future internet

MDPI

Article

Utilizing Blockchain for IoT Privacy through Enhanced ECIES with Secure Hash Function

Yurika Pant Khanal [1], Abeer Alsadoon [1], Khurram Shahzad [1,*], Ahmad B. Al-Khalil [2], Penatiyana W. C. Prasad [1], Sabih Ur Rehman [1] and Rafiqul Islam [1]

[1] School of Computing, Mathematics and Engineering, Charles Sturt University, Melbourne 3062, Australia; yurikapant@gmail.com (Y.P.K.); alsadoon.abeer@gmail.com (A.A.); cwithana@csu.edu.au (P.W.C.P.); sarehman@csu.edu.au (S.U.R.); mislam@csu.edu.au (R.I.)

[2] College of Science, Department of Computer Science, The University of Duhok, Duhok 42001, Iraq; ahmad.al-khalil@uod.ac

* Correspondence: kshahzad@csu.edu.au

Abstract: Blockchain technology has been widely advocated for security and privacy in IoT systems. However, a major impediment to its successful implementation is the lack of privacy protection regarding user access policy while accessing personal data in the IoT system. This work aims to preserve the privacy of user access policy by protecting the confidentiality and authenticity of the transmitted message while obtaining the necessary consents for data access. We consider a Modified Elliptic Curve Integrated Encryption Scheme (ECIES) to improve the security strength of the transmitted message. A secure hash function is used in conjunction with a key derivation function to modify the encryption procedure, which enhances the efficiency of the encryption and decryption by generating multiple secure keys through one master key. The proposed solution eliminates user-dependent variables by including transaction generation and verification in the calculation of computation time, resulting in increased system reliability. In comparison to previously established work, the security of the transmitted message is improved through a reduction of more than 12% in the correlation coefficient between the constructed request transaction and encrypted transaction, coupled with a decrease of up to 7% in computation time.

Keywords: Internet of Things; blockchain; ECIES; secure hash function; privacy; reliability

Citation: Khanal, Y.P.; Alsadoon, A.; Shahzad, K.; Al-Khalil, A.B.; Prasad, P.W.C.; Rehman, S.U.; Islam, R. Utilizing Blockchain for IoT Privacy through Enhanced ECIES with Secure Hash Function. *Future Internet* **2022**, *14*, 77. https://doi.org/10.3390/fi14030077

Academic Editors: Rattikorn Hewett and Paolo Bellavista

Received: 10 January 2022
Accepted: 24 February 2022
Published: 28 February 2022

1. Introduction

With the recent advances in technology, several Internet of Things (IoT) devices are being developed and implemented in our day to day life. These IoT devices collect personal data from the user to carry out different processes across several applications. Given the involvement of these devices in our daily life, the collected data are prone to a variety of security and privacy threats [1,2], in particular the monitoring of user's activities and profile creation [3]. Moreover, users do not have control over their data and necessary information regarding how it is being collected and how it is further processed. It thus becomes essential to protect the privacy rights of the users and facilitate them with the ability to control their transmitted data under the IoT landscape.

Data profiles can be utilised for individual identification purposes and therefore, collecting data and creating user data profiles pose a severe threat towards privacy and personal integrity. Even if the IoT data are not connected directly to an individual, it is possible to collect IoT data and create profiles of individuals. These profiles can be used to identify individuals or groups of individuals and pose a direct threat to user privacy. If data from IoT devices are combined with data from other sources such as social media, the identification of groups and/or individuals becomes much easier. One of the most critical parts of data collection via IoT devices is that most of the time, consumers are not aware of what data are being collected and how they are being used. Even in cases

where consumers agree to the collection of data for a specific application, it is difficult for them to perceive the number of ways that data may be used in the future. The work of [4] investigates the possibilities to recognise a user based on when they communicate, what kind of applications they use, the type of devices they are surrounded by and their geographical location.

Traditionally, a user's sensitive data are stored on centralized servers [5], which can be easily tampered by the third party resulting in additional security and privacy threats, since user data was accessible without obtaining consent from the user. To address this issue, Blockchain-based solutions have been proposed in the IoT system, where several approaches have been advocated to protect user privacy [6–10]. Blockchain technology has dramatically enhanced user privacy and data access owing to its decentralized nature, enabling all participating nodes in the Blockchain to provide services equally [11]. In case of a node failure, other nodes keep providing the service, removing single point of failure that is a major problem in the traditional methods. The immutability feature of blockchain technology protects the data from being tampered and safely store the data in the form of blocks [12]. These features of blockchain technology eliminate the limitations of traditional centralized servers used in IoT applications. However, they still suffer from issues such as privacy protection and behavior regulation of access policy. In order to trace the real identity in an unusual transaction and preserve the privacy of the user in the data access policy, it is necessary to protect authenticity and confidentiality of the transmitted message while obtaining the consent needed for data access in the IoT system.

Our focus in this work is on protecting the confidentiality and authenticity of user consents during data transmission in IoT systems. We aim to preserve user privacy by maintaining the integrity of user consents before data transmission takes place in the IoT network. To improve the security strength of the encryption and decryption keys of the request transaction and response, we propose a two-pronged approach. Firstly, we proposed the use of a Secure Hash Function (SHF) [13] to derive private and public keys and secondly, we recommend the use of Key Derivation Function (KDF) to derive multiple keys to prevent the attacker from detecting the actual key value. The improved security strength decreases the correlation coefficient between constructed request transactions and encrypted transactions, enhancing user privacy in IoT systems. The proposed solution also improves the reliability of the system compared to a recent work of Lin et al. [14] by eliminating user-dependent variables and reducing the computation time.

The rest of the paper is organized as follows: Section 2 discusses in detail the recent advances in blockchain security measures, with a focus on its application in the IoT landscape. We detail the proposed scheme in Section 3, providing the major steps and associated details. Section 4 discusses the benefits of the proposed scheme, providing comparison to related works. In Section 5, we present analysis and detailed results of our scheme, demonstrating the efficacy in terms of average correlation coefficient and computation time, whereas the interim results on different datasets are also provided. Finally, the paper is concluded in Section 6, provisioning some future research directions.

2. Related Works

Blockchains are tamper evident and tamper resistant digital ledgers implemented in a distributed fashion, usually without a central authority. At their basic level, they enable a community of users to record transactions in a shared ledger such that under normal operation of the blockchain network, no transaction can be changed once published [15]. Unlike traditional methods, blockchain enables peer-to-peer transfer of digital assets without any intermediaries. Blockchain is often regarded as a public ledger in which all committed transactions are stored in a chain of blocks, and this chain continuously grows when new blocks are appended to it. The blockchain technology's key characteristics include decentralisation, persistency, anonymity and auditability.

2.1. Ethereum Public Blockchain

Ethereum represents a blockchain providing an abstract layer that enables all users to create their own rules for ownership, formats of transactions, and state transition functions, which is achieved through the use of smart contracts [16]. The consensus in the Ethereum network is based on modified GHOST protocol. Ethereum is created to tackle the issue of stale blocks in the network since the GHOST protocol includes stale blocks into calculations of the longest chain. The authors in [17] enhanced user privacy in a mobile crowdsensing system with spatial location privacy-preserving and greedy algorithms to improve data quality and preserve location privacy. They constructed a blockchain-based location privacy-preserving mobile crowdsensing system where the decentralization and immutability of Blockchain avoids security issues. However, the algorithm used in this scheme is based on the estimated value, so any inaccurate estimate may lead to significant problems and does not ensure data quality and reliability of the system.

