MI3SE: A Multi-User Index-Based Searchable Symmetric Encryption Scheme for Improving Security of Data in Connected Electronic Devices

MI3SE: A Multi-User Index-based Searchable Symmetric Encryption Scheme for Improving Security of Data in Connected Electronic Devices

Seyed Ahmad Soleymani, *Member, IEEE,* Shidrokh Goudarzi, *Member, IEEE,* Mohammad Hossein Anisi, *Senior Member, IEEE,* Pei Xiao, *Senior Member, IEEE,* Lyudmila Mihaylova, *Senior Member, IEEE,* Wenwu Wang, *Senior Member, IEEE*

Abstract—In the Industrial Internet of Things (IIoT), outdoor electronic devices serve crucial roles across sectors, providing vital data for decision-making. However, their exposure to open outdoor environments makes them vulnerable to unauthorized access, physical theft, or compromise, endangering both the device and its data. Ensuring the security of outdoor devices and their data is thus critical. This study addresses data security in outdoor IIoT devices by supporting the encryption of all IIoT-related data in device memory. Accessing and retrieving this data requires operations on encrypted data. Hence, we introduce a Searchable Symmetric Encryption (SSE) scheme called MI3SE, which ensures each device's encryption key is unique and valid for a period based on the device's security sensitivity. Moreover, MI3SE meets key security requirements, including confidentiality, integrity, forward secrecy, and backward secrecy. It is specifically designed to mitigate physical compromise and query pattern analysis through a two-keyword query approach and withstand various attacks, as validated by rigorous security analysis. Comparative evaluations against benchmark schemes underscore the efficacy of MI3SE in terms of both security and performance. Moreover, comprehensive non-mathematical security analysis and simulation experiments affirm the enhanced accuracy and efficacy of MI3SE in securing sensitive data stored in outdoor IIoT devices.

Index Terms—Searchable Symmetric Encryption (SSE), IIoT, Outdoor Electronic Devices, Data Security.

1 Introduction

The Industrial Internet of Things (IIoT) has emerged as a transformative force, revolutionizing industries across the globe by leveraging the power of connected devices and advanced analytics. In today's rapidly evolving landscape, IIoT devices play a pivotal role in various sectors, serving as the backbone of digital transformation initiatives. These devices, ranging from sensors and actuators to gateways and controllers, enable organizations to collect, process, and analyze vast amounts of data in real time, thereby facilitating informed decision-making and operational optimization [1].

Along with this technological revolution, outdoor IIoT devices such as Weather sensors, Smart agricultural sensors, and Industrial equipment monitors have attracted significant attention due to their ability to extend connectivity and intelligence to remote and challenging environments. Whether deployed in agricultural fields, energy installations, or transportation networks, outdoor IIoT devices serve as the eyes and ears of modern industrial operations, capturing crucial data points that drive efficiency, productivity, and sustainability. By utilizing the capabilities of outdoor IIoT devices, industries can monitor environmental conditions, track asset performance, and mitigate operational risks in real time, thereby unlocking new paths for innovation and growth.

Despite advancements in IIoT device technology, significant security challenges persist [2]. Researchers have proposed various solutions to enhance security in IIoT environments. For instance, [3] introduced BSFR-SH, a novel blockchain-enabled security framework designed to detect and counter ransomware attacks within smart healthcare systems. In [4], an innovative Software-Defined Networking (SDN)-orchestrated Deep Learning (DL) approach was introduced for intelligent IDS in smart consumer electronics networks. Additionally, [5] addressed the threat of distributed denial of service (DDoS) attacks on IoT servers by introducing a deep reinforcement learning-based multi-layer IoT-DDoS defense system (DRL-MLDS) with robust reward metrics. These studies contribute valuable insights and solutions to improve security in IIoT ecosystems. However, the outdoor IIoT devices are susceptible to various threats due to their exposure to harsh weather conditions, potential physical tampering, and deployment in remote locations. Such conditions elevate the risk of system vulnerabilities. Their deployment in the open and outdoor environments also makes them vulnerable to physical theft, posing a direct threat to the sensitive data they store. To deal with threats like counterfeiting and unauthorized access in hardware systems and IoT devices, Physical Unclonable Functions (PUFs) can be an ussful method. PUFs utilize unique semiconductor device characteristics for cryptographic purposes, leveraging manufacturing variations to create individual and non-replicable device responses [6]. However, safeguarding the security of data stored in outdoor IIoT devices (*data-at-rest*) still becomes paramount [7]. Robust security measures and encryption protocols such as Advanced Encryption Standard (AES), Rivest-Shamir-Adleman (RSA), El-

- *S. A. Soleymani and L. Mihaylova are with the Department of Automatic Control and Systems Engineering, University of Sheffield, Sheffield, UK. E-mail: a.soleymani@sheffield.ac.uk; s.soleymani@surrey.ac.uk; l.s.mihaylova@sheffield.ac.uk;*
- *Sh. Goudarzi is with the School of Computing and Engineering, University of West London, London W5 5RF, UK. E-mail: shidrokh.goudarzi@uwl.ac.uk;*
- *M. H. Anisi is with the School of Computer Science and Electronic Engineering, University of Essex, Colchester CO4 3SQ, UK. E-mail: m.anisi@essex.ac.uk;*
- *P. Xiao is with the Institute for Communication Systems (5GIC), University of Surrey, Guildford, GU2 7XH, UK. E-mail: p.xiao@surrey.ac.uk;*
- *W. Wang is with the Centre for Vision, Speech and Signal Processing (CVSSP), University of Surrey, Guildford, GU2 7XH, UK. E-mail: w.wang@surrey.ac.uk*

liptic Curve Cryptography (ECC), and also the new block ciphers such as HARPOCRATES [8] are efficient in mitigating potential risks and ensuring the confidentiality of stored data [9].

One approach to addressing these challenges is to encrypt all data collected by IIoT devices using these encryption protocols before storing it in the device's memory [10]. This ensures that even if the device is physically stolen, the stored data remains secure. However, this method introduces a new challenge: accessing or retrieving *data-at-rest* by authorized users and other entities within the IIoT network. Since the *data-at-rest* is encrypted, processing and performing queries on this encrypted data becomes a significant hurdle.

