applied system innovation

MDPI

Article

Securing Cyber Physical Systems: Lightweight Industrial Internet of Things Authentication (LI2A) for Critical Infrastructure and Manufacturing

Alaa T. Al Ghazo [1,*], Mohammed Abu Mallouh [1], Sa'ed Alajlouni [1] and Islam T. Almalkawi [2]

[1] Department of Mechatronics Engineering, Faculty of Engineering, The Hashemite University, P.O. Box 330127, Zarqa 13133, Jordan; mmallouh@hu.edu.jo (M.A.M.); saed@hu.edu.jo (S.A.)
[2] Computer Engineering Department, Faculty of Engineering, The Hashemite University, P.O. Box 330127, Zarqa 13133, Jordan; eslam.malkawi@hu.edu.jo
* Correspondence: alghazo@hu.edu.jo

Abstract: The increasing incorporation of Industrial Internet of Things (IIoT) devices into critical industrial operations and critical infrastructures necessitates robust security measures to safeguard confidential information and ensure dependable connectivity. Particularly in Cyber Physical Systems (CPSs), IIoT system security becomes critical as systems become more interconnected and digital. This paper introduces a novel Lightweight Industrial IoT Authentication (LI2A) method as a solution to address security concerns in the industrial sector and smart city infrastructure. Mutual authentication, authenticated message integrity, key agreement, soundness, forward secrecy, resistance to a variety of assaults, and minimal resource consumption are all features offered by LI2A. Critical to CPS operations, the approach prevents impersonation, man-in-the-middle, replay, eavesdropping, and modification assaults, according to a security study. The method proposed herein ensures the integrity of CPS networks by verifying communication reliability, identifying unauthorized message modifications, establishing a shared session key between users and IIoT devices, and periodically updating keys to ensure sustained security. A comprehensive assessment of performance takes into account each aspect of storage, communication, and computation. The communication and computing capabilities of LI2A, which are critical for the operation of CPS infrastructure, are demonstrated through comparisons with state-of-the-art systems from the literature. LI2A can be implemented in resource-constrained IIoT devices found in CPS and industrial environments, according to the results. By integrating IIoT devices into critical processes in CPS, it is possible to enhance security while also promoting urban digitalization and sustainability.

Keywords: authentication; intelligent sensors; industrial internet of things (IIoT); forward secrecy (FS); symmetric key; cybersecurity; SCADA

check for updates

Academic Editor: Claudio Zunino

Received: 13 October 2024
Revised: 3 January 2025
Accepted: 8 January 2025
Published: 14 January 2025

Citation: Al Ghazo, A.T.; Abu Mallouh, M.; Alajlouni, S.; Almalkawi, I.T. Securing Cyber Physical Systems: Lightweight Industrial Internet of Things Authentication (LI2A) for Critical Infrastructure and Manufacturing. *Appl. Syst. Innov.* **2025**, *8*, 11. https://doi.org/10.3390/asi8010011

Copyright: © 2025 by the authors. Published by MDPI on behalf of the International Institute of Knowledge Innovation and Invention. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

The development of the Industrial Internet of Things (IIoT) has initiated an evolution in which resource allocation is optimized via network connectivity and real-time data exchange among interconnected systems and devices. Within the framework of Cyber Physical Systems (CPSs), this interconnectivity is vital for the effective administration of urban services and infrastructure [1,2]. Terminal devices, including smart meters and sensors, are of utmost importance in this ecosystem as they enable communication via the Internet and streamline the exchange of data among diverse urban systems. In time-sensitive smart

city applications, however, traditional centralized approaches, which frequently rely on cloud servers for data storage, extraction, and analysis, present obstacles such as single points of failure and latency issues. To surpass these challenges and guarantee the uninterrupted functioning of smart city infrastructure, a device–edge-cloud framework has been implemented. Edge servers are strategically placed in proximity to IIoT devices to offer storage and computational assistance. This configuration effectively mitigates latency and improves the dependability of critical urban functions. By utilizing this decentralized methodology, smart cities are able to optimize the capabilities of IIoT technologies while minimizing the drawbacks linked to centralized systems. As a result, urban efficiency, sustainability, and resilience are enhanced.

As IIoT progresses towards device–edge-cloud topologies, ensuring security becomes a crucial problem due to both logical and physical changes in the system architecture [3–5]. In situations such as intelligent factories, where sensors, users, routers, and edge stations work together, it is essential to prioritize secure communication. Authentication is crucial for building confidence between entities that communicate inside complex ecosystems [6–8].

This paper introduces the Lightweight Industrial IoT Authentication (LI2A) scheme, specifically designed to address the unique security concerns commonly encountered in industrial settings employing IIoT technology. The proposed system framework contains a Trusted Authority (TA), Users (C_i), and Industrial IoT devices ($IIoT_j$), creating a resilient framework for secure communication and authentication. The approach includes crucial security elements such as mutual authentication, authenticated message integrity, key agreement, soundness, forward secrecy, robustness against various attacks, and minimum resource overhead.

When considering security in the context of IIoT, the decision between using symmetric key or public key-based cryptography is of utmost importance. This study specifically examines symmetric key authentication, with a particular emphasis on session key agreement, despite the presence of various other authentication systems. The effectiveness of symmetric key authentication systems, as discussed in [9–13], is assessed in terms of their capability to establish mutual authentication with edge nodes, while also ensuring participant anonymity and forward secrecy.

The LI2A technique stands out by specifically targeting the drawbacks of conventional symmetric key authentication schemes. Forward secrecy (FS) is implemented to guarantee that, in the event of a long-term secret key being compromised, it is not possible to retrieve earlier-session keys. This functionality improves the security of communication channels, particularly in IIoT contexts where devices have limited resources. This paper presents a thorough security assessment, showcasing the efficacy of LI2A in mitigating many possible threats like impersonation, man-in-the-middle attacks, replay attacks, eavesdropping, and modification attacks.

This paper also assesses the performance of LI2A in terms of storage requirements, communication costs, and compute complexity, in addition to considering security aspects such as mutual authentication, message integrity, and forward secrecy. Comparisons with other schemes emphasize the efficiency and appropriateness of LI2A for use in IIoT devices that have limited resources and are frequently used in industrial environments.

LI2A presents itself as a highly promising option for ensuring the security of communications in Industrial IoT settings. It provides a lightweight yet strong authentication architecture. This scheme not only fulfills rigorous security criteria but also demonstrates exceptional performance, rendering it very suitable for the dynamic field of industrial automation. The utilization of LI2A has the capacity to improve the security of industrial

systems, hence facilitating the dependable and secure incorporation of IIoT devices into critical processes.