In [18], the focus was on enhancing a smart healthcare system using a blockchain to preserve the privacy of the health data and ensure that diagnoses are not tempered. The proposed solution decreases the computation and communication cost comparing to the traditional system when preserving privacy in smart healthcare. However, the computation time is not fixed as the scheme requires users to update their key each time the transaction is updated. The researchers in [19] designed and implemented a decentralized reputation system to develop trust in the public fog nodes for enabling the IoT devices to rely on them securely. It provides safety against security vulnerabilities associated with IoT data and maintains the integrity of the data. The method uses the opinions of multiple users regarding the performance of public fog nodes to calculate reputation score for the future user to uses this system, which shows the unreliability of the system performance since the change in users' opinions changes the reputation score and increases the computation cost. Several computing task offloading schemes in mobile edge computing for IoT devices have been developed in [20]. The developed system uses a Blockchain-enabled edge computing framework and non-dominated sorting genetic algorithm to maintain data integrity while performing a task offloading process. Moreover, it adopts simple additive weighting and multi-criteria decision making techniques to select the most suitable offloading schemes. The task offloading system consumes 5% less energy than compared methods and decreases offloading time and energy consumption with data integrity and privacy protection. However, this work does not consider the security of VM instances while moving from one edge computing device to another device for obtaining load balance.

2.2. Consortium Blockchain

Consortium blockchain is a type of blockchain with authorized nodes to maintain distributed shared databases. Constructed by several organizations, the consortium blockchain is partially decentralized as only a small portion of nodes would be selected to determine the consensus. Among other advantages, recent works [21] have shown that it offers high potential for the establishment of decentralized electricity trading system with moderate cost. The authors of [22] propose a blockchain-based secure and privacy-preserving personal health information sharing scheme for diagnosis improvements in e-Health systems, where private and consortium blockchain are constructed by devising their data structures, and consensus mechanisms. In order to achieve data security, access control, privacy preservation and secure search in this work, all the data including the health information, keywords and the patients' identities are public key encrypted with keyword search. In [23], the authors construct a consortium blockchain framework for detecting malicious codes in malware and extracting the corresponding evidences in mobile devices. The work performs feature modelling by utilizing statistical analysis method, where the framework is composed of a detecting consortium chain shared by test members and a public chain shared by users. The authors also design a multi-feature detection method of

Android-based system for detecting and classifying malware, and establish a fact-base of distributed Android malicious codes by blockchain technology.

2.3. Hyperledger Fabric Blockchain

Hyperledger Fabric is an implementation of a distributed ledger platform for running smart contracts, leveraging familiar and proven technologies, with a modular architecture allowing pluggable implementations of various functions [24]. Designed as an extensible general-purpose permissioned blockchain, Hyperledger Fabric is the first blockchain system that supports the implementation of distributed applications written in standard programming languages [25]. This essentially allows them to be executed consistently across many nodes, giving impression of execution on a single globally-distributed blockchain computer, making Fabric the first distributed operating system for permissioned blockchains. The authors of [26] showed that the security can be enhanced by using proof of block and trade consensus algorithms to validate trade and blocks before allocating them to the ledger. Their solution uses a lightweight consensus algorithm, resulting in reduced computation time. However, it is resource intensive as it requires each trade to be validated before and at the time of block formation.

In [27], the authors proposed to improve privacy in industrial IoT with a Blockchain-based secure data sharing model for distributed multiple parties. They used federated learning algorithms to transform raw data generated in industrial IoT into the corresponding data model and share it. This model helps prevent data leakage, and data owners can assess before giving access to share their data in Industrial IoT. It provides high efficiency and enhanced security over traditional solutions. However, stable accuracy is difficult to achieve with the increase in the number of data providers. Also, an increase in the number of data providers requires a system to scale data for performing the computation. The consensus protocol is enhanced in [28] by checking the data loss before the data transmission to the blockchain network. This system uses a gossip-based diffusion function that guarantees the data collected from the sensor device are transmitted to the honest node of the blockchain network. However, this system does not consider the traffic that may increase in the network when the nodes are busy in replicating the processing outcome. The improvement of privacy with novel blockchain-based distributed key management scheme was discussed in [29], which eliminates the potential threat caused by a trusted third party. It uses multi-blockchain network that improves verification and saves storage space for IoT devices. The results showed that the scalability of the system is suitable to resource constrained IoT systems. However, a preshared key strategy in asymmetric cryptography is used, resulting in increased computation and communication overhead.

2.4. Blockchain Mechanisms for IoT Security

Blockchain-based frameworks to preserve user privacy in IoT have been proposed in a majority of works. The authors of [30] proposed a blockchain-based data acquisition scheme for a secure collection of data from IoT devices using Unmanned Aerial Vehicles (UAVs). This solution was researched by collecting data from IoT devices using UAV and storing safely in blockchain through mobile edge computing. However, in this approach, the required verification increases the latency. The researchers in [31] enhanced privacy in IoT with the Hyperledger Fabric Blockchain framework and Attribute Based Access Control (ABAC) to ensure efficient access control even under large number of requests in the IoT environment. The performance of this approach is analysed using two terminals which may increase the computational cost. The authors of [5] enhanced the publish/subscribe model with a blockchain-based secure publish/subscribe system to protect the privacy of publishers and subscribers. This model uses the Ethereum platform to ensure identity protection of the publisher and subscriber, using public key encryption with an equality test to guarantee the confidentiality of IoT data transmitted in the blockchain network. Though the authors present a promising way to preserve privacy in IoT system, the use of

Diffie–Hellman protocol for encryption procedure does not resist security attack, causing the user to compromise the security of their personal data.

Based on consortium blockchain, the security and privacy in IoT were enhanced in [32] with a novel attribute-based access control scheme. This scheme avoids the need to maintain an access control list in the IoT system as compared to traditional access control technologies. The access policies are made up of attributes and stored in the form of transaction in the blockchain. The performance analysis of their system shows storage overhead increases linearly with an increase in the number of attributes, whereas the computation overhead is also linear in the number of attributes. The security analysis shows that their scheme provides resistance to various security attacks in the IoT system. However, the key pair developed for authentication of the transaction does not boost the security strength of the encrypted transactions. In [14], the authors enhanced user privacy preservation in the IoT system with a novel secure mutual authentication system to provide traceability and privacy protection of access policy and user consent. The use of ECIES protects the confidentiality and privacy of request transaction message and response data that is transmitted to obtain necessary consents before data transmission in IoT. It gives a correlation coefficient of 0.34499 between constructed request transactions and encrypted transaction with a computation time of 102.733 ms. The ECIES is implemented to generate the public/private keys for encrypting and decrypting the request transaction data and response data. However, keys generated from the publicly exposed point on the elliptic curve result in violating user privacy.

A major concern regarding the adoption of blockchain technology in IoT networks is the enormous energy consumption associated with blockchains. This perception inevitably raises concerns about the further adoption of this technology, a fact that inhibits rapid uptake of what is widely considered to be a ground-breaking and disruptive innovation [33]. This fact, along with the significant increase in energy consumption caused by IoT networks has created a new challenge and diverted the focus towards creating an eco-friendlier IoT ecosystem, which provides energy efficient services and enables the production and use of renewable energy [34]. The combination of blockchains and a green IoT is focused on reducing energy consumption and adopting renewable resources rather than on energy generated by fossil fuels. Furthermore, recent studies [33,35] have shown that blanket statements about the energy consumption related to blockchains should be reviewed with care. Although Bitcoin and other proof-of-work blockchains do indeed consume a lot of power, alternative blockchain solutions with significantly lower power consumption are already available today, and new promising concepts are being tested that could further reduce the power consumption of large blockchain networks.