Searchable Symmetric Encryption (SSE) schemes offer a promising solution to the challenge of querying encrypted data [11], [12]. These schemes enable efficient searching and retrieval of data while maintaining the confidentiality of the stored information. With SSE, encrypted data can be searched and retrieved without the need to decrypt it first, achieved through cryptographic techniques allowing secure keyword-based searches on encrypted data. SSE schemes typically involve generating encrypted indices or searchable data structures facilitating efficient search operations. By allowing queries directly on encrypted data, SSE schemes ensure sensitive information remains protected, crucial in scenarios where security of *data-at-rest* is paramount, such as outdoor IIoT devices. However, SSE schemes face challenges including the need for efficient search operations, strong security guarantees against cryptographic attacks, preservation of search query privacy, support for dynamic operations, scalability, and practical deployment considerations. Addressing these challenges is essential to ensure SSE schemes effectively balance security, privacy, and practicality in real-world applications.

Several studies, such as [13]–[21], have focused on safeguarding *data-at-rest* by employing SSE schemes. Kurosawa et al. [14] introduced the concept of verifiable searchable symmetric encryption along with its associated security notions, including reliability and privacy. They demonstrated that security against non-adaptive adversaries aligns with their definitions of reliability and privacy. Additionally, they addressed a limitation in SSE-2 [13] by introducing a verifiable SSE scheme that satisfies both reliability and privacy conditions, which are equivalent to security. Furthermore, they extended the scheme to support file modification, deletion, and addition of documents [15]. In [16], a keyword-based searchable symmetric encryption scheme is introduced, facilitating conjunctive keyword search and handling negative keyword search. This scheme includes the conversion of boolean queries to Searchable Normal Form and supports the execution of Boolean queries. Similarly, authors in [17] proposed a keyword-based searchable symmetric encryption scheme that achieves sub-linear complexity for worst-case disjunctive queries. Zhang et al. [18] developed an efficient and dynamic private keyword search scheme called DEPKS for inverted index–based encrypted data. This scheme has strong security notions namely statistical plaintext privacy and statistical predicate privacy. In [19] a searchable encryption scheme using the trapdoor permutation function (TPF) is proposed for cloud-based IoT. In order to prevent leakage of past query information, the notion of forward search privacy is defined in [20]. To achieve this security goal, they developed the hidden pointer technique and proposed an SSE scheme, named Khons. A dynamic searchable symmetric encryption named Eurus is designed in [21]. In this work, the linkage among queries is hidden in order to achieve strong forward and backward privacy. They also hide search patterns, size patterns, and access patterns to prevent information leakage.

Motivation: The security of *data-at-rest*, particularly within outdoor IIoT devices and server/storage systems, is a critical concern in contemporary industrial settings. The susceptibility of outdoor IIoT devices to physical theft elevates the risk of unauthorized access to sensitive information, posing significant threats to data confidentiality and integrity. Deployed in remote or open environments, these devices are exposed to heightened risks of theft and tampering, necessitating robust measures to safeguard data integrity and privacy. To address this challenge, storing encrypted *data-at-rest* in the memory or storage of IIoT devices emerges as a potential solution. However, efficient retrieval of encrypted data from storage for various operations and functions remains essential. SSE schemes offer a promising avenue to tackle this issue. Despite recent advancements, existing SSE schemes encounter notable limitations when handling new challenges related to efficient search and retrieval of encrypted data, particularly in the context of multi-user environments and two-keyword queries. These schemes often lack a structured and secure index tailored for efficient management of such challenges, resulting in significant complexities and performance issues. Consequently, there is a growing demand for SSE schemes that not only ensure robust security but also address practical challenges associated with multi-user scenarios and complex query requirements.

Contributions. This paper has the following contributions:

- Develop MI3SE, a novel multi-user index-based SSE scheme tailored to address the security and privacy challenges associated with *data-at-rest* in outdoor IIoT devices and server/storage systems, particularly in the event of device theft.
- Design and implement efficient mechanisms within the SSE scheme to enable secure and privacy-preserving search operations on encrypted data stored in IIoT devices or dedicated servers, ensuring data confidentiality, data integrity, forward secrecy, and backward secrecy.
- Conduct comprehensive security analyses, including formal security analysis using the Real-Ideal paradigm and non-mathematical security analysis, to evaluate the robustness of the proposed SSE scheme against both passive and active attacks, demonstrating its efficacy in real-world scenarios.

The rest of the paper is organized as follows. Section 2 discusses the needed background knowledge and Section 3 presents the proposed scheme. Section 4 provides the corresponding formal proof and Section 5 discusses how the proposed scheme meets the security requirements. Section 6 provides the performance evaluation and analysis of the proposed scheme. Finally, the paper concludes in Section 7.

2 Preliminaries

We define the entities acting in the proposed work along with threat models and security requirements and goals to be addressed.

2.1 System Model

In the IIoT environment, data generated by indoor and outdoor IoT devices is either stored in the device's storage or transferred to data servers for analysis and processing. The data stored in the IIoT devices and dedicated storage servers could be accessed by queries from authorized users. In this study, a data sharing system for the IIoT environment is designed with the coexistence of some entities: Proxy Server (PXS), indoor/outdoor IIoT Devices (ISD), Gateway (GW), and User

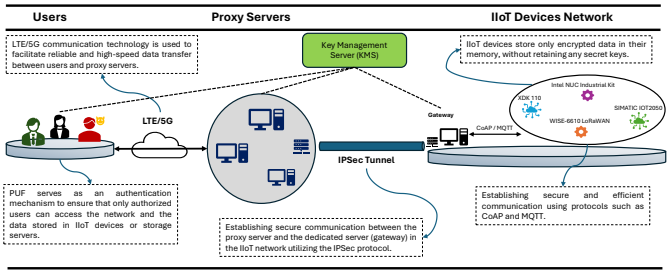


Figure 1: The proposed network architecture.

(U) as un-trusted entities as well as Key Management Server (KMS) as secure and fully trusted entity. The designed system model is depicted in Figure 1.

In this model, the user initiates communication with the ISDs to access required services and retrieve data. This involves establishing a connection to the PXSs using available access technologies such as LTE/5G. Initially, PXSs authenticate the user before enabling secure communication with ISDs. In this work, user authentication is verified using a Physical Unclonable Function (PUF) mechanism along with the user ID, password (PWD), and/or fingerprint [22]. However, it is important to note that detailed exploration of PUF mechanisms is outside the scope of this work. Subsequently, PXS evaluates the authentication process. Upon receiving a request from an authorized user, the PXS applies the query to the encrypted data stored in the ISDs via the Gateway. Utilizing secure communication through the IPSec protocol, the PXS extracts the requested documents and sends the results back to the user. In the network architecture, the communication between ISDs and the Gateway is secured using protocols such as Message Queuing Telemetry Transport (MQTT) and Constrained Application Protocol (CoAP). These protocols ensure that data transmitted between the ISDs and the Gateway is encrypted, maintaining data integrity and confidentiality during transit. Notably, ISDs only store encrypted data in their memory or storage without retaining any secret keys. When encryption is required, the ISD sends a request to the KMS to obtain the necessary encryption key. This key is valid for a specified period, such as 24 hours, after which a new encryption key must be used. The validity period can vary for each device, depending on its security sensitivity. This ensures that even if a key is compromised, the potential damage is limited to a short time frame, enhancing the overall security of the system. The periodic rotation of encryption keys also helps in mitigating the risk of long-term key exposure and strengthens the defense against potential attacks. The KMS is responsible for the generation, storage, distribution, and lifecycle management of encryption keys, ensuring that keys are kept secure and only accessible to authorized entities. This setup guarantees that even if an ISD is compromised, the data remains protected and inaccessible without the appropriate encryption keys, which are securely managed by the KMS.