Below is a summary of this paper's contributions:

1. The LI2A scheme contributes a tailored authentication method designed specifically for Industrial Internet of Things (IIoT) environments. It addresses the unique challenges presented by IIoT devices in industrial settings, ensuring a robust and secure communication framework.
2. LI2A introduces a comprehensive set of security features, including mutual authentication, message integrity, key agreement, soundness, forward secrecy, and resilience against diverse attacks. This ensures a high level of security, fostering mutual trust between IIoT devices and users while preventing various types of potential threats.
3. The scheme demonstrates efficiency in resource utilization, particularly with minimal storage requirements and communication overhead. This is crucial for IIoT devices with limited computing and storage capacities, making LI2A well suited for deployment in resource-constrained industrial settings.
4. LI2A's performance evaluation showcases its superiority in terms of communication and computation costs compared to alternative schemes. The scheme's lightweight design does not compromise on security, making it a practical and efficient solution for securing communications in dynamic industrial automation environments.
5. The performance evaluation of LI2A highlights its superiority in terms of communication and computation costs compared to alternative schemes. This feature enhances the confidentiality of communication and adds an additional layer of protection against potential breaches.

Related Work

Ref. [14] introduces an authentication scheme for IoT-based RFID systems, employing symmetric cryptography. The authors' approach involves a shared key between the server and tag device for mutual authentication, with key updates occurring in each session. However, concerns arise regarding the synchronicity of the shared key state. Ref. [15] devised a symmetric key-based authentication scheme for servers and sensors. In their storage-less authentication proposal, the shared key remains unchanged. Notably, auxiliary information accompanies the shared key to authenticate identities; although this information is updated each session, it is transmitted over a public channel. Consequently, if the shared key is compromised, all previous sessions become insecure. The authors of [16] extended their work with a symmetric AKA authentication scheme that ensures unlinkability. In this scenario, user equipment (UE) and home network (HN) share a pre-established key for mutual authentication. The scheme protects previous sessions by preventing the calculation of the session key, even if the pre-shared key is exposed. Ref. [17] presents an improved and lightweight authentication scheme based on [18]. The user and gateway, as well as the gateway and IoT device, share a key for message integrity protection. Importantly, the shared key remains static throughout the entire session, posing a risk where a compromised IoT device or user exposes the shared key, granting access to all messages, including the session key. Ref. [19] proposed a scheme where keys are shared between adjacent layers in the authentication process, mirroring the vulnerabilities of [17] in terms of compromised shared keys jeopardizing the security of the communication channel. Ref. [20] introduces a three-factor symmetric key authentication key agreement scheme for multiserver environments. The registration center calculates the shared key with the server based on the server's identity, using it to encrypt the authentication message for server authentication. However, the absence of updates throughout the sessions implies that an attacker gaining the shared key between the registration center and the server can

recompute all previous session keys. The authors of [21] designed an authentication scheme for unmanned vehicles, where secret keys are shared between users and ground station servers, as well as servers and drones. This scheme shares the same weakness as [20], where the exposure of shared keys jeopardizes the security of the communication channel. Ref. [22] employs hash functions and XOR operations to achieve mutual authentication, key agreement, and message integrity. To establish a secure session key between the user and the Industrial IoT device, three messages are required, built through a series of hash and XOR operations. Ref. [23] proposes a distributed control plane with elliptic curve cryptography. The distributed authentication mechanism, immune to central failure and able to detect attacks, is analyzed for security using the AVISPA tool. It relies heavily on a secure channel on the distributed plane. Ref. [24] introduces an authentication framework for smart homes and industrial environments. The authors utilized Ascon-authenticated encryption with a hash function to construct their architecture.

In contrast to existing schemes, the proposed LI2A takes a new approach to mutual authentication in IIoT environments. LI2A addresses the limitations identified in previous works by emphasizing limited interaction for seamless authentication while ensuring instant and secure communication. The proposed scheme is specifically designed to mitigate vulnerabilities caused by shared key exposure, a lack of session key updates, and a reduction in communication overhead, all of which are common flaws in many symmetric key authentication schemes. LI2A combines the advantages of limited user–device interaction with instant authentication, resulting in increased security and efficiency in the IIoT ecosystems.

2. System Model and Security Requirements

As shown in Figure 1, the system model suggested in this study has three discrete entities: (1) a Trusted Authority (TA); (2) a User (C_i); and (3) Industrial IoT devices ($IIoT_j$). Both Internet of Things (IoT) devices and users/controllers are presumed to have tamper-resistant memory, which ensures that data, including shared authentication keys, stored within this memory cannot be illegally accessed [25].

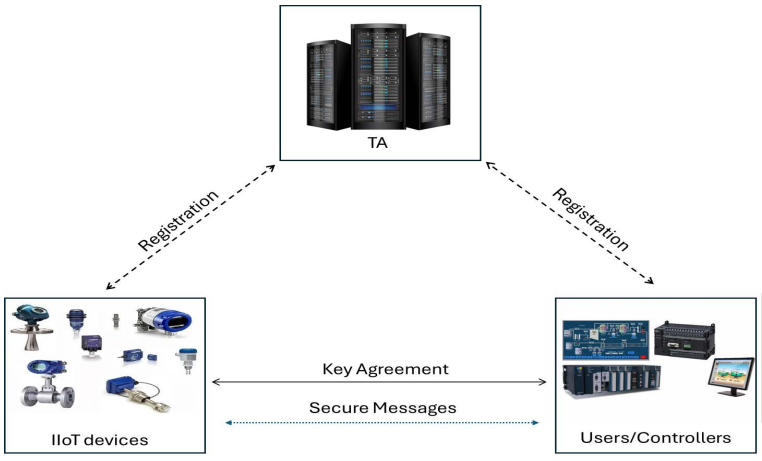

Figure 1. System model.

1. The Trusted Authority (TA) is responsible for generating system parameters and facilitating online registration. Building upon prior assumptions [26], it is postulated that TA possesses robust storage capacities, hence ensuring the protection of its computational operations against potential external vulnerabilities. As a result, TA has the capability to securely retain the identities of entities, in addition to their related authentication master keys.

2. Users (C_i): Users representing PLC/SCADA (Programmable Logic Controller/ Supervisory Control and Data Acquisition) employ the aid of a TA to verify Industrial Internet of Things (IIoT) devices during the process of registration. Afterward, they perform computations to derive a key and update their authentication keys.
3. Industrial Internet of Things ($IIoT_j$) Devices: are widely used in numerous domains and are typically described as devices with few resources that are capable of connecting to the Internet. These devices commonly rely on the delegation of their storage and computing functions to centralized servers over wireless connections for the purpose of data analysis and administration.

Within our proposed system model, users/controllers and IIoT devices initially engage in an online registration process with a TA, followed by an offline registration process with one another. When adding a new IIoT or a new user/controller (a new communication path is allowed in the system), the new device sends a message to TA for the purpose of validation. The TA sends a message to the IIoT device and the User (the communication partners). After the reception of these messages, both the User and the IIoT device proceed to generate and confirm the authentication key for future messages. The TA and $IIoT$ device selection methods ensure the soundness of the suggested strategy.

