment strategy on blockcahin. The privacy and data security are considered thoroughly with the encryption scheme, while the reputation of publishers, payment and data sharing are deployed on smart contracts that allows all operations are transparent. The reputation system and the punishment rules against malicous acts of subscribers and publishers. Yet, without brokers, providers and subscribers may need to reveal more sensitive information like IP address in order to match the both sides. Another broker-less model in [11] protects the subscribers' privacy by encrypting users' interests with the light-weight PKEwET [12], which allows publishers to match the subscribers' interests in cipher text.

Decentralized storage systems allow users to store files in a distributed network that is maintained by individual nodes around the world instead of a central service provider. Nevertheless, DLTs are often used as the backbone of these systems as data storage and also an incentive layer to encourage people get involved in the network. Filecoin [13] in Inter-Planetary File system(IPFS) [14] is an incentive layer to incent nodes to provide storage. IPFS is a content-based addressing storage model in a peer-to-peer network, which users can obtain the data with the unique hash value through the network. However, no cryptographic system is applied for user-uploaded files. Sia [15] splits the uploaded file into multiple data segments encrypted with the owner's private key, then cipher text is sent to the Sia nodes that rent the storage in Siacoin through smart contracts. Files are duplicated in multiple nodes to prevent data loss.

III. System Architecture

We propose a decentralized, three-tier data marketplace architecture comprising data providers, subscribers, and brokers. Providers publish data, subscribers initiate purchases, and brokers mediate transactions, manage publishing, and handle metadata generation.

A. Participants

Figure 3 outlines three key roles:

- **Data Providers:** Entities that generate continuous data streams and offer them for purchase to improve their services.
- **Subscribers:** Consumers who purchase data streams to enhance their operations, avoiding the cost of independent data collection.
- **Brokers:** Trusted intermediaries with higher computational resources. They manage trade execution and data publishing on behalf of providers and subscribers and collect brokerage fees.

B. Components

Our framework integrates multiple modules to support secure, verifiable, and transparent data exchange:

1) Masked Authenticated Messaging (MAM): Built atop IOTA's Tangle, MAM enables publishing and retrieval of encrypted data streams as zero-value transactions. Instead of data copies, access rights are traded, allowing on-demand, location-independent data access. Using an IOTA seed, providers generate channels and endpoints that form singly linked lists to ensure forward secrecy. Session keys encrypt messages and control access. Providers can preview products through public central endpoints. Brokers, responsible for MAM operations, create and manage channels and handle data publishing. Figure 4 shows the MAM setup.

Authentication in MAM uses the Merkle Signature Scheme (MSS), ensuring data origin and integrity. As MSS requires a fixed tree size, providers must predefine channel and endpoint lengths.

2) TangleID: TangleID enables decentralized, self-sovereign identity management using Decentralized Identifiers (DIDs) as per W3C standards. DID documents are published on MAM, enabling GDPR compliance and proving data origin. Participants use key pairs and IOTA seeds to create public/private MAM channels, supporting secure communications and digital signatures.

Trust is established via Verifiable Credentials issued by trusted authorities. These credentials are signed and shared through the Tangle to validate message authenticity and participant reputation.

3) Ethereum Smart Contracts: Smart contracts on Ethereum facilitate secure, programmable trade execution. The Product Contract stores metadata including MAM IDs, encrypted session keys, and provider identities. It also manages key exchange between providers and subscribers, minimizing broker involvement and enhancing trust.

4) Blind Signature: To protect session keys from brokers, blind signatures are used. This cryptographic approach ensures brokers cannot decipher the keys while still signing them. Using RSA blind signatures, a subscriber blinds the session key with a random factor and the broker signs it. The original key can be recovered and verified without revealing its content to the signer. Figure 6 demonstrates this process.

$$C = r^e m \tag{1}$$

$$S = C^d \tag{2}$$

$$\frac{S}{r} = m^d \tag{3}$$

This method ensures the broker signs a key they cannot access, protecting the confidentiality of traded data.

IV. Trading Model

In the following, we describe the data trading process in detail. To participate a data marketplace, data providers and subscribers have to first register. Then, the data provider can launch its product on the marketplace. Once a product is launched, it is searchable on IPFS where brokers upload the information of the product and can be subsequently traded. The whole trading and refunding process is defined in smart contracts which are easily traceable and irreversible.