2.2 Security Requirements

Our scheme aims to achieve the following primary objectives, guided by our analysis of the security and privacy challenges inherent in IIoT devices, as discussed in Section 1:

- **Data Confidentiality:** Ensuring that adversaries cannot extract any information from the transmitted or stored data, thus maintaining its confidentiality.
- **Data Integrity:** This ensures that data remains accurate, consistent, and unchanged throughout its entire duration, safeguarding against unauthorized modifications or corruption.
- **Authentication:** This process verifies the identity of users, devices, or entities attempting to access the system or resources, ensuring that only authorized entities gain entry.
- **Access Control:** This regulates and restricts access to resources based on predefined rules and policies, determining who can access specific resources and what actions they can perform, thus preventing unauthorized access and maintaining confidentiality.
- **Privacy-Preserving Queries on Encrypted Data:** Preventing adversaries from discerning the queries or inferring underlying data based on these queries, except for the access and query patterns.
- **Forward Secrecy:** It ensures that if an encryption key used to encrypt data collected by a device is compromised, previously encrypted data remains secure and cannot be decrypted by the attacker. Each set of data collected within a specific period is encrypted with a unique key, so past data remains protected even if the current key is exposed.
- **Backward Secrecy:** It ensures that if an encryption key is compromised, future data collected and encrypted by the the device remains secure. New encryption keys are generated for data collected after the compromise, ensuring that the attacker cannot decrypt data collected in the future even if they possess the old key.

2.3 Threat Model

In our threat model, we primarily consider two major security concerns: physical compromise and query pattern analysis.

- **Physical Compromise:** Physical compromise and the potential loss of complete device memory are significant threats in the context of publicly deployed IIoT devices. Attackers may gain physical access to these devices, allowing them to extract and analyze the entire memory content. This can lead to the exposure of sensitive data and encryption keys, undermining the security of the stored information. Publicly deployed devices, especially those in remote or unsecured locations, are particularly vulnerable to such attacks.
- **Query Pattern Analysis:** Another significant threat is the analysis of query patterns to infer sensitive information or gain insights into user behavior. Λ may monitor and analyze the queries made to the IIoT system, attempting to deduce valuable information about the nature of the data being accessed, user preferences, or operational patterns. By exploiting query pattern analysis, attackers can potentially compromise data privacy and security, posing a significant risk to the confidentiality and integrity of the IIoT system.

3 A Searchable Symmetric Encryption Scheme

An SSE scheme is a cryptographic technique that enables efficient searching and retrieval of encrypted data while maintaining its confidentiality [23]. It allows users to securely search over encrypted data without needing to decrypt it first. SSE schemes typically involve generating encrypted indices or searchable data structures that facilitate efficient search operations while preserving the privacy of the underlying data. A specific use case for SSE scheme in in IIoT is securing the data collected by IIoT devices during predictive maintenance operations. SSE scheme can encrypt the sensor data stored in IIoT devices, ensuring its confidentiality and integrity. Authorized users can then securely access and analyze the encrypted data

Table 1: Definition of Notations in MI3SE

Notation	Description
PXS	Proxy server
ISD	IIoT device
U	User
h	Secure one-way hash function
s	System private key
λ	Security parameter
Γ	Sequence of encrypted indexes
X	Sequence of encrypted keyword-index pairs
T_s	Search token
I_s	Ciphertext's identifier
SN	Sequences of sensor nodes
D, C	Set of documents, and encrypted documents
W_D	Set of keywords related to documents in D
WP	A set of keyword-index pair (w, ρ)
(w, ρ)	w refers to keyword and ρ is the index list
k	A secret key
k_d	A document encryption key
k_w	A keyword encryption keys.
x_q	An encrypted keyword-index pair
h_{w_q}	It refers to encrypted w_q
h_{ρ_q}	It is the encrypted index list ρ_q

to identify potential equipment issues without compromising its security. This ensures that sensitive operational data remains protected, even in the event of device theft or unauthorized access.

In this study, we developed our SSE scheme called "MI3SE". This scheme initiates the searchable encryption scheme on *data-at-rest* in IIoT devices. MI3SE involves generating search tokens from the encrypted data and using them as queries sent to the IIoT devices or data server. The IIoT devices or data server, in turn, can search the encrypted data using these tokens and retrieve the relevant encrypted documents. The definitions and notations used throughout the paper are defined in Table 1.

3.1 MI3SE: Proposed SSE Scheme

The outdoor IoT devices within the IIoT network are prone to security threats, particularly theft, which puts the stored data in these devices at risk of unauthorized access. Moreover, the proxy servers are semi-trusted entities and they can thus potentially be compromised by different adversaries. An approach to cope with this concern is to encrypt measured or sensed data prior to uploading it to the storage. However, the encryption of the data can potentially impact its usability, particularly in terms of conducting searches on the ciphertext. To address this issue, we propose a multi-user index-based and secure searchable symmetric encryption scheme based on the two-keyword query in the computing environment. MI3SE is deployed within an architectural framework comprising U as the user, PXS as the proxy server, and ISD as the IIoT device. In this setup, an authorized user can transmit a request along with the token generated by PXS and GW to the ISD. Subsequently, the ISD performs a search operation over the encrypted data stored and provides the relevant encrypted documents in response.

Considering $SN = \{sn_1, sn_2, \cdots, sn_n\}$ and $W_U = \{w_1, w_2, \cdots, w_m\}$ be respectively sequences of n sensor nodes and a set of keywords related to all devices. Let $D = \{d_1, d_2, \cdots, d_n\}$ be a set of documents where each document d_q with a unique identifier ID_q is a sequence of words from W_U. It is important to note that the documents in this context are not limited to text files, but can encompass any type of data, provided that there exists an efficient algorithm that can associate each keyword with a corresponding document from the set of documents D.