3. Modified ECIES with Secure Hash Function

The proposed scheme is intended to protect the integrity of transmitted messages while obtaining necessary consents for data transmission in IoT. Moreover, it provides resistance against different attacks and ensures reliable auditing of the user data access policy. To provide confidentiality and authentication of the transmitted data, both the request transaction and response data are authenticated once they are encrypted. We have chosen the proposed method in Lin et al. [14] as the basis for our designed solution. The mutual authentication system shows the access request transaction and response data while obtaining necessary consents. It protects against any data leakage and data loss, ensures reliable behavior auditing and protects the user access policy, preventing any malicious attack and possibility of consents versioning. The request transaction data are encrypted using ECIES and authenticated using message authentication code. The access request transaction and response data are firmly secured and authenticated, providing enhanced security while managing user data access policy and consents [18].

The use of an SHF to generate private and public keys prevents an attacker from detecting the actual values of the keys from which it is derived, even in the case where the hash function is known. This feature enhances the privacy preservation in IoT, providing

resistance to detect the actual value of the key is used to encrypt the message. A detailed flow diagram of the proposed scheme is shown in Figure 1. In the following, we detail the major stages involved in our proposed scheme.

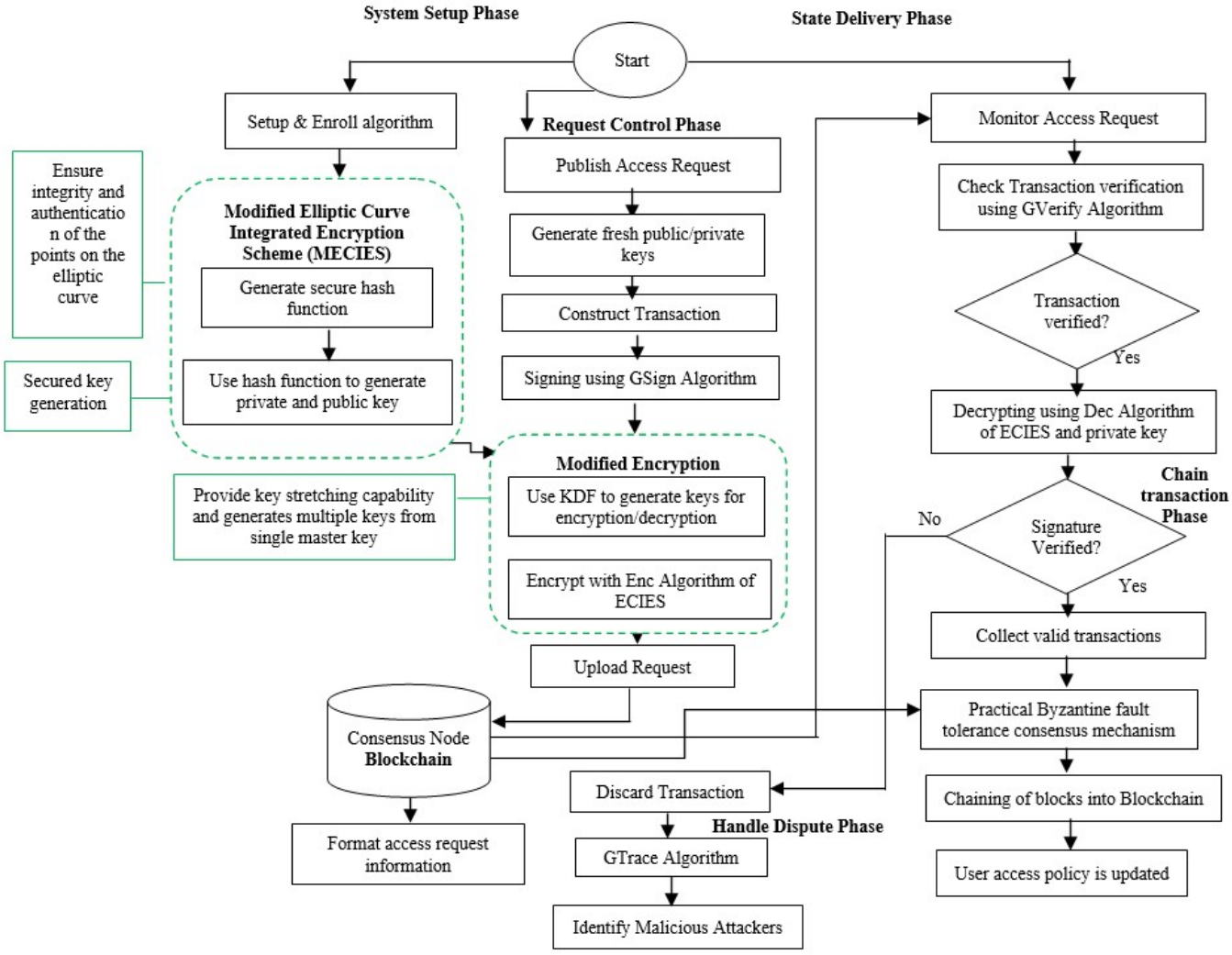

Figure 1. Flow Diagram of the Proposed Scheme—Modified ECIES with Secure Hash Function.

System Setup—The setup and enroll algorithm is invoked in this step to obtain keys for signing and verifying the transaction. After taking in the security parameters, λ^n, to obtain the public parameters, σ^n, we first generate a hash to ensure the security of the derived key, since the generated hash function is used to compute the private and public keys, denoted by δ_R and δ_P, respectively. The unique hash generation, corresponding to message m, is denoted as:

$$h(m) = \psi(m) \quad | \quad \psi : \{0,1\}^* \rightarrow \{0,1\}^{256},$$

where ψ is the unique hash generation function [30,36], and we have used SHA-256. In some works, for example, [14], the private and public key is calculated from publicly exposed points on the elliptic curve that can be easily detected by the attacker, and user privacy can be compromised. The security strength of the key ensures the confidentiality and authenticity of the transmitted message for obtaining the user consents before processing the user data.

The security strength of the key ensures the confidentiality and authenticity of the transmitted message for obtaining the user consents before processing user data in IoT.

Thus, if SHF is used to determine the value for the key rather than choosing publicly exposed points on the elliptic curve, the transmitted message will be highly protected. The secure hash value is used to generate the private key rather than randomly choosing a publicly exposed point on the elliptic curve as a private key and computing public key from the chosen private key. The private key δ_R is generated based on the hash function using the key generator function $\Gamma(\cdot)$, given as:

$$\delta_R = \Gamma(h) \quad | \quad \Gamma : \{0,1\}^* \rightarrow \{0,1\}^{\kappa},$$

where κ is the designated key size [30,36]. After the generation of private key δ_R from the hash $h(m)$, corresponding to message m, the public key δ_P is calculated based on:

$$\delta_P = \delta_R * (E_x, E_y),$$

where (E_x, E_y) corresponds to the x and y coordinates of the point P on the elliptic curve E of finite field and P has the order of large prime number q [30]. Hence, the use of the hash function protects the value of the key being detected even if the hash function is known. As a result, private and public keys are secured and provide resistance to several security attacks enhancing user privacy protection.

Request Control—Once access request is published, new public and private keys are produced to avoid replay attack and profiling [14], where the uniquely generated hash is used to compute the private key instead of a randomly chosen key. The transaction to access the data is constructed and signed using the GSign algorithm. Request transaction data are then encrypted and verified using different keys. Since the randomly generated points on the elliptic curve can be detected by any attacker as multiple keys to encrypt the transmitted message, the proposed solution uses a KDF algorithm [37] to derive multiple keys from one secured master key. KDF follows an iterative process to derive multiple keys and ensure that an attacker is not able to identify origin of the master key [32]. After keys are generated, request transaction data are encrypted using the Enc. algorithm of ECIES and is authenticated using the MAC algorithm, where the encryption process is given by:

$$C_P = Encrypt(T_r, \delta_P),$$

and C_P represents the encrypted access request transaction data that is then uploaded to the blockchain network.