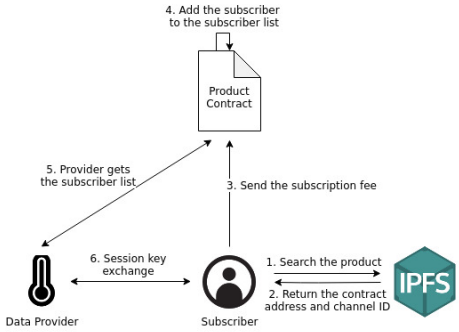

Fig. 7. The process for the product trading.

A. Set up

At the beginning, all participant need to register on TangleID in order to get a DID document and public/private key pairs. The DID document is shared within the following operations to perform authentication, key exchangement and secure communication.

B. Trading

The entire trading process is as shown in Fig. 7. Once a subscriber wants to subscribe certain streaming data that is generated, he/she has to pay subscription fee to the Product Contract and will be automatically added to the subscriber list by the smart contract.

In the proposed architecture, the "access" to specific data stream is traded instead of giving out the raw data copies. Subscribers can retrieve data on MAM at any time without equipping extra storage to preserve data, and one can also query for arbitrary section of data rather downloading them all. Thus, the most important part of trading process is to give the session key k, the encryption key of data stream, to the subscribers. In order to exchange data, Azaria et al [16] seek an off-chain solution which is based on an end-to-end communication to ensure the efficiency and low costs. However, it is hard to avoid that the data source is unavailable or the sender sends incorrect data intentionally. Instead of the off-chain solution, we use smart contract to transfer the session key [17] that not only ensures the consistency of the session key, the availability of the source and the traceability of the record, but also prevents malicious participants to cheat others. With the help of brokers and smart contracts, both providers and subscribers do not need to be online at the same time to proceed the trading process.

```
function addEncryptKey(
  address _subscriber,
  string memory _encryptKey
) public {
  require(
      msg.sender == provider,
```

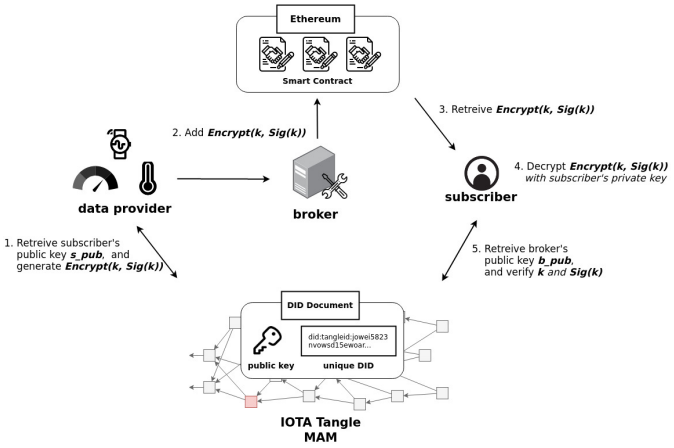

Fig. 8. Session key exchange process between the data provider and subscriber.

```
        "Only provider can add blindedKey."
    );
    require(
        subscriber2Purchase[_subscriber].isKeyAdded == false,
        "One encryptKey has been added."
    );
    subscriber2Purchase[_subscriber].encryptKey = _encryptKey;
    subscriber2Purchase[_subscriber].isKeyAdded = true;
    emit newEncryptKey(_subscriber, _encryptKey);
}
```

Listing 1. Update encryption key

The trading process is shown in Fig. 8. The data provider can obtain public keys of each subscriber from the DID document. For each subscriber, the data provider encrypts the session key and broker's signature with the subscriber's public key and sends the ciphertext $Encrypt(k + Sign(k))$ to the product contract by calling $addEncryptKey()$ (Listing 1). Subscribers listen to the $newEncryptKey$ event which is triggered when the ciphertext is updated, and decrypt the ciphertext to obtain the session key k and signature $Sign(k)$.