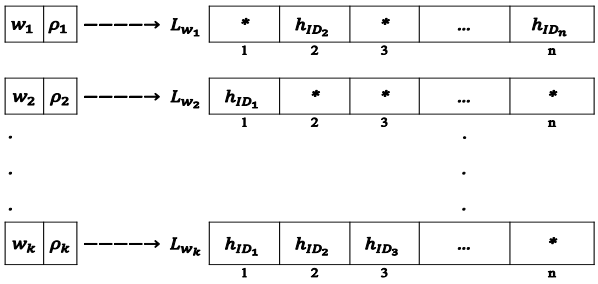


Figure 2: Keyword-index pair and related list.

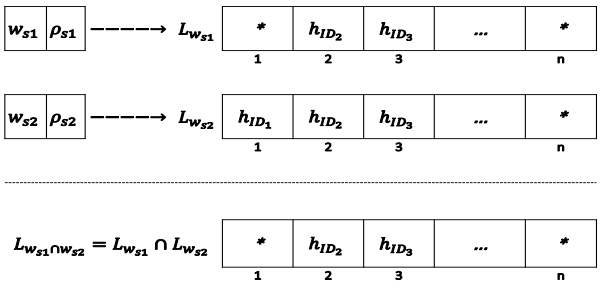


Figure 3: Intersection of two sets.

Let $W_D = \{w_1, w_2, \cdots, w_k\}$ where $W_D \subset W_U$ is a set of most important keywords related to the documents in D. Also, consider $WP = \{(w_1, \rho_1), (w_2, \rho_2), \cdots, (w_k, \rho_k)\}$ as a set of keyword-index pair (w, ρ) wherein w refers to keyword and ρ is the index list L_w. For each keyword $w_f \in W_D$, we have a sequence of encrypted documents' identifiers $L_{w_f} = \{*, h_{ID_2}, h_{ID_3}, *, \cdots, h_{ID_s}, *\}$ where i.e., ID_s is the identifier of document d_s that contains the keyword w_f and $*$ is dummy identifier. As we can see in Figure 2, it is assumed that each keyword has a list including n corresponding identifiers where $n = |SN|$. If the number of identifiers related to the keyword is less than n, we assign a dummy identifier to the keyword. We note that dummy identifiers do not represent any meaningful identifier. By this way, the differentiation in the number of keywords would not help an adversary to predict the underlying device. This provides privacy-preserving property and improves the security.

Given two keywords $w_s = \{w_{s1}, w_{s2}\}$, since our scheme is a two-keyword query, we represent $D_{w_s} \subset D$ as a sequence of documents that contains both keywords w_{s1} and w_{s2}. As shown in Figure 3, by applying the set operation intersection, $L_{w_{s1} \cap w_{s2}}$ takes only the encrypted document identifiers that are in both $L_{w_{s1}}$ and $L_{w_{s2}}$. In other words, $L_{w_{s1} \cap w_{s2}}$ keeps encrypted identifiers of documents that are in D_{w_s}.

Moreover, if $C = \{c_1, c_2, \ldots, c_n\}$ is a set of encrypted documents, then $C_{w_s} \subset C$ refers to the ciphertexts that are encryptions of the documents in D_{w_s}. The encrypted documents C along with the encrypted index Γ will be uploaded to the server storage. Besides, the encrypted WP will be stored in the storage device located at the gateway. Given a pair of value $\big(WP, (w_{q,1} \wedge w_{q,2})\big)$, our scheme returns the encrypted identifiers of documents that contain both keywords.

In a scenario, to retrieve a specific document stored in the storage of an IIoT device Dev_j, the user U_i submits a request along with a pair of keywords $w_s = \{w_{s1}, w_{s2}\}$ to a proxy server located between the user and Dev_j. Then, proxy server executes a query over encrypted keywords and returns a token search T_s as output. Next, proxy server sends the token

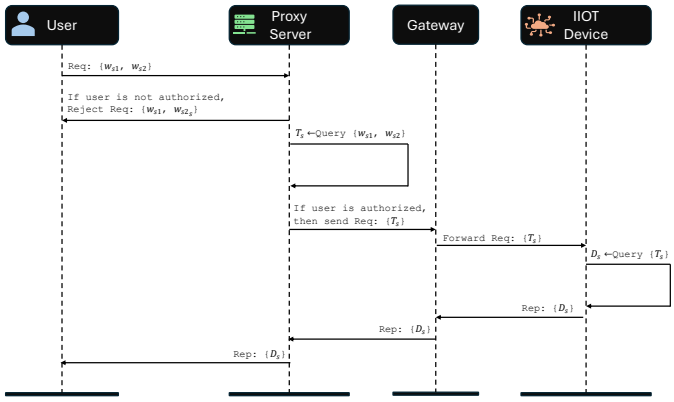


Figure 4: The process of retrieving a document from a device by a user via proxy server.

search T_s using a request to the proper Dev_j via GW. Finally, Dev_j returns a ciphertext related to the T_s after performing a query on encrypted documents (see Figure 4). Note that each user, based on access level, has a set of specific keywords and he/she can send request only on the allowable keywords. A device ignores requests that contain unallowable keywords. The access level will be set by the proxy server and it depends on the role of the user in the system. This leads to an increase in the security strength of the proposed scheme. In the following, we define our scheme.

Definition 1: *We define MI3SE consisting of the following phases. This scheme contains six Probabilistic Polynomial-Time (PPT) algorithms* MI3SE=(KeyGen,EncDoc,EncKey,SearchTokenGen,Query,Dec) *such that:*

- $K \leftarrow KeyGen(1^\lambda)$*: It inputs* λ *as security parameter and outputs* $K = (k, k_d, k_w)$ *where k is a secret key,* k_d *and* k_w *are respectively document and keyword encryption keys.*
- $(\Gamma, C) \leftarrow EncDoc\,(K,D)$*: It takes the secret key K and a sequence of documents D. The output is a sequence of encrypted indexes* Γ *related to documents and ciphertext C.*
- $X \leftarrow EncKey(K,WP)$*: It inputs the secret key K and a set of keyword-index pair WP, and returns* $X = \{x_1, x_2, \cdots, x_k\}$ *where* $x_q = \left(h_{w_q}, h_{\rho_q}\right)$ *is an encrypted keyword-index pair,* h_{w_q} *refers to encrypted* w_q *and* h_{ρ_q} *is the encrypted index list* ρ_q*.*
- $T_s \leftarrow SearchTokenGen\,(X, h_{w_{s1}} \wedge h_{w_{s2}})$*: Given the encrypted keyword-index pair in X and two encrypted keywords* $h_{w_{s1}}$ *and* $h_{w_{s2}}$*, it returns a token* T_s*. The output can be a sequence of indexes related to the corresponding documents.*
- $I_s \leftarrow Query(\Gamma, C, T_s)$*: It takes as input encrypted index* Γ*, ciphertext C, and search token* T_s *and returns as output an identifier* I_s*.*
- $d \leftarrow Dec(K, c)$*: It takes the secret key K and a ciphertext c as inputs and returns a document d as output.*

4 Formal Proof

In this section, we formally prove the proposed scheme, MI3SE, is secure under the Real-Ideal paradigm. The real-ideal paradigm is a methodological approach that used to analyse the security of a scheme by comparing its behavior in the real world (real-world execution) with its behavior in an idealized, hypothetical scenario where certain ideal conditions are assumed [24].