In order to establish robust security measures and safeguard privacy from potential risks, we have developed a lightweight message authentication protocol utilizing a symmetric key and incorporating forward secrecy (FS) capabilities. This approach has been meticulously designed to accommodate the unique requirements of an IIoT environment. The latest research shows that the proposed scheme must meet the following security and privacy requirements [27–30]:

- Mutual authentication: The purpose of mutual authentication is to facilitate the verification of communication reliability between entities. This measure guarantees that parties can have a high level of assurance regarding the authenticity of the communicating entity's identity while also preventing hostile parties from assuming the identities of legitimate entities.
- Ensuring authenticated message integrity: It is imperative that messages utilized for authentication purposes between entities have the capability to be identified if any unauthorized alterations or tampering have occurred.
- Key agreement: The proposed scheme aims to establish a mutually shared session key solely between the user and the Industrial Internet of Things (IIoT) device. This key will be utilized for the authentication of messages exchanged between the user and the IoT device.
- Soundness: Following the completion of a legitimate session, it is expected that the shared keys will achieve a state of synchronization. In essence, it is vital to periodically update shared authentication keys, which are used to authenticate entities and are employed to compute session keys. This practice guarantees the accuracy and uniformity of the session key.
- Forward secrecy: In the event that the derivation session key is compromised, it is imperative that unauthorized individuals are unable to gain access to previously transmitted messages and the corresponding session keys, hence guaranteeing the preservation of their confidentiality.
- Resilience against diverse attacks: The proposed scheme must possess a sufficient level of robustness to effectively withstand a range of attacks, encompassing impersonation attacks, man-in-the-middle attacks, modification attacks, replay attacks, and known session key attacks.
- Minimal resource overhead: Considering the limited resources of IoT devices, the security of the proposed scheme must be achieved with minimal computational and

communication overhead. This is essential to ensure the feasibility of the scheme within the context of IIoT devices.

Therefore, the objective of the LI2A framework is to optimize the trade-off between resource efficiency and security robustness. Specifically, the total resource consumption $\mathcal{R}$ is minimized, which includes storage overhead (R_S), communication costs (R_C), and processing requirements (R_P). This optimization is constrained by the need to ensure mutual authentication, message integrity, and forward secrecy.

It is important to acknowledge that adding devices or users/controllers to industrial systems (the scope of this paper) requires careful planning and approval. Any changes or modifications to the system can disrupt the production process and potentially impact the automation system that governs the underlying process/manufacturing. In addition, every device within the industrial system will be assigned specific communication partners that collaborate with the IIoT device to retrieve and utilize specific data for control decisions. These devices must be identified and programmed into the PLC/SCADA system during the installation process. Given that the addition of new devices to the system requires prior planning and direct oversight from system engineers, it is reasonable to assume that the channels between the TA, users, and IIoT devices are secure and free from any advisory interference during the registration phase. Furthermore, this implies that the online registration phase occurs exclusively at time t_0, which corresponds to the moment when a new device is introduced into the system.

3. Proposed LI2A Scheme

This section presents a detailed breakdown of the LI2A concept and outlines its forward secrecy. The analysis comprises the processes of system initialization, registration, and authentication with key agreement. The registration step between users (IoT devices) and TA is presumed to take place on a secure channel, whereas the remaining components of the scheme function inside an open channel. The notations utilized in the LI2A system are presented in Table 1.

Table 1. Notations utilized in LI2A.

Notation	Description
TA	Trusted Authority/system administrator
C_i	i-th user (e.g., PLC or SCADA)
$IIoT_j$	j-th Industrial IoT device
ID_i	Identity of user C_i
ID_j	Identity of device $IIoT_j$
T_i	Current timestamp
K_s	Session key
$K_{p/p}^{TA}$	TA private and public master keys
K_m	Master private key generated by TA
r_i	Random number generated by C_i
r_j	Random number generated by $IIoT_j$
$h()$	One-way hash function
$\oplus$	XOR operation (bitwise)
T_{ACK}	Acknowledgement timestamp
$h(T_{ACK} \parallel K_s)$	Integrity check hash
P	Prime number generated during initialization
τ	Current timestamp included in messages
T_{m3}	Timestamp for message m_3
ΔT	Expiration time for session keys
k_{i-1}	Previous session key during forward secrecy

3.1. Online Registration

The registration phase executed between the User C_i, the IIoT device $IIoT_j$, and the TA:

1. TA generates the master private key K_m.
2. TA selects a unique identity for C_i and $IIoT_j$, denoted as ID_i and ID_j, respectively.
3. TA generates a prime number p.
4. TA generates a random number $r_0 \leftarrow \{0,1\}^q$.
5. Once the TA generates all these parameters, it sends $\{K_m, ID_i, ID_j, r_0, P, \tau\}$ to both C_i and $IIoT_j$, where τ is the current timestamp.

Figure 2 demonstrates the online registration phase.

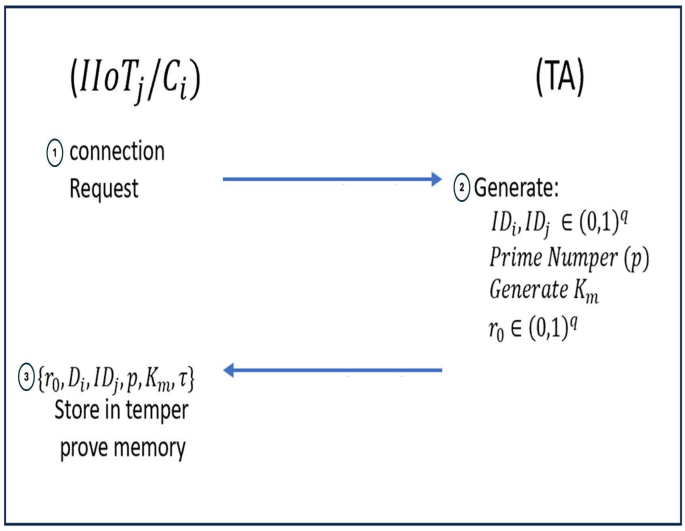

Figure 2. Online registration phase.

3.2. Key Agreement

The key agreement phase is carried out between C_i and $IIoT_j$, and after the two authenticated entities receive the initialized online information from TA, the registration stages are explained as follows:

1. Both C_i and $IIoT_j$ compute $K_s = h(\tau \| ID_i \| ID_j \| K_m \| r_0)$ and store it in the temper proof memory.
2. C_i computes $m_1 = K_s \oplus ID_i$ and $m_2 = K_s \oplus ID_j$.
3. C_i generates $m_3 = (m_1, m_2, T_{m_3}, h(T_{m_3} \| K_s))$ and sends it to $IIoT_j$. Hashing T_{m_3} with K_s ensures its integrity.
4. Once $IIoT_j$ receives m_3, it computes the following:
   (a) $ID_i = (m_1 \oplus K_s)$;
   (b) $ID_j = (m_2 \oplus K_s)$;
   (c) $T_i = h(T_{m_3} \| K_s)$.
5. $IIoT_j$ compares the calculated ID_i and ID_j to its stored communication pairs and compares the computed T_i to the received T_{m_3}. If the results match, it confirms the calculated K_s and sends a key acknowledgment message $m_4 = (T_{ACK}, h(T_{ACK} \| K_s))$ to C_i.
6. Once C_i receives m_4, it computes $T_i = h(T_{ACK} \| K_s)$ and compares the result to T_{ACK}. If the result matches, this confirms that K_s is correct and synced.
7. C_i will then choose a random time period for the timer ΔT, which represents the expiration time for K_s. Once ΔT expires, the forward secrecy stage begins.

Figure 3 demonstrates the key agreement phase.