Subscribers can obtain the broker's public key on the DID document as well, so they can verify that the validation of signature and the session key is the only one certified by the broker. Afterward, with the MAM channel ID and session key k, subscribers can obtain the data.

C. Refunding

To ensure fairness, subscription fees are held until data is published to the MAM channel. If data is delayed or incomplete, subscribers can vote for a refund. Once votes exceed a threshold, the smart contract redistributes fees proportionally among the provider, broker, and subscribers.

The refund allocations are computed as:

$$F_{\text{DataProvider}}(i) = N \cdot \text{price} \cdot \frac{i-1}{M} \cdot (1 - F_b) - F_t \quad (4)$$

$$F_{\text{Broker}}(i) = N \cdot \text{price} \cdot \frac{i-1}{M} \cdot F_b - F_t \quad (5)$$

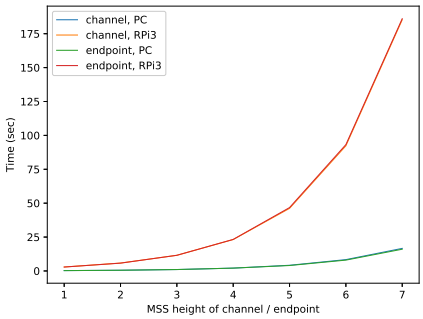

Fig. 9. Time cost of MAM creation.

$$F_{\text{Subscriber}}(i) = \text{price} \cdot \frac{M - i + 1}{M} - F_t \quad (6)$$

Where i is the number of delivered data points, M is the expected total, F_b is the broker fee, F_t is the transaction cost, N is the number of subscribers, and price is the subscription cost.

This approach ensures providers are paid only for delivered data, while subscribers are partially refunded for undelivered content.

TABLE I
TIME MEASUREMENT OF CHANNEL CREATION

height of MSS	PC (sec)	Raspberry Pi 3 (sec)
1	0.26183	2.908702
2	0.524076	5.805524
3	1.045942	11.555660
4	2.092989	23.178036
5	4.19515	46.164079
6	8.361586	92.320173
7	16.651607	185.292243

TABLE II
TIME MEASUREMENT OF ENDPOINT CREATION

height of MSS	PC (sec)	Raspberry Pi 3 (sec)
1	0.256425	2.887064
2	0.505679	5.767912
3	0.999524	11.550455
4	1.994017	23.260508
5	3.965007	46.748366
6	7.918925	93.182975
7	16.561419	186.064562

V. EVALUATIONS

In this framework, participants are not required to operate a full IOTA node. Instead, they interact with the Tangle through lightweight client libraries, which significantly reduces the system's hardware requirements. All evaluations in this section assume such a lightweight setup.

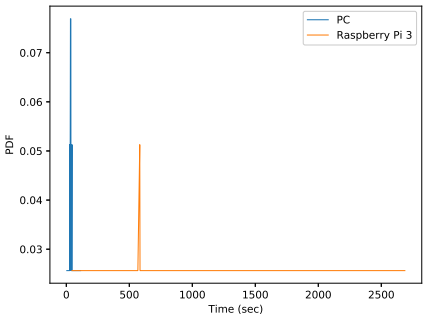

Fig. 10. Time cost of sending a message through MAM.

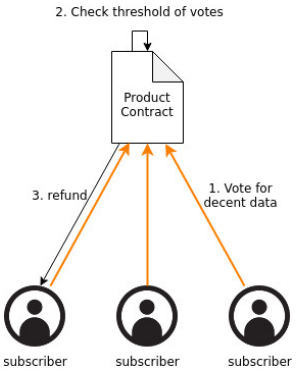

Fig. 11. The process of refund.

A. MAM Performance Evaluation

Masked Authenticated Messaging (MAM) plays a vital role in secure and verifiable data publishing. Since data providers may frequently upload data or manage multiple channels simultaneously, MAM operations are potential bottlenecks in the system.

We assess two key MAM operations: channel/endpoint creation and message publication. Experiments were performed on a desktop PC (Intel i7-8700, 16GB RAM) and a Raspberry Pi 3B (ARM Cortex-A53, 1GB RAM).