4.1 MI3SE Analysis Using Real-Ideal Model

Here, we prove MI3SE security under the Real-Ideal paradigm. An SSE scheme is true if the query always returns the correct set of indices that are being searched for. Given the adversary $\mathbf{\Lambda}$ and a stateful simulator S, two probabilistic games $\mathbf{Real}_{\mathbf{\Lambda}}(\lambda)$ and $\mathbf{Ideal}_{\mathbf{\Lambda},S}(\lambda)$ are defined:

$\mathbf{Real}_{\mathbf{\Lambda}}(\lambda)$: *A challenger calls* $KeyGen(1^\lambda)$ *to generate key K, and on the other hand adversary* $\mathbf{\Lambda}$ *outputs both D and WP and receives* (Γ, C) *and X such that* $(\Gamma, C) \leftarrow EncDoc_K(D)$ *and* $X \leftarrow EncKey_K(WP)$ *from the challenger. The adversary makes a polynomial number of adaptive queries* $\{w_1 \wedge w_2, T, d\}$ *and for each query q, receives either a search token* τ_s *such that* $\tau_s \in T_s \leftarrow SearchTokenGen_X(h_{w_1} \wedge h_{w_2})$*, an identifier* γ_d *such that* $\gamma_d \in I_d \leftarrow Query_{\Gamma,C}(T)$ *from the challenger. Finally,* $\mathbf{\Lambda}$ *returns a bit b that is output by the experiment.*

$\mathbf{Ideal}_{\mathbf{\Lambda},S}(\lambda)$: *Adversary* $\mathbf{\Lambda}$ *outputs D. Given* $L_1(K, WP)$*, S generates a sequence of keyword-index pair X and sends it to* $\mathbf{\Lambda}$*. The adversary makes a polynomial number of adaptive queries* $q \in \{w_1 \wedge w_2, K, WP, \Gamma, C, T_s\}$ *and, for each query q, the simulator is given either* $L_2\,(h_{w_{s1}} \wedge h_{w_{s2}})$*, or* $L_3(\Gamma, C, T_s)$*. The simulator S returns an appropriate token* τ_s^* *and an identifier* γ_d^**. Finally,* $\mathbf{\Lambda}$ *returns a bit b that is output by the experiment.*

Given the allowable leakage profile $\mathcal{L}_{MI3SE} = (\mathcal{L}_1, \mathcal{L}_2, \mathcal{L}_3)$, the MI3SE scheme is $\mathcal{L}_{MI3SE}$ − secure against adaptive chosen-keyword attacks if for all PPT adversaries $\mathbf{\Lambda}$ and a negligible function negl(λ), there exists a stateful simulator S such that

$$|\Pr\left[\mathbf{Real}_{\mathbf{\Lambda}}(\lambda) = 1\right] - \Pr\left[\mathbf{Ideal}_{\mathbf{\Lambda},S}(\lambda) = 1\right]| \leq \text{negl}(\lambda)$$

Theorem 2. *MI3SE, denoted as* $\mathbf{\Xi}$ *= (KeyGen,EncDoc,EncKey,SearchTokenGen,Query,Dec) achieves* $\mathcal{L}_{MI3SE}$ *− secure against adaptive chosen-keyword attacks in the random oracle model if MI3SE=(Enc,Gen,Dec) is CPA-secure and if H is the pseudo-random function.*

Proof 2. We establish the security of our scheme by ensuring that an adversary $\mathbf{\Lambda}$ is incapable of distinguishing between the real execution of the scheme and a simulated one, as per the construction of the simulator S.

$\mathcal{L}_1$(K, WP): In the case of keyword encryption, S can simulate an encrypted keyword by choosing a fixed size random string, since h_{w_i} in the real execution is a pseudo-random string. Similarly, the encrypted index h_{ρ_i} is also a pseudo-random string and as a result, the simulation can be conducted in the same manner. Recall that we assume each keyword has a list consisting of related documents' identifiers. To mitigate the risk of an adversary learning the keyword-based solely on differences in the number of corresponding documents, we ensure that the number of elements in each list is equal. In this case, S can simulate the list L_w related to each keyword since encrypted documents' identifiers, as elements of the list, are a pseudo-random string. In terms of encrypted document simulation by S, since every document consists of keywords and corresponding values and given keywords are a pseudo-random string, S simulates the document by choosing the keywords randomly. Since S is capable of simulating the encrypted keywords and indexes, an ideal execution of the keyword-index encryption scheme can be distinguished from other executions with only a negligible probability.

This article has been accepted for publication in IEEE Transactions on Consumer Electronics. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCE.2024.3495049

Let MI3SE be a multi-user index-based and private-key encryption scheme contains PPT algorithms and λ as security parameter. The MI3SE is IND-CPA if for all PPT adversary $\mathcal{A}$, $Adv^{ind-cpa}_{\mathrm{MI3SE},\mathcal{A}}(\lambda) = \left|\Pr\left[\mathcal{A}^{K\leftarrow KeyGen(1^\lambda),\ c\leftarrow EncDoc(K,d)} = 1\right] - \Pr\left[\mathcal{A}^{c\xleftarrow{R}\{0,1\}^*} = 1\right]\right|$ is negligible. Also, consider $H: \{0,1\}^\lambda \times \{0,1\}^* \rightarrow \{0,1\}^\lambda$ be the pseudo-random function. A function H is pseudo-random if for all PPT adversary $\mathcal{A}$, $Adv^{prf}_{H,\mathcal{A}}(\lambda) = \left|\Pr\left[\mathcal{A}^{H(K,\ \cdot),\ K\xleftarrow{R}\{0,1\}^\lambda} = 1\right] - \Pr\left[\mathcal{A}^{g(\cdot),\ g\xleftarrow{R}H} = 1\right]\right|$ is negligible. Let MI3SE = (KeyGen, EncDoc, EncKey, SearchTokenGen, Query, Dec) be our construction with six PPT algorithms as follows:

$\mathrm{K} \leftarrow \mathbf{KeyGen}(1^\lambda)$:

- Generate randomly a keyword encryption key k_w, document decryption key k_d, encryption key between GW and FEN k_{gf}, encryption key between U and FEN k_{uf}
- Set $K = (k_w, k_d, k_{gf}, k_{uf})$

$(\Gamma, \mathrm{C}) \leftarrow \mathbf{EncDoc}(\mathrm{K}, \mathrm{D})$:

- For each document d_i, compute the corresponding encrypted document as $c_i = \mathrm{Enc}(k_d, d_i, r_i)$, where r_i is a pseudo-random nonce.
- Set $C = \{c_1, c_2, \ldots, c_n\}$
- Set $\Gamma = \{\gamma_1, \gamma_2, \ldots, \gamma_n\}$ where $\gamma_i = h_{ID_i} \parallel r_i$.

$\mathrm{X} \leftarrow \mathbf{EncKey}(\mathrm{K}, \mathrm{WP})$:

- For each keyword-index pair $(w_i, \rho_i) \in WP$, compute the corresponding encrypted $\chi_i = (h_{w_i}, h_{\rho_i})$ where $h_{w_i} = H(k_w \parallel w_i)$ and $h_{\rho_i} = H(k_w \parallel \rho_i)$.
- Set $X = \{x_1, x_2, \ldots, x_k\}$.

$\mathrm{T_s} \leftarrow \mathbf{SearchTokenGen}(\mathrm{X}, \mathrm{h}_{w_{s1}} \wedge \mathrm{h}_{w_{s2}})$:

- Extract the indexes $h_{\rho_{s1}}$ and $h_{\rho_{s2}}$ related to $h_{w_{s1}}$ and $h_{w_{s2}}$ from X.
- Extract L_{s1} and L_{s2} related to the $h_{\rho_{s1}}$ and $h_{\rho_{s2}}$, respectively.
- Compute $L_{s1 \wedge s2} = L_{s1} \cap L_{s2}$ using the conjunctive operation.
- Compute $\tau_i = L_{s1 \wedge s2}[i] \parallel r_i$.
- Set $T_s = \{\tau_1, \tau_2, \ldots, \tau_n\}$ and returns T_s.

$\mathrm{I_s} \leftarrow \mathbf{Query}(\Gamma, \mathrm{C}, T_s)$:

- If $\tau_i \in T_s$ is a member in Γ , retrieves the γ_j related to τ_i.
- Set $\mathrm{I_s}$ as a sequence of indexes of the related ciphertexts.

$\mathrm{d} \leftarrow \mathbf{Dec}(\mathrm{K}, \mathrm{c})$:

- Decrypt ciphertext c_i.

Figure 5: Construction of MI3SE scheme.

$\mathcal{L}_2(\mathrm{h}_{w_{s1}} \wedge \mathrm{h}_{w_{s2}})$: In the search token generation, the scheme discloses two encrypted keywords. In order to simulate the search token generation, the simulator S makes a list of search tokens. Once S receives a request for token generation, it checks whether this token has been generated previously or not. If yes, S retrieves the matching token from the list, otherwise, it selects a search token randomly from one of the tokens and returns it to the user and in addition adds new token to the list.

$\mathcal{L}_3(\Gamma, \mathrm{C}, \mathrm{T}_s)$: The scheme discloses the query tuple $(\Gamma, \mathrm{C}, \mathrm{T}_s)$ in the case of a query. The tuple consists of a sequence of encrypted indexes, ciphertexts, and a search token. The simulator S initializes a list to simulate the query results, which stores the index of the ciphertext and the corresponding matching query. This allows the simulator S to simulate previous queries. When a new query is received, S checks whether the same query has already been made or not. If so, S returns to the user after retrieving the matched query from the list. Otherwise, S returns a ciphertext by choosing an encrypted index randomly from one of the encrypted indexes. Finally, this query along with the query's output will be inserted into the list.

Based on the previous statement, it becomes evident that an adversary gains knowledge of the leakage profile when it successfully accesses both the stored encrypted keyword-index pair and the encrypted query. In addition, the indexes and the keywords, as well as the encrypted documents kept in the device storage, don't permit this information to be accessed by an adversary by understanding the underlying keywords unless the profile leaks. Furthermore, our SSE ensures that such attacks from proxy servers are not carried out since it is a randomly encrypted index. In the case of predict privacy [18], as the security notion of randomized SSE schemes, our scheme guarantees the privacy of a user's search pattern since it uses randomly generated tokens.

5 Discussion

This section discusses how MI3SE fulfils the security requirements discussed in Section 2.2. Also, Comparison with the proposed work and related works in the IIoT environment is provided [10], [25], and [26].

Table 2 provides a comparison of the security and functional features between our scheme and other relevant schemes. In the table, ● entry denotes that the scheme fulfills the corresponding requirement, while ○ indicates that the scheme does not meet the desired objective. As illustrated in this table, MI3SE and [10] support user authentication, and formal security proof features. The MI3SE and [10], [25], [26] resist data modification attacks. Besides, the formal security proof is not supported in the schemes [10]. The MI3SE and the schemes in [10], [26], under privacy-query and *data-at-rest* provide data privacy. The aforementioned comparison reveals that our scheme satisfies all of the security requirements, whereas the benchmark schemes only fulfill a subset of the security requirements and as a result, our proposed solution outperforms other schemes in terms of functionality features and security.

5.1 Data Confidentiality

MI3SE ensures data confidentiality through robust encryption mechanisms. All data stored in IIoT devices and server/storage systems is encrypted using strong algorithms, such as AES and HARPOCRATES. The choice of encryption algorithm depends on specific security requirements and the deployment environment. While both HARPOCRATES and AES are designed for data encryption and decryption, HARPOCRATES offers a more dynamic and potentially secure approach with its key-driven substitution mechanisms and novel structural design. Conversely, AES is a well-established and widely implemented cipher, supported by extensive analysis and optimization across various platforms. Regardless of the encryption algorithm used,

Table 2: Comparison of security and functional features.