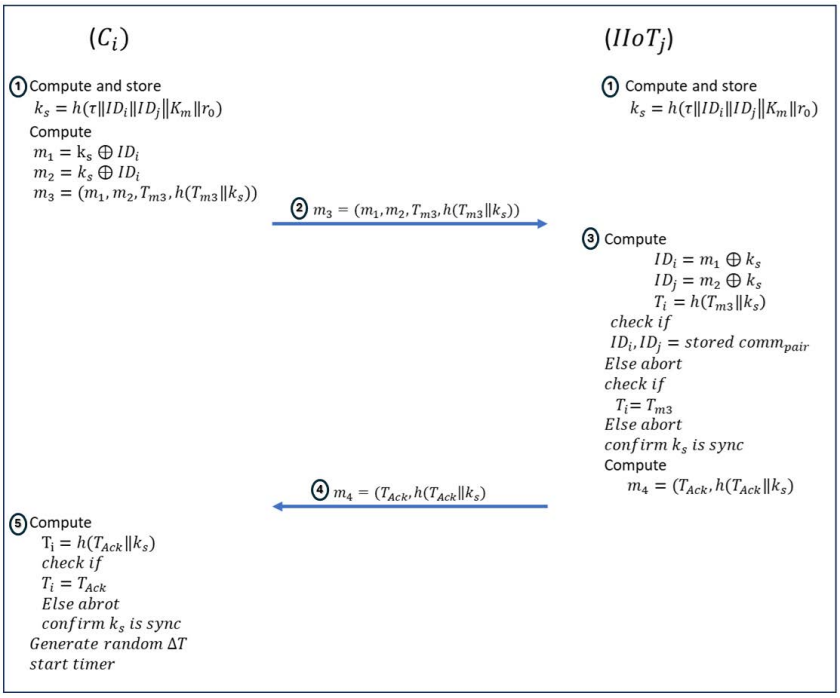

Figure 3. Key agreement phase.

3.3. Forward Secrecy

In authentication methods that use symmetric keys, session keys are often derived from shared keys. Recognized limits include the risk of insider compromise and the vulnerability to side-channel attacks. As a result, it is critical to protect previous communication sessions, emphasizing the importance of forward secrecy. This means that even if a long-term secret key is disclosed, the previous session key cannot be recovered. This feature enables the deployment of finite-lived session keys in our suggested solution for IIoT environments. In our proposed scheme, the FS stage starts every time ΔT expires. The steps for forward secrecy are as follows:

1. C_i triggers the FS session.
2. C_i generates a random number $r_i \leftarrow \{0,1\}^q$.
3. C_i computes $m_{fs_1} = r_i \oplus k_{i-1}$.
4. C_i computes $k_i = h(r \| T_i \| k_{i-1})$ and stores it.
5. C_i sends $m_{fs_2} = (m_{fs_1}, T_i, h(T_i \| k_{i-1}))$ to $IIoT_j$.
6. $IIoT_j$ computes $r_i = m_{fs_1} \oplus k_{i-1}$ and confirms the authenticity of T_i.
7. $IIoT$ computes $k_i = h(r_i \| T_i \| k_{i-1})$, stores it, and sends a key acknowledgment message $m_{fs_3} = (T_{ACK}, h(T_{ACK} \| K_s))$ to C_i.
8. Once C_i receives m_{fs_3}, it computes $T_i = h(T_{ACK} \| K_s)$ and compares the result to T_{ACK}. If the result matches, this confirms that K_s is correct and synced.
9. C_i will then choose a new random time period for the timer ΔT.

Figure 4 demonstrates the forward secrecy phase.

It should be noted that for all exchanged messages, the LI2A protocol stores the last T_i as T_{i-1} and drops any messages that have $T_i \leq T_{i-1}$.

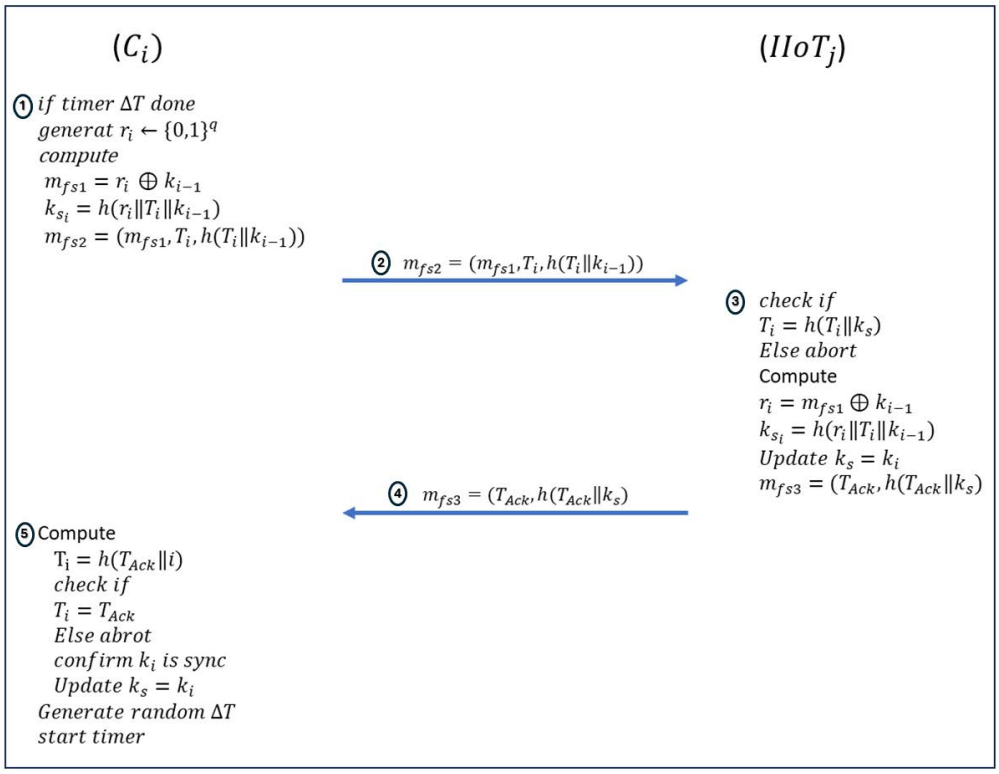

Figure 4. Forward secrecy phase.