1) Channel and Endpoint Creation: Channel and endpoint creation depends on the height of the Merkle Signature Scheme (MSS). Higher heights increase capacity but demand more processing. For our tests, MSS heights ranged from 1 to 7, with average creation times recorded over 100 iterations (see Tables I and II).

As shown in Fig. 9, both channel and endpoint creation on a PC are efficient across all heights. On the Raspberry Pi, performance declines significantly when the height exceeds 4, suggesting limited suitability for resource-constrained devices. This supports the case for offloading such tasks to brokers.

2) Message Publication: Publishing messages to MAM involves:

- **Tip Selection:** Choosing two unconfirmed transactions (tips) to approve.

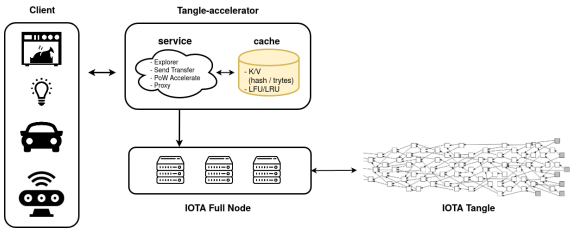

Fig. 12. The structure of Tangle-accelerator.

- **Proof-of-Work (PoW):** Solving a lightweight cryptographic puzzle to protect against spam.

These steps require both network stability and processing power. Fig. 10 displays the timing distribution for message publishing. While the PC shows stable results, the Raspberry Pi's performance is highly variable due to PoW complexity.

These results indicate that low-power IoT devices struggle with MAM operations due to limited computational and network resources. Delegating these tasks to brokers—powerful machines capable of running Ethereum clients and Tangle-accelerators—can significantly improve performance.

Tangle-accelerator, a local caching proxy for IOTA, enables fast MAM processing by reducing access latency to remote nodes and accelerating PoW (see Fig. 12). However, further optimization of MAM performance remains a critical area for future work.

VI. Conclusions

This paper presented a comprehensive framework for a decentralized publish/subscribe model tailored for secure and autonomous data and service trading in the Internet of Things (IoT) ecosystem. By leveraging a combination of well-established standards and open specifications, the proposed architecture was designed to be vendor-neutral and industry-agnostic, enabling seamless integration across diverse platforms and devices.

At the core of the system is a hybrid architecture that integrates blockchain technology with the IOTA Tangle, facilitating immutable audit trails, transparent contract execution, and decentralized identity management. This integration ensures that all participants—whether data providers, subscribers, or brokers—can authenticate each other through verifiable credentials while maintaining privacy and security. The use of decentralized identities (DIDs) further strengthens the trust framework, eliminating reliance on centralized authorities and minimizing identity-related vulnerabilities.

The system employs Masked Authenticated Messaging (MAM) for secure and validated data streaming. While MAM ensures message confidentiality and integrity, the performance evaluation indicated that low-power devices such as Raspberry Pi struggle with the computational overhead of MAM operations, especially during endpoint creation and Proof-of-Work (PoW) tasks. To mitigate this, we introduced the concept of offloading MAM operations to more powerful brokers.

These brokers, equipped with Ethereum clients and Tangle-accelerator services, enable efficient processing and transaction throughput, thereby lowering the barrier for resource-constrained devices to participate in the network.

Furthermore, the proposed model supports an autonomous trading mechanism using smart contracts, allowing data and services to be exchanged programmatically with minimal human intervention. This automation ensures scalability, reduces transaction latency, and fosters a real-time marketplace where participants can dynamically publish or subscribe to digital assets based on agreed terms.

In addition to its technological innovation, the framework promotes data ownership, transparency, and interoperability—key values in emerging data economies. The combination of blockchain-based smart contracts, IOTA-based scalable communication, and decentralized identity verification lays the foundation for a robust and trustworthy marketplace.

However, despite these advantages, several challenges remain. Performance bottlenecks in MAM processing, particularly on edge devices, highlight the need for further optimization or lightweight alternatives. Similarly, privacy-preserving mechanisms such as blind signatures, although effective, warrant deeper exploration in terms of scalability and usability in dynamic environments.