State Delivery—In this phase, consensus nodes in the blockchain network monitor the access request, checking the transaction verification using a signature verification algorithm. If the transaction is verified, it is decrypted using the Dec algorithm of ECIES and private key [14,30], which provides target device information and control orders, given by:

$$(D_i, C) = Decrypt(C_P, \delta_R),$$

and D_i and C represent the target device and control information, respectively. The consensus node of the blockchain network formats the data request to ensure the data access request is received from a valid requestor. The information access request to the user and response from the user is encrypted and authenticated. The authentication tag is recomputed to ensure the response is received from a valid user. If the authentication tag matches, only then the response from the user is decrypted to obtain response information about the request.

Chain Transaction—The transactions are retrieved in the smart contract of the blockchain network, where signatures are verified to check the validity of the transaction. If the transaction is valid, they are collected, and the block is formed. The consensus nodes use the Practical Byzantine Fault Tolerance consensus mechanism to chain the blocks [38]. The user access policy is then updated, which helps in managing consents set by the user.

Dispute Handling—The unusual transactions are traced by detecting abnormal and unusual behavior, where GTrace algorithm is executed to reveal the real identity in the

unusual transactions. It helps to prevent impersonation attacks by identifying unusual behavior and showing the real identity of the attacker.

The flow of the modified ECIES is shown in Figure 2, whereas the steps of the proposed scheme are shown in Algorithm 1.

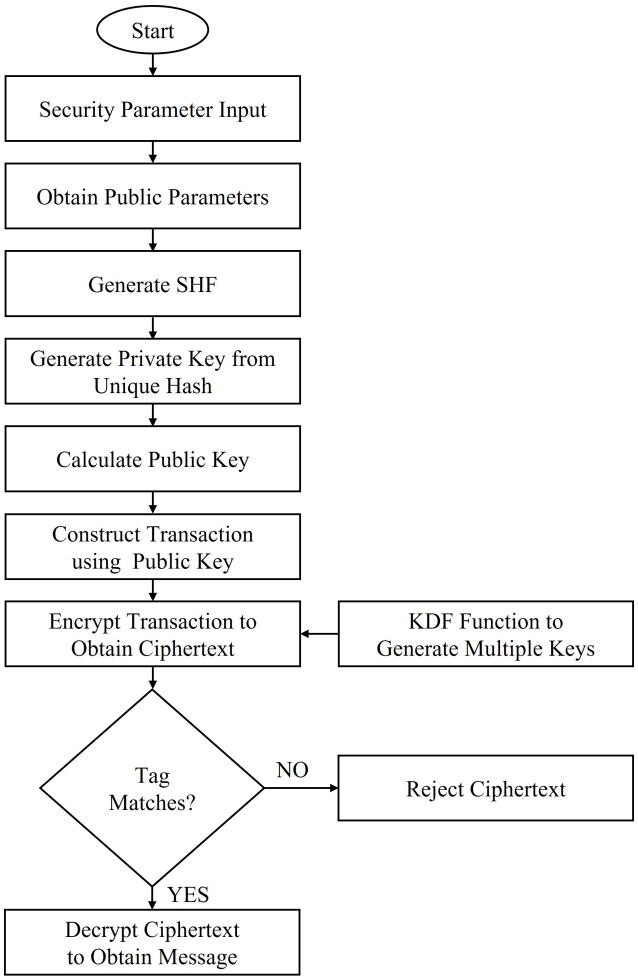

Figure 2. Flowchart of the Proposed Elliptic Curve Integrated Encryption Scheme with SHF.

Algorithm 1 Proposed ECIES with Secure Hash Utilization.

Input: Security parameter λ^n and Transactional Request Data T_r
Output: Response Data

1: **Generate** $\sigma^n \leftarrow \lambda^n$
2: **Compute** $h(m) \leftarrow m$
3: **Compute** $\delta_R \leftarrow h(m)$
4: **Compute** $\delta_P \leftarrow \delta_R * (E_x, E_y)$
5: **Construct** $T \leftarrow \delta_P$
6: **Encryption** $C_P \leftarrow Encrypt(T_r, \delta_P)$
7: **Authentication Check**
if Tags Match
$(D_i, C) \leftarrow Decrypt(C_P, \delta_R)$
else Reject C_P
end

Computation Time for Proposed Scheme

In this section, we calculate the computation time of the proposed scheme, which is given as:

$$T = T_b + T_c,$$

where T is the final computation time, T_b is a computation time for transaction generation and verification, and T_c is the initial computation time. Here, T_b is given as:

$$T_b = \sum_{i=1}^{T_r} T_r^i(t) + \sum_{i=1}^{\frac{N_s}{2}+1} N_s^i(t),$$

where $T_r^i(t)$ is a time for generation of one trade, and $N_s^i(t)$ is a time for verification by session node. Moreover, T_c is given by:

$$T_c = T_1 + T_h + T_2,$$

where T_h is the time for generation of hash function, and T_1 and T_2 correspond to the time of public/private key calculation and public parameter generation.

4. Benefits of Modified ECIES with SHF

The proposed solution helps improve the confidentiality and authenticity of the transferred message to obtain consents protected by using an SHF to generate private and public keys. This improves the correlation coefficient between transmitted messages and encrypted transactions. Along with this, it also ensures that the attacker is not able to detect the value of the key even in case hash function is known to the attacker because points on the elliptic curve are the order of a large prime number. In some of previous woks, the computation time is affected by the number of users, thus with the increase in the number of users, the computation time also increases, indicating the unreliability of the system. In the proposed scheme, the computation time is calculated by eliminating the user dependent variable, showing a higher system reliability.

SHF is utilized to generate private and public keys for improving the security strength of the transmitted message. The private key is generated from the SHF based on SHA-256, while the public key is calculated from the private key and points on the elliptic curve of the finite field that is the order of a large prime number. Hence, if the attacker tries to compute the point on the curve, they will not be able to detect the value of the key. In order to improve the efficiency of the encryption and decryption, the KDF is used to generate secured multiple keys from one master key. Some previous works [14] randomly select the publicly exposed point on the curve as a value of the key resulting in several security vulnerabilities that impact user privacy. Using publicly exposed points on the curve that are vulnerable to several attacks as a private and public key, will exploit the user privacy in IoT. Hence, the use of SHF will guarantee that the integrity of the key is protected, and the attacker is not able to detect the actual value of the key. In the proposed scheme, we have kept a regard for the authenticity and integrity protection of the transmitted message while consent management for enhancing user privacy in IoT by using ECIES with an SHF generation.

5. Results and Discussion

This section presents the analysis and results of the proposed scheme. Considering the relevance of Lin et al. [14] to our work, we provide a detailed comparison of our work with the results presented in Lin et al. [14]. MATLAB R2019a was used to implement and evaluate the prototype of the proposed model on a personal computer (PC). For the implementation, 'secp256r1' is used as the elliptic curve domain parameter [39] to develop the public parameter of the elliptic curve, whereas SHA-256 is used to secure

the hash function generation. Four groups of 50, 150, 250, 500 device information were used as a dataset, where these datasets were taken from online resources [40]. Ten samples of device information from each group are taken to construct the request transactions. We considered attributes such as device_ID, device_Type, device_Model, and device_SN (serial number) from the device information for creating the tansaction request. The completed request transaction is encrypted and decrypted for both Lin et al. [14] and the proposed scheme. The strength of the transmitted message is measured in terms of the correlation coefficient between the constructed request transaction and encrypted request transaction. The performance evaluation of the proposed scheme is based on the comparison of correlation coefficient and computation time with that of Lin et al. [14].