4. Security Analysis

1. Mutual Authentication: The process of mutual authentication embedded within the LI2A framework is a cornerstone of its security architecture. This intricate mechanism ensures that $IIoT_j$ authenticates C_i and vice versa, fostering a mutual trust environment. $IIoT_j$, during the authentication process, meticulously checks the received message m_3, conducting computations such as $ID_i = (m_1 \oplus K_s)$, $ID_j = (m_2 \oplus K_s)$, and $T_i = h(T_{m_3} \| K_s)$. These computations generate unique identifiers and a hash value, forming the basis for the verification process, which $IIoT_j$ undertakes to confirm the identity of C_i. Reciprocally, C_i engages in the same authentication process, leveraging message m_4 to compute $T_i = h(T_{ACK} \| K_s)$ and ensuring its congruence with the received T_{ACK}. This mutual authentication not only establishes a secure identity confirmation but also solidifies the foundation of trust between $IIoT$ entities. Thus, the LI2A scheme not only supports mutual authentication but elevates it to a critical security parameter in IIoT ecosystems, fostering a robust and trustworthy communication paradigm.
2. Message Integrity: A fundamental aspect of secure communication is the assurance of message integrity. The LI2A scheme addresses this with a focus on the secure exchange of information between entities. Shared session keys play a pivotal role in this process, acting as cryptographic anchors that detect any unauthorized alterations to the transmitted data. The LI2A scheme employs cryptographic mechanisms to detect and prevent unauthorized changes, ensuring that the communicated information remains untampered. In the event of any deviation, the system promptly identifies and flags a verification failure, providing a robust defense against potential security breaches. This commitment to preserving the unaltered nature of communication underscores the LI2A scheme's dedication to reliable and trustable data exchanges in IIoT environments.

3. Session Key Agreement: The exclusive sharing of session keys (K_s) between Users (C_i) and IIoT devices ($IIoT_j$) is a pivotal aspect of the LI2A security framework. During the initial registration phase, $IIoT_j$ dynamically generates the session key using $K_s = h(\tau \| ID_i \| ID_j \| K_m \| r_0))$, subsequently confirming C_i's identity. Simultaneously, C_i independently computes the session key $K_s = h(\tau \| ID_i \| ID_j \| K_m \| r_0))$ and verifies its consistency by validating $m_4 = (Ack, T_{ACK}, h(T_{ACK} \| K_s))$. This intricate yet secure process ensures a robust and verifiable agreement on the session key, enhancing the overall security posture of the LI2A scheme. The session key agreement mechanism not only provides a secure foundation for communication but also establishes a dynamic and adaptive security posture that is crucial for addressing evolving threats in IIoT ecosystems.
4. Soundness: A critical security parameter is exemplified in the LI2A scheme through a meticulously designed process. Upon receiving the message from the TA, $IIoT_j$ computes the session key (K_s) and seamlessly incorporates it into the validity value m_3 for C_i's authentication. Simultaneously, C_i engages in a parallel process, computing its session key (Ks) upon receiving the message from TA and utilizing a validation mechanism. This synchronous and coordinated approach guarantees the soundness of the LI2A scheme, emphasizing the accuracy and uniformity of cryptographic key usage. The assurance of soundness not only validates the legitimacy of communication but also establishes a consistent and reliable cryptographic framework that forms the backbone of secure interactions in IIoT scenarios.
5. Forward Secrecy: The new session key K_{s_i} is determined through $k_{s_i} = h(r_i \| T_i \| k_{s_{i-1}})$, with $K_{s_{i-1}}$ representing the session key shared exclusively between C_i and $IIoT_j$. Soundness ensures that K_s undergoes updates at least once at the conclusion of each session, and the one-way secure hash function signifies that the previous $K_{s_{i-1}}$ cannot be recomputed even if the current K_{s_i} is revealed. Consequently, an adversary cannot retrieve the previous session key using the exposed K_{s_i}. The LI2A scheme upholds forward secrecy.
6. Resistance Against Various Attacks: We assess herein the resilience of our proposed LI2A scheme against impersonation attacks, man-in-the-middle attacks, replay attacks, eavesdropping attacks, modification attacks, and known session key attacks. The specific details are outlined below:
    (a) Impersonation Attack: Let us assume that adversary $\mathbb{A}$ intends to impersonate $C_i / IIoT_j$ and establish successful communication with the other entity. Consider C_i as an example; $\mathbb{A}$ must acquire C_i's current authentication session key. One possible method involves $\mathbb{A}$ stealing K_s from C_i. However, this confidential information is securely stored in C_i's tamper-proof memory, making it inaccessible to $\mathbb{A}$. Consequently, $\mathbb{A}$ cannot obtain the key and generate a valid authentication message to communicate with $IIoT_j$. Alternatively, $\mathbb{A}$ might attempt to acquire K_s from previous messages. Due to the irreversibility of the one-way hash function, $\mathbb{A}$ is unable to calculate K_s. This analysis similarly applies to impersonating $IIoT_j$. Thus, the proposed LI2A scheme demonstrates resilience against impersonation attacks. This security property also applies to internal nodes, where K_s is unique for each node.
    (b) Man-in-the-Middle Attack: An adversary, $\mathbb{A}$, has the capability to acquire exchanged messages (m_3, m_4) from senders and fabricate a valid set ($\overline{m_3}, \overline{m_4}$) to send to receivers. Thanks to the LI2A scheme's resistance to impersonation attacks and its support for mutual authentication, $\mathbb{A}$ is unable to generate valid messages that would pass through the validation process. As a result, the proposed LI2A scheme effectively guards against man-in-the-middle attacks.

(c) Replay Attack: The timestamp is integrated into the exchanged messages, and the recipient can verify their accuracy to ensure that the timestamp remains unaltered. This verification mechanism assures that malicious actor $\mathbb{A}$ does not transmit the same message repeatedly where the I2A protocol stores the last T_i as T_{i-1} and drops any messages that have $T_i \leq T_{i-1}$. As a result, the proposed LI2A method efficiently prevents replay assaults.

(d) Eavesdropping: Extracting any information from the transmission is impossible. This is due to the authentication key K_s being securely stored in tamper-proof memory. Additionally, the hash function is irreversible, making it impossible to derive K_s from the messages. As a result, no adversary can obtain K_s, and the proposed LI2A scheme effectively guards against eavesdropping attacks.

(e) Modification Attack: The message flow includes the session key K_s. The receiver can assess the potential modification of the message by examining its consistency. Consequently, any alterations would lead to verification failure, demonstrating that the proposed LI2A scheme effectively guards against modification attacks.

(f) Known Session Key Attack: Within the LI2A scheme, a known session attack refers to the scenario in which malicious actor $\mathbb{A}$ attempts to acquire another session key using their known session key. However, as the derivation of the session key undergoes updates at least once at the conclusion of each session, it becomes impossible for attacker $\mathbb{A}$ to calculate the previous or subsequent session key from the known K_s. Consequently, the proposed LI2A scheme effectively withstands known session key attacks.

This extended and comprehensive security analysis underscores the sophistication, comprehensiveness, and resilience of the proposed LI2A scheme in providing secure authentication for Industrial IoT environments. The multifaceted security features and proactive measures woven into the design exemplify its commitment to addressing a spectrum of potential security threats, ensuring the confidentiality, integrity, and authenticity of communication in IIoT ecosystems.

Security Proof

Definition 1 (Entity Authentication (EA)). *Within the security experiment, a scheme instance π_{s_i} is considered maliciously accepted with the targeted participant P_j under the following conditions:*

1. *In the r_{0th} query of the adversary, $\pi_{s_i}.\alpha = accepted$ and $\pi_{s_i}.p_{id} = P_j$.*
2. *The participants P_i and P_j are $r - corrupted$ and $r' - corrupted$ with $r_0 < r, r'$.*
3. *There are no instances like π_{s_i} and π_{t_j} that share partnership, ensuring uniqueness.*

In the above list, α signifies the instance's state (i.e., $\alpha \in$ accepted, rejected, or running), and p_{id} denotes the identity of the communication participant in the instance π_{s_i}.