In conclusion, this paper demonstrates the feasibility and benefits of a decentralized, secure, and automated publish/-subscribe system for digital asset exchange. The proposed architecture not only ensures data authenticity and user privacy but also promotes a low-entry barrier, making it highly suitable for real-world IoT deployments. Future work will focus on refining MAM efficiency, integrating advanced cryptographic schemes, and testing the model in larger-scale deployments to assess its real-world viability and impact.

REFERENCES

[1] K. Borodulin, G. Radchenko, A. Shestakov, L. Sokolinsky, A. Tchernykh, and R. Prodan, "Towards digital twins cloud platform: Microservices and computational workflows to rule a smart factory," in *Proceedings of the10th International Conference on Utility and Cloud Computing*, ser. UCC '17. New York, NY, USA: ACM, 2017, pp. 209–210.

[2] S.-H. Wang, S.-W. Cheng, and C.-C. J. Huang, "Puyuma: Linux-based rtos experimental platform for constructing self-driving miniature vehicles," in *Intelligent Computing*. Springer International Publishing, 2019, pp. 985–994.

[3] S. Mahajan, W.-L. Wu, T.-C. Tsai, and L.-J. Chen, "Design and implementation of iot-enabled personal air quality assistant on instant messenger," in *Proceedings of the 10th International Conference on Management of Digital EcoSystems*, ser. MEDES '18. New York, NY, USA: ACM, 2018, pp. 165–170.

[4] M. J. Hertzler. Privacy issues related to smart grid technologies. [Online]. Available: https://www.splunk.com/en_us/blog/security/smart-grid-data-the-wild-west-of-privacy-rights.html

[5] P. Eugster, P. Felber, R. Guerraoui, and A.-M. Kermarrec, "The many faces of publish/subscribe," *ACM Comput. Surv.*, vol. 35, pp. 114–131, 06 2003.

[6] C. Esposito, "A tutorial on reliability in publish/subscribe services," *Proceedings of the 6th ACM International Conference on Distributed Event-Based Systems, DEBS'12*, 07 2012.

[7] M. B. Abdullahi and G. Wang, "Secure publish-subscribe-based in-network data storage service in wireless sensor networks," in *2012 IEEE 8th International Conference on Distributed Computing in Sensor Systems*, May 2012, pp. 297–299.

[8] G. S. Ramachandran, K. Wright, L. Zheng, P. Navaney, M. Naveed, B. Krishnamachari, and J. Dhaliwal, "Trinity: A byzantine fault-tolerant distributed publish-subscribe system with immutable blockchain-based persistence," in *2019 IEEE International Conference on Blockchain and Cryptocurrency (ICBC)*, May 2019, pp. 227–235.

[9] S. Hashemi, F. Faghri, and R. Campbell, "Decentralized user-centric access control using pubsub over blockchain," 09 2017.

[10] Y. Zhao, Y. Li, Q. Mu, B. Yang, and Y. Yu, "Secure pub-sub: Blockchain-based fair payment with reputation for reliable cyber physical systems," *IEEE Access*, vol. 6, pp. 12 295–12 303, 2018.

[11] P. Lv, L. Wang, H. Zhu, W. Deng, and L. Gu, "An iot-oriented privacy-preserving publish/subscribe model over blockchains," *IEEE Access*, vol. PP, pp. 1–1, 03 2019.

[12] G. Yang, C. H. Tan, Q. Huang, and D. S. Wong, "Probabilistic public key encryption with equality test," in *Topics in Cryptology - CT-RSA 2010*, J. Pieprzyk, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 119–131.

[13] P. Labs. Filecoin: A decentralized storage network. [Online]. Available: https://filecoin.io/filecoin.pdf

[14] J. Benet, "Ipfs - content addressed, versioned, p2p file system," 07 2014.

[15] D. Vorick and L. Champine. Sia: Simple decentralized storage. [Online]. Available: https://sia.tech/sia.pdf

[16] A. Azaria, A. Ekblaw, T. Vieira, and A. Lippman, "Medrec: Using blockchain for medical data access and permission management," in *2016 2nd International Conference on Open and Big Data (OBD)*, Aug 2016, pp. 25–30.

[17] R. J. Pooja Gupta, Salil S. Kanhere, "A decentralized iot data marketplace," 2019.