Security Attributes	[10]	[25]	[26]	MI3SE
Data Confidentiality	●	●	●	●
Data Integrity	●	○	○	●
Authentication	●	○	○	●
Access Control	○	○	○	●
Privacy-Preserving Query	●	●	●	●
Forward and Backward Secrecy	○	○	○	●

MI3SE ensures that the secret keys for encryption and decryption are never stored in the memory of IIoT devices. Instead, these keys are securely managed and kept separate from the devices utilizing KMS. By securely managing and separating the keys, the encryption process becomes robust, and only encrypted data is stored on the devices. Since the encryption keys are dynamically retrieved from the secure environment during the encryption and decryption processes, this method provides a strong safeguard. Even if the devices are physically stolen or compromised, the data stored on them remains protected and inaccessible to unauthorized parties, ensuring that encrypted data cannot be decrypted without authorization. Therefore, access to the encrypted data is restricted to authenticated users, and unauthorized access attempts are prevented through authentication mechanisms such as biometric verification and user credentials. By implementing these security measures, MI3SE effectively safeguards the confidentiality of data-at-rest, mitigating the risk of unauthorized access and data breaches.

5.2 Data Integrity

MI3SE ensures the integrity of encrypted data throughout its lifecycle using robust cryptographic mechanisms that prevent unauthorized modification or tampering. Any attempt to alter the encrypted data results in data corruption, thereby preserving its integrity. Additionally, each ISD regularly sends a copy of the encrypted data to the proxy server via the gateway. This periodic transmission ensures that even if a device is physically compromised, the data remains accessible and secure at the proxy server. By maintaining a backup of encrypted data at the proxy server, MI3SE provides redundancy and enhances security, making it difficult for attackers to access or tamper with sensitive information stored in the ISDs.

5.3 Authentication

MI3SE employs robust authentication mechanisms to verify the identity of users before granting them access to encrypted data. This ensures that only authorized users with the proper credentials or biometric information can access sensitive information stored in IIoT devices and server/storage systems.

5.4 Access Control

MI3SE enforces strict access control policies to regulate access to encrypted data based on user privileges and permissions. Authorized users are granted access to specific data based on their roles and permissions, while unauthorized users are denied access. This helps prevent unauthorized access and ensures that sensitive data is only accessible to authorized personnel.

5.5 Encrypted Data: Privacy-Preserving Queries

MI3SE ensures privacy-preserving queries on encrypted data through its innovative design and cryptographic techniques. The scheme allows users to perform queries on encrypted data stored in IIoT devices and server/storage systems without compromising the confidentiality of the underlying data. This is achieved through the use of secure search token generation and query processing mechanisms. When a user initiates a query, MI3SE generates a secure search token based on the query keywords provided by the user. This search token is used to securely search the encrypted data without revealing any sensitive information about the query or the underlying data. The search token is designed to preserve the privacy of the query, ensuring that only authorized users can access the search results. Additionally, MI3SE employs advanced cryptographic techniques such as homomorphic encryption and secure multiparty computation to perform query processing on encrypted data. These techniques enable the scheme to execute complex queries on encrypted data without decrypting it first, further enhancing the privacy and security of the query process. By enabling privacy-preserving queries on encrypted data, MI3SE ensures that sensitive information remains protected from unauthorized access or disclosure, while still allowing authorized users to retrieve relevant data for analysis and decision-making purposes. This capability is crucial for maintaining the confidentiality of data-at-rest in IIoT environments and safeguarding against potential privacy breaches

5.6 Forward Secrecy

MI3SE achieves forward secrecy by ensuring that encryption keys used for data encryption are dynamically generated and have a limited lifespan. When encryption is required, the ISDs send a request to the KMS to obtain a unique encryption key that is valid for a specified period, such as 24 hours. After this period, the encryption key is discarded, and a new key is requested for any subsequent encryption processes. This approach ensures that even if an encryption key is compromised, only the data encrypted during its validity period is at risk. All previously encrypted data remains secure, as past encryption keys are not stored and cannot be used to decrypt older data. This key management strategy effectively prevents attackers from decrypting historical data, thereby maintaining forward secrecy. However, frequent key rotation may add to both the computational and communication overhead. To address this challenge, an adaptive key management strategy is proposed, where the key rotation period is dynamically adjusted based on the sensitivity of the data and its access patterns. For example, less sensitive or infrequently accessed data may use a longer key rotation period, thereby reducing the overhead without compromising security. Through these measures, MI3SE strikes a balance between robust security and operational efficiency, ensuring that the benefits of forward secrecy are realized without imposing excessive burdens on the system's performance.

5.7 Backward Secrecy

MI3SE ensures backward secrecy by regularly updating the encryption keys and not reusing old keys for new data encryption. After the encryption key's validity period expires, a new key is generated and used for subsequent data encryption. This means that even if a current encryption key is compromised, future data remains secure because it will be encrypted with a new, uncompromised key. The KMS manages the lifecycle of encryption keys, ensuring that keys are securely generated, distributed, and retired. By preventing the reuse of old keys and ensuring that new data is always encrypted with fresh keys, MI3SE effectively safeguards future data against any compromise of previous encryption keys, thereby maintaining backward secrecy.

6 Performance Evaluation and Analysis

To evaluate the performance and security of the proposed MI3SE scheme, we conducted extensive simulations using a carefully configured and robust environment. The MI3SE scheme was implemented on the Java platform, utilizing the JPBC library version Pbc-05.14 for cryptographic operations and the Java Cryptography Extension (JCE) library for encryption and decryption functionalities. These simulations were performed on a Windows 11 system with a 13th Gen Intel Core i7-13700H processor and 16 GB of memory. The simulation environment was designed to closely mimic real-world scenarios, allowing for comprehensive testing of MI3SE under various conditions, including different network configurations, data sizes, and user behaviors. A key aspect of our simulation involved the use of the Enron Email Dataset [26], a real-world dataset that provides a rich source of text-based data, making it ideal for simulating scenarios related to SSE schemes. The Enron Email Dataset, which consists of email communications from Enron Corporation employees, is commonly employed in research for tasks such as information retrieval, text classification, and sentiment analysis. By leveraging this dataset, we were able to simulate realistic use cases and thoroughly assess the performance and security of MI3SE in practical settings. Through these detailed simulations, we demonstrate the capability of MI3SE to handle real-world challenges effectively while maintaining robust security standards.

6.1 MI3SE Evaluation

In this evaluation, we analyze MI3SE with respect to its space and time complexity, search time efficiency, and False Positive Rate (FPR). These metrics provide insights into the performance and effectiveness of the MI3SE scheme in securely searching encrypted data while maintaining low computational overhead and minimizing false positive results.