Let $\mathbb{P}_{\Pi}^{AKE}(\mathbb{A})$ be the probability that adversary $\mathbb{A}$ wins the game in the authentication and key exchange. Then, let n be the number of entities, and m be the maximum number of instances per entity. Game 0: This game simulates the honest execution of the LI2A scheme. The adversary $\mathbb{A}$ tries to maliciously accept an instance π_{s_i} with a targeted participant P_j. Therefore,

$$Pr[G0] = \mathbb{P}_{LI2A}^{AKE}(\mathbb{A}) \tag{1}$$

Game 1: The challenger aborts if there is an instance with a timestamp exceeding the maximum threshold and a random number collision. Given the potential number of

random values as $n \times m$, each is randomly sampled from the set $\{0,1\}^q$. Consequently, the likelihood of having two equal random numbers is $([nm(nm-1)]/2^q)$; therefore,

$$Pr[G0] \leq Pr[G1] + (nm(nm-1)/2^q)) \tag{2}$$

Game 2: The challenger (C) tries to guess the maliciously accepted instance. The game terminates if the challenger's guess is wrong. The maximum of instances is $n \times m$; therefore,

$$Pr[G2] \leq Pr[G1] \times 1/(n \times m) \tag{3}$$

Game 3: In this game, we analyze the probability of the adversary forging messages successfully and replace the update of authentication session keys with the pseudorandom keyed hash function $h(K,.)$. At the commencement of the l_{th} session, the authentication session keys have undergone updates $l-1$ times. Considering a maximum of m sessions, the overall loss is capped at $(m-1)\mathbb{P}^{suf-k}_{hash}(C)$ during the replacement process. Therefore,

$$Pr[G3] \leq Pr[G2] + (m-1)\mathbb{P}^{suf-k}_{hash}(C) \tag{4}$$

Game 4: This game terminates if the maliciously accepted instance receives a valid message but has no matching session. We replace each update (K) using truly random functions $F_0, ..., F_{m-2}$. Synchronization of session key K_s updates at most $m-1$ times. Therefore,

$$Pr[G3] \leq Pr[G4] + (m+1)\mathbb{P}^{FS}_{update}(\beta) \tag{5}$$

where β is the adversary against a forward secrecy update.

Ultimately, the only method to induce malicious acceptance in π is by sending a valid message that did not result from any other instance, leading to the challenger's abortion. Consequently, $Pr[G4] = 0$.

According to Games 0–4, we have

$$\begin{aligned} \mathbb{P}^{AKE}_{LI2A}(\mathbb{A}) \leq & nm((nm-1)2^{-q} + (m+1)\mathbb{P}^{FS}_{update}(\beta) \\ & + (m-1)\mathbb{P}^{suf-k}_{hash}(C)) \end{aligned} \tag{6}$$

Thus, the probability of adversary $\mathbb{A}$ winning against EA security is negligible.

5. Performance

IIoT devices typically face limits in terms of both computing and storage capacity. To determine the efficiency of LI2A, it is necessary to study its storage and computation requirements. This section examines LI2A and four alternative methods [22,31–33] with a focus on communication costs, storage requirements, and computing complexity. Table 2 shows the performance evaluation of LI2A, where "Rand" represents the invocation of the random number generator and "hash" represents the operation of a one-way hash function. Please note that in analyzing the communication cost, we focus on the key agreement stage as it is the primary component of the scheme, in comparison to the online registration phase (which occurs only once and utilizes an engineering computer, as mentioned earlier, mainly involving the TA adding C_i and $IIoT_j$ to the network) and the FS stage (which requires smaller storage, message size, and computational power).

Table 2. Performance summary.

Parameter	Component	Cost
Storage	C_i	576 bits
	$IIoT_j$	576 bits
	TA	$N_{C_i} \cdot 288 + N_{IIoT_j} \cdot 288$
Computation	C_i	7 hash + 1 Rand / 5 hash + 2 Rand
	$IIoT_j$	7 hash / 5 hash
	TA	3 Rand
Communication	C_i	800 bits
	$IIoT_j$	288 bits
	TA	0 bits

Storage requirement: For each IIoT device, the TA necessitates storage comprising 256 bits for the shared authentication master key and 32 bits for the corresponding identity ID_j. Additionally, TA also requires equivalent storage to maintain the shared authentication master key and for the user identity ID_i. Conversely, the IIoT device or user must store the shared authentication master key K_m, session key K_s, ID_i, and ID_j. In total, the IIoT device or user requires a storage capacity totaling $2 * 256$ bits + $2 * 32$ bits.

Communication costs: To achieve the security level corresponding to the 1024-bits RSA algorithm, we represent the lengths of various elements as follows: $|H|$ for the hash function (256 bits), $|\oplus K|$ for the message Xored with a key (256 bits), $|q|$ for a random number (256 bits), $|ID|$ for an identifier (32 bits), and $|T|$ for a timestamp (32 bits). In the key agreement phase of the LI2A scheme, C_i initiates the process by sending a message $m_3 = (m_1, m_2, T_{m_3}, h(T_{m_3} \| K_s))$ to $IIoT_j$. $IIoT_j$ responds by sending a message $m_4 = (T_{ACK}, h(T_{ACK} \| K_s))$ to C_i. Consequently, the communication costs are as follows: 800 bits for the user side $(|\oplus K| + |\oplus K| + |T| + |H|)$, 288 bits for the $IIoT$ side $(|T| + |H|)$, and 0 bits for the TA as it only participates in the online registration. The communication costs are compared to those of other state-of-the-art schemes in Figure 5. It can be seen that LI2A outperforms these schemes in terms of overall communication cost.

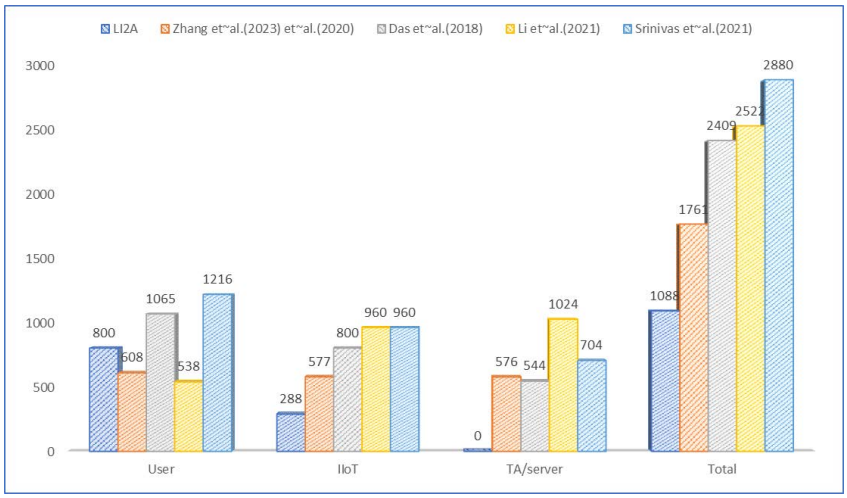

Figure 5. Communication cost comparison with state-of-the-art schemes [22,31–33].