We note that the correlation coefficient measures the closeness between the mapped points on the elliptic curve for the constructed request transaction and encrypted request transaction. The lower the value of the correlation coefficient, the more secure the encrypted transaction. We compared samples taken from our result with the device ID attribute of the 50-device group set from the dataset. This result consisted of the encrypted transaction for request transactions in the request control stage for both Lin et al. [14] and the proposed scheme, where the comparison is based on the correlation coefficient between constructed request transactions and encrypted request transactions.

Table 1 includes the device ID attribute of three samples; the constructed request transaction for each device ID and encrypted request transaction in Lin et al. [14] and our proposed scheme. The measured correlation coefficient here improves from 0.3451 to 0.3052 in the first sample of device ID attributes, which clearly demonstrates the improved security strength of the encrypted transaction due to the lower correlation coefficient. Apart from the device ID samples, we tested other attributes of the device information such as device_Type, device_Model, and device_SN attributes. Ten samples were taken from each of the datasets of 50-, 150-, 250-, and 500-device group set. The results are obtained during the request control stages before uploading the request transaction into the smart contract of the blockchain network, and are shown in Tables 2–5, respectively. It is evident from the provided tables that the proposed solution improves the correlation coefficient between the constructed request transaction and encrypted transaction, providing increased security strength of the encrypted transaction.

We also calculate the average values of the correlation coefficient and computation time for the proposed scheme and for Lin et al. [14], as shown in Table 6. The result shows a noticeable improvement in both the correlation coefficient and the computation time compared to Lin et al. [14]. Figure 3 shows the average correlation coefficient results for the proposed scheme and for Lin et al. [14], which demonstrates the security strength of the transmitted message. The results for Lin et al. [14] are shown in blue, while the orange color indicates the result for the proposed solution. Every paired blue-orange bar represents the correlation coefficient of the 50-, 150-, 250-, and 500- device group sets with the attributes device_ID, device_Type, device_Model, and device_SN, respectively. The average correlation coefficient for the proposed scheme for device_ID samples of the 50-device group dataset is reduced to 0.30122, whereas it is 0.34499 for Lin et al. [14]. Similarly, the average correlation coefficient for device_Model samples of 250-device group dataset is also reduced to 0.30359, whereas it is 0.34853 for Lin et al. [14]. Finally, the average correlation coefficient for device_SN samples of 500-device group dataset for the proposed solution is reduced to 0.30089 comparing to the record of 0.34433 for Lin et al. [14]. We attribute the degree of improvement in the correlation coefficient to the modified private and public keys for encryption in the proposed scheme. The proposed scheme improves the correlation coefficient from 0.04344 to 0.04377 between constructed request transactions and encrypted request transaction, which shows increased security strength of the transmitted message.

Table 1. Constructed Request Transaction and Encrypted Request Transaction in Lin et al. [14] and the Proposed Scheme.

Sample	Encrypted Request Transaction Samples from Lin et al. [14]		Encrypted Request Transaction Samples—Proposed Scheme	
Device_ID	**Constructed Request Transaction**	**Encrypted Transaction**	**Constructed Request Transaction**	**Encrypted Transaction**
5c504f2863	01||pk1||5c504f2863||o	nMgxrrzzltep	01||pk1||5c504f2863||o	#M25*^gh%@sEj_N
7j533g3785	01||pk2||7j533g3785||r	VzBsirblemqxj	01||pk2||7j533g3785||r	&2bgh?+5f*63^"bL+
2p488d4936	01||pk3||2p488d4936||c	blskQohnerJk	01||pk3||2p488d4936||c	Ox32?@><ghtSE21

Table 2. Correlation Coefficient and Computation Time Comparison of Lin et al. [14] and Proposed Scheme—Device ID Samples.

S. No.	Device_ID Samples	Constructed Request Transaction	Lin et al. [14]			Proposed Scheme		
			Encrypted Transaction	**Correlation Coefficient**	**Computation Time (ms)**	**Encrypted Transaction**	**Correlation Coefficient**	**Computation Time (ms)**
1	5c504f2863	01||pk1||5c504f2863||o	nMgxrrzzltep	0.3451	108.45	#M25*^gh%@sEj_N	0.3052	97.87
2	7j533g3785	01||pk2||7j533g3785||r	VzBsirblemqxj	0.3287	102.67	&2bgh?+5f*63^"bL+	0.2881	95.35
3	2p488d4936	01||pk3||2p488d4936||c	blskQohnerJk	0.3695	110.88	Ox32?@><ghtSE21	0.3197	100.01
4	3r622h2678	01||pk4||3r622h2678||w	kGniopHcqts	0.3586	105.5	&&4*^xo78?//@br	0.3074	97.36
5	8x923a0995	01||pk5||8x923a0995||r	pxtrJvnerKlsgh	0.3218	100.3	Xx(+09%#<>P582j#	0.2821	90.8
6	5z307b2305	01||pk6||5z307b2305||o	SzhioFnopsltr	0.3524	109.25	53>BJIO@+*29_ba	0.3117	99.3
7	1k408m7277	01||pk7||1k408m7277||r	zcxvtDlfspqrv	0.3247	98.6	pM@0873##ghi++	0.2851	97.2
8	4v978x0355	01||pk8||4v978x0355||r	QlnioghTsrvbe	0.3618	96.33	ST<**3789#(j;st_bt	0.3125	91.78
9	6g388k5669	01||pk9||6g388k5669||o	twchjkioAans	0.3499	99.24	C5!(^78#"gmRb+523	0.3071	93.68
10	9s028n6082	01||pk10||9s028n6082||c	ifniodfXtcrnig	0.3374	96.11	+93x0"^&pSq*?84((+	0.2933	91.45

Table 3. Correlation Coefficient and Computation Time Comparison of Lin et al. [14] and Proposed Scheme—Device Type Samples.

S. No.	Device_ Type Samples	Constructed Request Transaction	Lin et al. [14]			Proposed Scheme		
			Encrypted Transaction	Correlation Coefficient	Computation Time (ms)	Encrypted Transaction	Correlation Coefficient	Computation Time (ms)
1	Lamp	01||pk1||lamp||o	hdlOxcjsmkbfaxb	0.3365	100.25	@2e78(^:xvyio#	0.2923	91.48
2	Fan	01||pk2||fan||c	IDvislzxkrFthjcs	0.3518	96.46	vM*{14s<"QJixh%j	0.3091	90.01
3	Air-conditioner	01||pk3||ac||r	lpCivzodalfioeLt	0.3624	109.84	##hj89!kb(**vm%l	0.3147	101.21
4	Television	01||pk4||tv||r	glaQivtsjiwecbmf	0.3267	104.3	F4!{9(&&Hjck"b_1	0.2865	96.45
5	Freezer	01||pk5||freezer||o	iozxJstovhgmcIDf	0.3378	98.8	Ox5%zkLR++8**d"	0.2934	93.26
6	Camera	01||pk6||camera||c	bchjShBixmveloz	0.3413	97.65	++fg^*294(siX3!%K	0.2984	92.68
7	Doorbell	01||pk7||doorbell||c	oxGjzbkdIvsohja	0.3649	95.38	&&59gX+jq6^^d!	0.3166	89.59
8	Door	01||pk8||door||r	mrXbjiwedjlHaMb	0.3672	94.71	3!cAm#]za!_vD8**	0.3193	89.45
9	Clock	01||pk9||clock||r	VbihKzrajioxbfk	0.3291	95.16	2!_xjdO(+"8fYios"	0.2891	89.78
10	Speaker	01||pk10||speaker||o	aKleioshBzerjioc	0.3534	97.12	56@kWx"67!++^*8)	0.3112	90.56

Table 4. Correlation Coefficient and Computation Time Comparison of Lin et al. [14] and Proposed Solutions—Device Model Samples.