Computation costs: We assessed the computing costs of the authentication and key agreement phase by calculating the number of significant operations, such as hash functions, scalar multiplications, and fuzzy extractors. To replicate the TA, we utilized the Ali Cloud

platform, which is equipped with an Intel Xeon CPU E5-2630, 2.30 GHz, 1-GB RAM, and 64-bit Ubuntu 14.04 operating system. A Google Nexus One smartphone with a 2 GHz ARM CPU and Android 4.4.2 was used for simulating user or IoT devices. By utilizing the Miracl library [34], we estimated the duration of execution for operations associated with schemes [22,31–33] and LI2A. The data regarding the duration of the running time are presented in Table 3. We excluded the bitwise XOR operation from our approach because it may be executed in a single cycle. The time it takes to generate a random number is estimated at $T_r \approx 8$ Th. Table 4 presents a theoretically based comparison of the required cryptographic operations and computational expenses of our system on the user side, IoT device side, and TA/server side in relation to schemes mentioned in references [22,31,32], and [33]. LI2A outperforms these schemes in terms of computational efficiency.

Table 3. Execution time of various operations.

Operation	Time (ms)
Hash function (T_h)	0.03
Random number generator (T_r)	0.24
Symmetric encryption (T_s)	0.009
Fuzzy extractor (T_f)	2.226
Scaler multiplication (T_m)	2.3
Point addition (T_a)	0.01
Bilinear pairing (T_e)	8.34

Table 4. Comparison of cryptographic operations and computational expense.

	LI2A	**[22]**	**[31]**	**[32]**	**[33]**
C_i	$7T_h + T_r / 5T_h + T_r$	$6T_h + Tr$	$14T_h + T_f$	$6T_m + 4T_a + 5T_h$	$T_f + 5T_m + T_a + 16T_h$
$IIoT_j$	$7T_h / 5T_h$	$6T_h / 4T_h / 7T_h + T_r$	$7T_h$	$T_e + 4T_m + 3T_a + 5T_h$	$4T_m + T_a + 8T_h$
TA	NA	$7T_h / 4T_h$	$9T_h$	T_s	$2T_m + 11T_h$

6. Conclusions

This work presents the Lightweight Industrial IoT Authentication (LI2A) method, specifically designed to address the security concerns that arise in Industrial Internet of Things (IIoT) contexts. The proposed system model comprises a Trusted Authority (TA), Users (C_i), and Industrial IoT devices ($IIoT_j$), establishing a robust framework for secure communication and authentication. Utilizing tamper-resistant memory for both users and devices ensures the safeguarding of authentication keys and critical data.

The proposed LI2A scheme incorporates essential security features, including mutual authentication, authenticated message integrity, key agreement, soundness, forward secrecy, resilience against diverse attacks, and minimal resource overhead. Our comprehensive security analysis demonstrated that LI2A effectively mitigates potential threats, ensuring a secure communication environment for users and IIoT devices.

The performance evaluation, which considers storage requirements, communication costs, and computation complexity, has revealed the efficiency of LI2A compared to alternative schemes cited in references [22,31–33]. The proposed scheme demonstrated superior communication and computational capabilities, making it particularly well suited for deployment in resource-constrained IIoT devices commonly found in industrial settings.

LI2A emerges as a promising and practical solution for securing communications within Industrial IoT environments. Its lightweight design achieves a balance between efficiency and security, offering a robust authentication framework tailored for the dynamic landscape of industrial automation. By addressing the unique challenges posed by IIoT,

LI2A contributes to the establishment of a secure and reliable foundation for integrating IIoT devices into critical industrial processes.

For future work, it is recommended to evaluate LI2A in real-world scenarios by applying it to specific industrial-use cases, such as securing Supervisory Control and Data Acquisition (SCADA) systems or smart manufacturing environments. This testing will provide practical insights into its performance, scalability, and adaptability in addressing sector-specific challenges. Additionally, investigating its integration with advanced IoT ecosystems, including edge computing frameworks and AI-driven predictive analytics, could further enhance its applicability and relevance in modern industrial settings.

Author Contributions: Conceptualization, A.T.A.G., M.A.M., S.A. and I.T.A.; methodology, A.T.A.G. and M.A.M.; validation, S.A. and I.T.A.; formal analysis, A.T.A.G. investigation, A.T.A.G., M.A.M., S.A. and I.T.A.; writing—original draft preparation, A.T.A.G., M.A.M., S.A. and I.T.A.; writing—review and editing, A.T.A.G., M.A.M., S.A. and I.T.A.; visualization, A.T.A.G., M.A.M., S.A. and I.T.A.; supervision, A.T.A.G.; project administration, A.T.A.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

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

References

1. Toh, C.K. Security for smart cities. *IET Smart Cities* **2020**, *2*, 95–104. [CrossRef]
2. Rathee, G.; Kumar, A.; Kerrache, C.A.; Iqbal, R. A trust-based mechanism for drones in smart cities. *IET Smart Cities* **2022**, *4*, 255–264. [CrossRef]
3. Mingozzi, E.; Tanganelli, G.; Vallati, C.; Di Gregorio, V. An open framework for accessing things as a service. In Proceedings of the 2013 16th International Symposium on Wireless Personal Multimedia Communications (WPMC), Atlantic City, NJ, USA, 24–27 June 2013; pp. 1–5.
4. Tahani, V.; Haddad Zarif, M.; Gholizadeh Narm, H. A new stable scheme against false data injection attacks in distributed control microgrid. *IET Cyber-Phys. Syst. Theory Appl.* **2024**, *9*, 27–40. [CrossRef]
5. Sahu, A.; Davis, K. Inferring adversarial behaviour in cyber-physical power systems using a Bayesian attack graph approach. *IET Cyber-Phys. Syst. Theory Appl.* **2023**, *8*, 91–108. [CrossRef]
6. Lydia, M.; Prem Kumar, G.E.; Selvakumar, A.I. Securing the cyber-physical system: A review. *Cyber-Phys. Syst.* **2023**, *9*, 193–223. [CrossRef]
7. Khalid, A.; Khan, Z.H.; Idrees, M.; Kirisci, P.; Ghrairi, Z.; Thoben, K.D.; Pannek, J. Understanding vulnerabilities in cyber physical production systems. *Int. J. Comput. Integr. Manuf.* **2022**, *35*, 569–582. [CrossRef]
8. Ali, U.; Idris, M.Y.I.B.; Frnda, J.; Ayub, M.N.B.; Khan, M.A.; Khan, N.; Jasim, A.A.; Ullah, I.; Babar, M. Enhanced lightweight and secure certificateless authentication scheme (ELWSCAS) for internet of things environment. *Internet Things* **2023**, *24*, 100923. [CrossRef]
9. Wang, F.; Xu, Y.; Zhu, L.; Du, X.; Guizani, M. LAMANCO: A lightweight anonymous mutual authentication scheme for *N*-times computing offloading in IoT. *IEEE Internet Things J.* **2018**, *6*, 4462–4471. [CrossRef]
10. Jangirala, S.; Das, A.K.; Vasilakos, A.V. Designing secure lightweight blockchain-enabled RFID-based authentication protocol for supply chains in 5G mobile edge computing environment. *IEEE Trans. Ind. Inform.* **2019**, *16*, 7081–7093. [CrossRef]
11. Nakkar, M.; AlTawy, R.; Youssef, A. Lightweight broadcast authentication protocol for edge-based applications. *IEEE Internet Things J.* **2020**, *7*, 11766–11777. [CrossRef]
12. Odelu, V.; Das, A.K.; Wazid, M.; Conti, M. Provably secure authenticated key agreement scheme for smart grid. *IEEE Trans. Smart Grid* **2016**, *9*, 1900–1910. [CrossRef]
13. Azees, M.; Vijayakumar, P.; Deboarh, L.J. EAAP: Efficient anonymous authentication with conditional privacy-preserving scheme for vehicular ad hoc networks. *IEEE Trans. Intell. Transp. Syst.* **2017**, *18*, 2467–2476. [CrossRef]
14. Mansoor, K.; Ghani, A.; Chaudhry, S.A.; Shamshirband, S.; Ghayyur, S.A.K.; Mosavi, A. Securing IoT-based RFID systems: A robust authentication protocol using symmetric cryptography. *Sensors* **2019**, *19*, 4752. [CrossRef]