S. No.	Device_ Model Samples	Constructed Request Transaction	Lin et al. [14]			Proposed Scheme		
			Encrypted Transaction	Correlation Coefficient	Computation Time (ms)	Encrypted Transaction	Correlation Coefficient	Computation Time (ms)
1	RX350	01||pk1||RX350||o	VbxdjklopStpd	0.3587	106.23	P#5!hbn2e<k"	0.3138	98.45
2	HS720A	01||pk2||HS720A||c	rbpMiosgtkbdji	0.3393	109.04	++dfg*7D$%j{	0.2954	99.34
3	ZT8808	01||pk3||ZT8808||r	pbfKlacTrxkfdv	0.3718	113.96	J9_}ndb^&10f	0.3215	103.85
4	XY290P	01||pk4||XY290P||r	Kgankobhmenx	0.3425	105.4	28g(7!kvy>?lb	0.2971	98.67
5	HDR6E	01||pk5||HDR6E||o	AchjeoPvmftugy	0.3274	104.55	"fs9!45@kcql++	0.2887	96.77
6	CBT26Z	01||pk6||CBT26Z||c	ZxjdriobstJbci	0.3368	110.75	#46e%Jcmp8!(*	0.2932	101.48
7	PB485D	01||pk7||PB485D||o	oxchksDLnfkwcy	0.3451	100.3	0x^{gno**57(%	0.2995	97.26
8	AVV56E	01||pk8||AVV56E||r	GbjiochtgjFcodef	0.3596	104.78	rDk##99!hsi_4%!	0.3152	97.73
9	BM5060	01||pk9||BM5060||c	abfdelUbjiotHny	0.3417	99.34	3!(gOx<@2dn+*>	0.2951	95.87
10	CR2030	01||pk10||CR2030||o	rvpmRtzderighj	0.3624	102.45	+8cY{&269f##k!	0.3164	98.03

Table 5. Correlation Coefficient and Computation Time Comparison of Lin et al. [14] and Proposed Solutions—Device Serial Number Samples.

S. No.	Device_ SN Samples	Constructed Request Transaction	Lin et al. [14]			Proposed Scheme		
			Encrypted Transaction	Correlation Coefficient	Computation Time (ms)	Encrypted Transaction	Correlation Coefficient	Computation Time (ms)
1	72020190805001	01||pk1||72020190805001||r	cwkzAldOxvionc	0.3472	103.75	oxK*3#"4z89!Ws<k#	0.3072	97.33
2	72020190805002	01||pk2||72020190805002||c	rcksiKlwgnoxhtVm	0.3381	107.22	@hs53!jL;("bKx>++	0.2951	99.58
3	72020190805003	01||pk3||72020190805003||r	MxjkdiyqosdGrdH	0.3564	109.55	##gP34{*oX629_jb*D	0.3115	102.67
4	72020190805004	01||pk4||72020190805004||o	ldfivrskTaovhxGc	0.3415	105.14	"lB*{@793!_jf+>VG	0.2973	98.97
5	72020190805005	01||pk5||72020190805005||o	bJoxjdlqieczgeorl	0.3261	99.34	PW(+*51U_"vz#A9<h	0.2861	96.88
6	72020190805006	01||pk6||72020190805006||c	xjloFaicehpbhowc	0.3347	109.15	9^qxc*{_fk@bi56!	0.2937	101.45
7	72020190805007	01||pk7||72020190805007||o	Lpwvnjxzaioerm	0.3641	99.62	++7Ox37"#bsT^y>*	0.3142	97.13
8	72020190805008	01||pk8||72020190805008||r	mhykdgyerioskzt	0.3572	104.01	*fV%g_h!{"6do>&r	0.3116	98.35
9	72020190805009	01||pk9||72020190805009||c	aQiocdjkguzpljXo	0.3487	98.15	Ix{&85+^dy@<>g#	0.3081	97.01
10	72020190805010	01||pk10||72020190805010||r	PfsklchioxDgerzbj	0.3293	101.73	&jc*;31k4!+M_"*5%#	0.2841	96.78

Table 6. Average Correlation Coefficient and Average Computation Time Results of Lin et al. [14] and Proposed Scheme (from tested samples).

Dataset	Samples	No. of Tests Taken	Lin et al. [14]		Proposed Scheme	
			Average Correlation Coefficient	Average Computation Time (ms)	Average Correlation Coeffcient	Average Computation Time (ms)
50-Device Group Set	Device_ID	10	0.34499	102.733	0.30122	95.48
150-Device Group Set	Device_Type	10	0.34711	98.967	0.30306	92.447
250-Device Group Set	Device_Model	10	0.34853	105.68	0.30359	98.745
500-Device Group Set	Device_SN	10	0.34433	103.766	0.30089	98.615

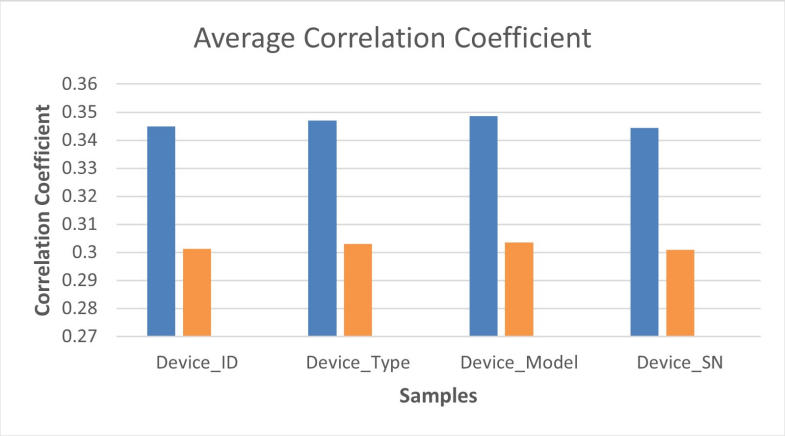

Figure 3. Average Correlation Coefficient results for Proposed Scheme and Lin et al. [14].

Figure 4 shows the average computation time results for both the proposed scheme and for Lin et al. [14] by calculating the execution time for each sample. The blue color indicates the results for Lin et al. [14], and the dark orange color indicates the result for the proposed solution. The paired blue-orange bars represent the average computation time for the 50-, 150-, 250-, and 500- device groupsets with the attributes device_ID, device_Type, device_Model, and device_SN, respectively.

- The average computation time for the proposed scheme of the device_ID samples of the 50-device group dataset is reduced to 95.48 ms, whereas it is 102.733 ms for Lin et al. [14];
- The average computation time for device_Type samples of 150-device group dataset is reduced to 92.447 ms compared to 98.967 ms of Lin et al. [14];
- The average computation time for device_Model samples of 250-device group dataset is 98.745 ms, which is less than the recorded value of 105.68 ms for Lin et al. [14];
- The average computation time for device_SN samples of 500-device group dataset. for the proposed solution is equal to 98.615 ms comparing to 103.766 ms for Lin et al. [14].