15. Braeken, A. Highly efficient symmetric key based authentication and key agreement protocol using Keccak. *Sensors* **2020**, *20*, 2160. [CrossRef] [PubMed]
16. Braeken, A. Symmetric key based 5G AKA authentication protocol satisfying anonymity and unlinkability. *Comput. Networks* **2020**, *181*, 107424. [CrossRef]
17. Alzahrani, B.A.; Chaudhry, S.A.; Barnawi, A.; Xiao, W.; Chen, M.; Al-Barakati, A. ILAS-IoT: An improved and lightweight authentication scheme for IoT deployment. *J. Ambient. Intell. Humaniz. Comput.* **2020**, *13*, 5123–5135. [CrossRef]
18. Banerjee, S.; Odelu, V.; Das, A.K.; Srinivas, J.; Kumar, N.; Chattopadhyay, S.; Choo, K.K.R. A Provably Secure and Lightweight Anonymous User Authenticated Session Key Exchange Scheme for Internet of Things Deployment. *IEEE Internet Things J.* **2019**, *6*, 8739–8752. [CrossRef]
19. Ghani, A.; Mansoor, K.; Mehmood, S.; Chaudhry, S.A.; Rahman, A.U.; Najmus Saqib, M. Security and key management in IoT-based wireless sensor networks: An authentication protocol using symmetric key. *Int. J. Commun. Syst.* **2019**, *32*, e4139. [CrossRef]
20. Ali, Z.; Hussain, S.; Rehman, R.H.U.; Munshi, A.; Liaqat, M.; Kumar, N.; Chaudhry, S.A. ITSSAKA-MS: An improved three-factor symmetric-key based secure AKA scheme for multi-server environments. *IEEE Access* **2020**, *8*, 107993–108003. [CrossRef]
21. Ali, Z.; Chaudhry, S.A.; Ramzan, M.S.; Al-Turjman, F. Securing smart city surveillance: A lightweight authentication mechanism for unmanned vehicles. *IEEE Access* **2020**, *8*, 43711–43724. [CrossRef]
22. Zhang, Y.; He, D.; Vijayakumar, P.; Luo, M.; Huang, X. SAPFS: An Efficient Symmetric-Key Authentication Key Agreement Scheme With Perfect Forward Secrecy for Industrial Internet of Things. *IEEE Internet Things J.* **2023**, *10*, 9716–9726. [CrossRef]
23. Salam, R.; Roy, P.K.; Bhattacharya, A. DC-IIoT: A Secure and Efficient Authentication Protocol for Industrial Internet-of-Things Based on Distributed Control Plane. *Internet Things* **2023**, *22*, 100782. [CrossRef]
24. Tanveer, M.; Badshah, A.; Alasmary, H.; Chaudhry, S.A. CMAF-IIoT: Chaotic map-based authentication framework for Industrial Internet of Things. *Internet Things* **2023**, *23*, 100902. [CrossRef]
25. Tuyls, P.; Škoric, B.; Kevenaar, T. *Security with Noisy Data: On Private Biometrics, Secure Key Storage and Anti-Counterfeiting*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2007.
26. Gope, P.; Millwood, O.; Sikdar, B. A scalable protocol level approach to prevent machine learning attacks on physically unclonable function based authentication mechanisms for internet of medical things. *IEEE Trans. Ind. Inform.* **2021**, *18*, 1971–1980. [CrossRef]
27. Banfi, F.; Maurer, U. Anonymous symmetric-key communication. In Proceedings of the Security and Cryptography for Networks: 12th International Conference, SCN 2020, Amalfi, Italy, 14–16 September 2020; Proceedings 12; Springer: Berlin/Heidelberg, Germany, 2020; pp. 471–491.
28. Boyd, C.; Davies, G.T.; de Kock, B.; Gellert, K.; Jager, T.; Millerjord, L. Symmetric key exchange with full forward security and robust synchronization. In Proceedings of the International Conference on the Theory and Application of Cryptology and Information Security, Singapore, 6–10 December 2021; Springer: Berlin/Heidelberg, Germany, 2021; pp. 681–710.
29. Hesse, J. Separating symmetric and asymmetric password-authenticated key exchange. In Proceedings of the International Conference on Security and Cryptography for Networks, Amalfi, Italy, 14–16 September 2020; Springer: Berlin/Heidelberg, Germany, 2020; pp. 579–599.
30. Avoine, G.; Canard, S.; Ferreira, L. IoT-friendly AKE: Forward secrecy and session resumption meet symmetric-key cryptography. In Proceedings of the Computer Security–ESORICS 2019: 24th European Symposium on Research in Computer Security, Luxembourg, 23–27 September 2019; Proceedings, Part II 24; Springer: Berlin/Heidelberg, Germany, 2019; pp. 463–483.
31. Das, A.K.; Wazid, M.; Kumar, N.; Vasilakos, A.V.; Rodrigues, J.J.P.C. Biometrics-Based Privacy-Preserving User Authentication Scheme for Cloud-Based Industrial Internet of Things Deployment. *IEEE Internet Things J.* **2018**, *5*, 4900–4913. [CrossRef]
32. Li, Y.; Cheng, Q.; Liu, X.; Li, X. A Secure Anonymous Identity-Based Scheme in New Authentication Architecture for Mobile Edge Computing. *IEEE Syst. J.* **2021**, *15*, 935–946. [CrossRef]
33. Srinivas, J.; Das, A.K.; Wazid, M.; Vasilakos, A.V. Designing Secure User Authentication Protocol for Big Data Collection in IoT-Based Intelligent Transportation System. *IEEE Internet Things J.* **2021**, *8*, 7727–7744. [CrossRef]
34. SDK, M.C. MIRACL Cryptographic SDK: Multiprecision Integer and Rational Arithmetic Cryptographic Library. 2020. Available online: https://github.com/miracl/MIRACL (accessed on 5 January 2025).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.