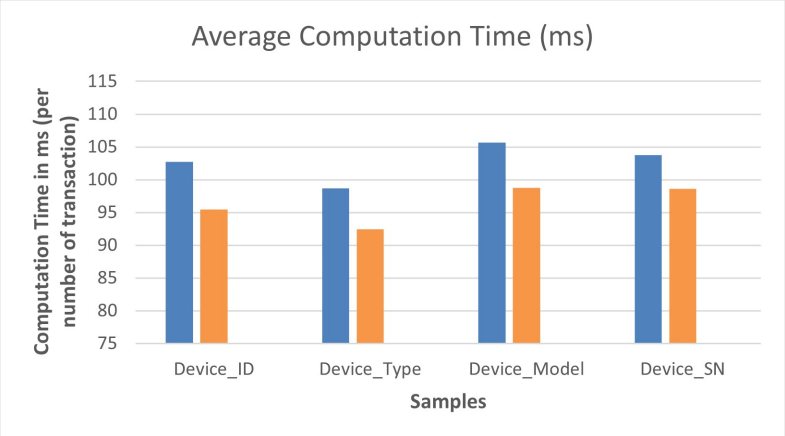

Figure 4. Average Computation Time results for Proposed Scheme and Lin et al. [14].

A comparison between our proposed scheme and Lin et al. [14] is presented in Table 7. Both solutions are based on ECIES that protect the confidentiality and privacy of request transaction messages and response data before data transmission in IoT. While Lin et al. [14] is mutually authenticated with ECIES, our proposed model modified the ECIES with an SHF. Using an SHF to derive private and public keys reduces the correlation coefficient, which improves the security strength of the request transaction data. Our contribution relies on the fact that SHF improves the strength of encryption/decryption of the transmitted message by adding new features for calculating private and public keys

from the safer elliptic curve point, as compared to the case of Lin et al. [14], which does not use hash function generation in the process of calculating the private and public keys. Moreover, in Lin et al. [14], the security strength of the key was compromised, resulting in the violation of user privacy in IoT. However, to enhance the privacy and reliability of the processed user data in IoT, the new features adopted in the proposed scheme greatly enhance user privacy in the IoT system. The use of KDF during the encryption procedure of the request control stage introduces key stretching capability in the proposed scheme, which helps to derive multiple keys from a single master key. This feature decreases the number of iterations while deriving keys for authentication. As a result, the proposed scheme achieves a reduction in encryption and decryption time. The computation time calculated in the proposed scheme eliminates user dependent variables by including time for transaction generation and verification to calculate computation time. This feature ensures the reliability of the proposed scheme with reduced computation time compared to Lin et al. [14] by an average of 7 ms per number of transactions.

Table 7. Comparison between Proposed Scheme and Lin et al. [14].

Approach	Proposed Scheme Modified ECIES with a SHF	Approach of Lin et al. [14] Mutual Authentication with ECIES
Encryption/ Decryption Strength	The strength of the encryption/decryption is measured in terms of the correlation coefficient. The improvement in the correlation coefficient is from 0.34499 to 0.30122	Provides an average correlation coefficient of 0.34499.
Computation time	Computation time is measured in terms of execution time. The computation time decreases from 102.733 ms to 95.48 ms, reducing the encryption/decryption time from 39.925 ms and 41.513 ms to 34.444 ms and 35.859 ms.	Provide an average computation time of 102.733 ms with average encryption decryption time of 39.925 ms and 41.513 ms.
Contribution 1	The generation of an SHF increases the security strength of the key by adding new features for calculating private and public keys from the safer elliptic curve points. With the generation of an SHF, the security strength of the transmitted message is improved, which enhances the user privacy in IoT.	Does not use hash function generation for computing private and public keys for encrypting the transmitted message in IoT, which results in the violation of user privacy.
Contribution 2	The KDF introduces key stretching capability and decreases the number of iterations processes while deriving keys for authentication. This reduces the time for encryption and decryption.	The computation time is affected by the number of users showing the system unreliability.

6. Conclusions and Future Work

Data security and user privacy have been the emerging needs in the IoT system. In this work, we presented a Blockchain-based scheme to preserve user privacy in IoT. The proposed scheme provides a secure platform that allows the access requester to send the request transaction data and receive the response data for the corresponding request. We propose to use ECIES with SHF, which is the new feature adapted from Lin et al. [14], to protect the confidentiality and authenticity of the transmitted request transaction and response data. The use of an SHF to derive private and public keys enhanced user privacy in IoT. This enhancement could improve the security strength of the request transaction data, which helps to derive multiple keys from the single master key; decreasing the number of iterations while deriving keys for authentication and elimination. As a result, it reduces the computation time in the proposed solution by an average of 7ms per number of transactions compared to the work of Lin et al. [14]. In the future, we need to explore other cryptographic approaches to provide a secure platform for users and data requester to exchange their data in the IoT environment. Future research needs to focus on issues other than protecting the confidentiality and authenticity of the request transaction data and response data to enhance user privacy in IoT, such as investigating and utilizing different techniques to integrate within the blockchain network for achieving enhanced privacy in the IoT system.

Author Contributions: Conceptualization, Y.P.K.; Methodology, K.S.; Project administration, A.A. and S.U.R.; Resources, S.U.R.; Supervision, A.A., A.B.A.-K., P.W.C.P., S.U.R. and R.I.; Writing—original draft, Y.P.K. and K.S.; Writing—review & editing, K.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Not applicable, the study does not report any data.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Stoyanova, M.; Nikoloudakis, Y.; Panagiotakis, S.; Pallis, E.; Markakis, E.K. A survey on the internet of things (IoT) forensics: Challenges, approaches, and open issues. *IEEE Commun. Surv. Tuts* **2020**, *22*, 1191–1221. [CrossRef]
2. Sfar, A.R.; Natalizio, E.; Challal, Y.; Chtourou, Z. A roadmap for security challenges in the Internet of Things. *Digit. Commun. Netw.* **2018**, *4*, 118–137. [CrossRef]
3. Rantos, K.; Drosatos, G.; Kritsas, A.; Ilioudis, C.; Papanikolaou, A.; Filippidis, A.P. A blockchain-based platform for consent management of personal data processing in the IoT ecosystem. *Secur. Commun. Netw.* **2019**, *2019*, 1431578. [CrossRef]
4. Fernquist, J.; Fängström, T.; Kaati, L. IoT data profiles: The routines of your life reveals who you are. In Proceedings of the European Intelligence and Security Informatics Conference (EISIC), Athens, Greece, 11–13 September 2017; pp. 61–67.
5. Lv, P.; Wang, L.; Zhu, H.; Deng, W.; Gu, L. An IoT-oriented privacy-preserving publish/subscribe model over blockchains. *IEEE Access* **2019**, *7*, 41309–41314. [CrossRef]
6. Minoli, D.; Occhiogrosso, B. Blockchain mechanisms for IoT security. *Internet Things* **2018**, *1–2*, 1–13. [CrossRef]
7. Khan, M.A.; Salah, K. IoT security: Review, blockchain solutions, and open challenges. *Future Gener. Comput. Syst.* **2018**, *82*, 395–411. [CrossRef]
8. Alfandi, O.; Khanji, S.; Ahmad, L.; Khattak, A. A survey on boosting IoT security and privacy through blockchain. *Clust. Comput.* **2020**, *24*, 37–55. [CrossRef]
9. Roy, S.; Ashaduzzaman, M.; Hassan, M.; Chowdhury, A.R. Blockchain for IoT security and management: Current prospects, challenges and future directions. In Proceedings of the IEEE International Conference on Networking, Systems and Security (NSysS), Dhaka, Bangladesh, 18–20 December 2018; pp. 1–9.
10. Bisogni, C.; Iovane, G.; Landi, R.E.; Nappi, M. ECB2: A novel encryption scheme using face biometrics for signing blockchain transactions. *J. Inf. Secur. Appl.* **2021**, *59*, 102814. [CrossRef]
11. Hammi, M.T.; Hammi, B.; Bellot, P.; Serhrouchni, A. Bubbles of Trust: A decentralized blockchain-based authentication system for IoT. *Comput. Secur.* **2018**, *78*, 126–142. [CrossRef]
12. Gai, K.; Wu, Y.; Zhu, L.; Zhang, Z.; Qiu, M. Differential privacy-based blockchain for industrial internet-of-things. *IEEE Trans. Ind. Inform.* **2019**, *16*, 4156–4165. [CrossRef]
13. Gnatyuk, S.; Kinzeryavyy, V.; Kyrychenko, K.; Yubuzova, K.; Aleksander, M.; Odarchenko, R. Secure hash function constructing for future communication systems and networks. In Proceedings of the International Conference of Artificial Intelligence, Medical Engineering, Education, Moscow, Russia, 6–8 October 2018; pp. 561–569.
14. Lin, C.; He, D.; Kumar, N.; Huang, X.; Vijayakumar, P.; Choo, K.-K.R. Homechain: A blockchain-based secure mutual authentication system for smart homes. *IEEE Internet Things J.* **2019**, *7*, 818–829. [CrossRef]
15. Yaga, D.; Mell, P.; Roby, N.; Scarfone, K. Blockchain technology overview. *arXiv* **2019**, arXiv:1906.11078.
16. Buterin, V. Ethereum White Paper: A Next Generation Smart Contract & Decentralized Application Platform; 1st version. 2014; Volume 53. Available online: https://translatewhitepaper.com/wp-content/uploads/2021/04/EthereumOrijinal-ETH-English.pdf (accessed on 9 January 2022).
17. Zou, S.; Xi, J.; Wang, H.; Xu, G. Crowdblps: A blockchain-based location-privacy-preserving mobile crowdsensing system. *IEEE Trans. Ind. Inform.* **2019**, *16*, 4206–4218. [CrossRef]
18. Xu, J.; Xue, K.; Li, S.; Tian, H.; Hong, J.; Hong, P.; Yu, N. Healthchain: A blockchain-based privacy preserving scheme for large-scale health data. *IEEE Internet Things J.* **2019**, *6*, 8770–8781. [CrossRef]
19. Debe, M.; Salah, K.; Rehman, M.H.U.; Svetinovic, D. IoT public fog nodes reputation system: A decentralized solution using Ethereum blockchain. *IEEE Access* **2019**, *7*, 178082–178093. [CrossRef]
20. Xu, X.; Zhang, X.; Gao, H.; Xue, Y.; Qi, L.; Dou, W. BeCome: Blockchain-enabled computation offloading for IoT in mobile edge computing. *IEEE Trans. Ind. Inform.* **2019**, *16*, 4187–4195. [CrossRef]
21. Kang, J.; Yu, R.; Huang, X.; Maharjan, S.; Zhang, Y.; Hossain, E. Enabling localized peer-to-peer electricity trading among plug-in hybrid electric vehicles using consortium blockchains. *IEEE Trans. Ind. Inform.* **2017**, *13*, 3154–3164. [CrossRef]
22. Zhang, A.; Lin, X. Towards secure and privacy-preserving data sharing in e-health systems via consortium blockchain. *J. Med. Syst.* **2018**, *42*, 1–18. [CrossRef]
23. Gu, J.; Sun, B.; Du, X.; Wang, J.; Zhuang, Y.; Wang, Z. Consortium blockchain-based malware detection in mobile devices. *IEEE Access* **2018**, *6*, 12118–12128. [CrossRef]

24. Cachin, C. Architecture of the hyperledger blockchain fabric. In *Workshop on Distributed Cryptocurrencies and Consensus Ledgers*; 2016; Volume 310, pp. 1–4. Available online: https://allquantor.at/blockchainbib/pdf/cachin2016architecture.pdf (accessed on 9 January 2022).
25. Androulaki, E.; Barger, A.; Bortnikov, V.; Cachin, C.; Christidis, K.; Caro, A.D.; Enyeart, D.; Ferris, C.; Laventman, G.; Manevich, Y.; et al. Hyperledger fabric: A distributed operating system for permissioned blockchains. In Proceedings of the 13th EuroSys Conference, Porto, Portugal, 23–26 April 2018; pp. 1–15.
26. Biswas, S.; Sharif, K.; Li, F.; Maharjan, S.; Mohanty, S.P.; Wang, Y. PoBT: A lightweight consensus algorithm for scalable IoT business blockchain. *IEEE Internet Things J.* **2019**, *7*, 2343–2355. [CrossRef]
27. Lu, Y.; Huang, X.; Dai, Y.; Maharjan, S.; Zhang, Y. Blockchain and federated learning for privacy-preserved data sharing in industrial IoT. *IEEE Trans. Ind. Inform.* **2019**, *16*, 4177–4186. [CrossRef]
28. He, S.; Tang, Q.; Wu, C.Q.; Shen, X. Decentralizing IoT management systems using blockchain for censorship resistance. *IEEE Trans. Ind. Inform.* **2019**, *16*, 715–727. [CrossRef]
29. Ma, M.; Shi, G.; Li, F. Privacy-oriented blockchain-based distributed key management architecture for hierarchical access control in the IoT scenario. *IEEE Access* **2019**, *7*, 34045–34059. [CrossRef]
30. Islam, A.; Shin, S.Y. BUAV: A blockchain based secure UAV-assisted data acquisition scheme in Internet of Things. *J. Commun. Netw.* **2019**, *21*, 491–502. [CrossRef]
31. Liu, H.; Han, D.; Li, D. Fabric-IoT: A blockchain-based access control system in IoT. *IEEE Access* **2020**, *8*, 18207–18218. [CrossRef]
32. Ding, S.; Cao, J.; Li, C.; Fan, K.; Li, H. A novel attribute-based access control scheme using blockchain for IoT. *IEEE Access* **2019**, *7*, 38431–38441. [CrossRef]
33. Sedlmeir, J.; Buhl, H.U.; Fridgen, G.; Keller, R. The energy consumption of blockchain technology: Beyond myth. *Bus. Inf. Syst. Eng.* **2020**, *62*, 599–608. [CrossRef]
34. Sharma, P.K.; Kumar, N.; Park, J.H. Blockchain technology toward green IoT: Opportunities and challenges. *IEEE Netw.* **2020**, *34*, 263–269. [CrossRef]
35. Sedlmeir, J.; Buhl, H.U.; Fridgen, G.; Keller, R. Recent Developments in Blockchain Technology and their Impact on Energy Consumption. *arXiv* **2021**, arXiv:2102.07886
36. Hakeem, S.A.A.; Abd El-Gawad, M.A.; Kim, H. A decentralized lightweight authentication and privacy protocol for vehicular networks. *IEEE Access* **2019**, *7*, 119689–119705. [CrossRef]
37. Krawczyk, H. Cryptographic extraction and key derivation: The HKDF scheme. In *Annual Cryptology Conference*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 631–648.
38. Sankar, L.S.; Sindhu, M.; Sethumadhavan, M. Survey of consensus protocols on blockchain applications. In Proceedings of the IEEE International Conference on Advanced Computing and Communication Systems, Coimbatore, India, 19–20 March 2017; pp. 1–5.
39. Brown, D.R. Sec 2: Recommended Elliptic Curve Domain Parameters. Standards for Efficient Cryptography, 2010. Available online: https://ci.nii.ac.jp/naid/10027922258/ (accessed on 9 January 2022).
40. Hang, L.; Kim, D.H. Design and implementation of an integrated iot blockchain platform for sensing data integrity. *Sensors* **2019**, *19*, 2228. [CrossRef] [PubMed]