6.1.1 Space Complexity

In terms of the storage burden metric, MI3SE incurs additional storage requirements compared to traditional schemes that search on ciphertext rather than plaintext. In MI3SE, we maintain a list of all keywords that are encrypted, and for each keyword, there is a corresponding list that keeps track of the presence of that keyword in each document. Let N_k denote the number of unique keywords in the dataset, and N_d represent the total number of documents. Mathematically, the storage burden can be expressed as the sum of the storage required for storing encrypted keywords and the storage required for maintaining the document lists for each keyword. This can be represented as:

$$StorageBurden = N_k \times SPC_k + N_k \times SPC_p + N_d \times SPC_{id} \quad (1)$$

where SPC_k, SPC_p, and SPC_{id} represent the storage required for storing encrypted keywords, pointers, and document identifiers, respectively. This additional storage overhead is necessary for enabling efficient keyword-based searches on encrypted data while maintaining data confidentiality and ensuring accurate search results.

6.1.2 Time Complexity

In Our SSE scheme, two PPT algorithms SearchTokenGen and Query are used to fetch the requested documents from the corresponding server storage. In this scheme, the SearchTokenGen algorithm is based on the query keywords. This algorithm returns the token related to the query encrypted keywords sent by users. The time complexity of SearchTokenGen algorithm is $O(|\mathrm{W}_D| + |\mathrm{D}|)$ where $|\mathrm{W}_D|$ and $|\mathrm{D}|$ refer to the number of total keywords and documents, respectively. It means this algorithm is linear in both the number of total keywords and documents. This algorithm is also influenced by the number of query keywords created by the user. In this algorithm, the number of query keywords has a slight impact on computation time whereas the impact of this value on communication time is slightly more obvious. The space storage consumption is also influenced by this value such that increasing the number of query keywords leads to increasing space consumption. Therefore, it needs to adjust an adequate value for the number of query keywords.

Table 3: Comparison of Search and Communication Complexity

Scheme	Search Complexity	Communication Complexity
[21]	$O(\|\mathrm{W}_D\|^2 \ \mathrm{Log}^2 \ \|\mathrm{D}\|)$	$O(\max\{\|\mathrm{I}_s\|, \|\mathrm{W}_D\|\} + \|\mathrm{D}\|)$
[27]	$O(\|\mathrm{WP}\|)$	$O(\lambda)$
[28]	$O(\|\mathrm{W}_D\| \ \mathrm{Log} \ \|\mathrm{W}_D\|)$	$O(\|\mathrm{W}_D\|^2)$
[29]	$O(\|\mathrm{I}_s\| \ \mathrm{Log}^2 \ \|\mathrm{WP}\|)$	$O(\|\mathrm{I}_s\| \ \mathrm{Log}^2 \ \|\mathrm{WP}\|)$
MI3SE	$O(\|\mathrm{W}_D\| + \|\mathrm{D}\|)$	$O(\|\mathrm{w}_s\| + \|\mathrm{T}_s\| + \|\mathrm{D}_s\|)$

The Query algorithm searches the encrypted documents related to the index/indexes in the generated token and returns the corresponding documents. The search complexity of the Query algorithm is $O(|\mathrm{D}|)$. It indicates that this algorithm is linear in the number of encrypted documents. This algorithm is also slightly influenced by the number of indexes in the token and the number of indexes is completely related to the number of query keywords. We can say the Query algorithm is influenced by the number of query keywords, like SearchTokenGen algorithm. Due to the logical AND operator employed in the SearchTokenGen algorithm, increasing the number of query keywords can lead to decreasing the number of indexes in the token, and vice versa. Table 3 provides a comparison with prior research works in terms of search and communication complexity. However, in the Query algorithm, the main goal is the accuracy of obtained documents as output. Here, we show the impact of the number of query keywords on the accuracy of our scheme. To this end, we evaluate our scheme under the different number of encrypted query keywords. The accuracy metric quantifies the proportion of accurate findings in relation to the total number of findings. The following equation is employed to calculate the overall accuracy.

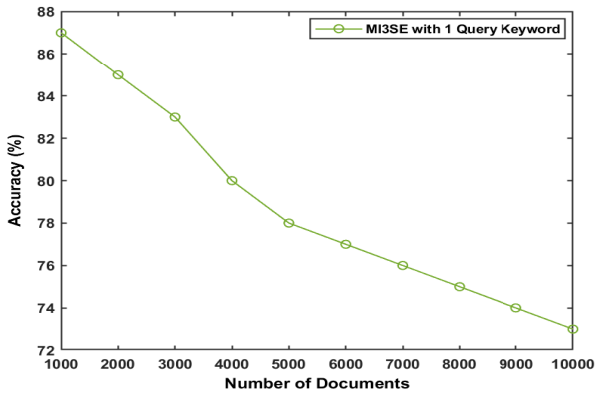
$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

where TP and TN refer to the number of documents properly fetched as related and unrelated documents whereas FP and FN are respectively the numbers of documents incorrectly fetched as related and unrelated documents.

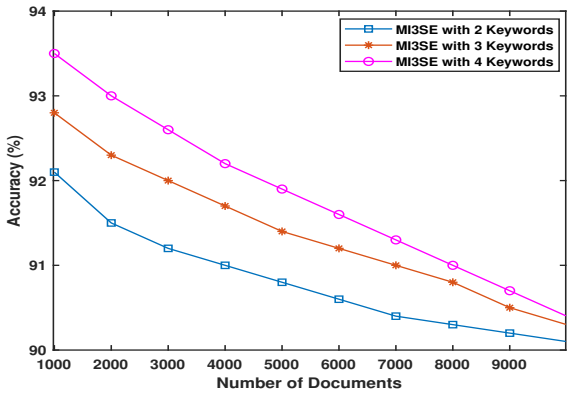
As shown in Figure 6, the multi-keyword search is more practical and accurate than the single-keyword search. In contrast, deploying the multi-keyword search will significantly increase the computation cost. Considering three factors computation time, communication time, and space storage consumption, in order to balance efficiency, we adopted the two-query keyword for our SSE scheme.

6.1.3 Search Time

In our evaluation, we examined the influence of varying numbers of keywords, ranging from 1 to 4, as well as the number of documents in the dataset, ranging from 1000 to 10000, on the search time. This comprehensive analysis allowed us to assess how changes in both the complexity of search queries and the



(a) Single-keyword search



(b) Multi-keywords search

Figure 6: Comparison the accuracy for our SSE scheme under different number of query keyword search = 1,2,3, and 4, when the number of documents $n = 10000$ and total keywords $|W_D| = 1500$.

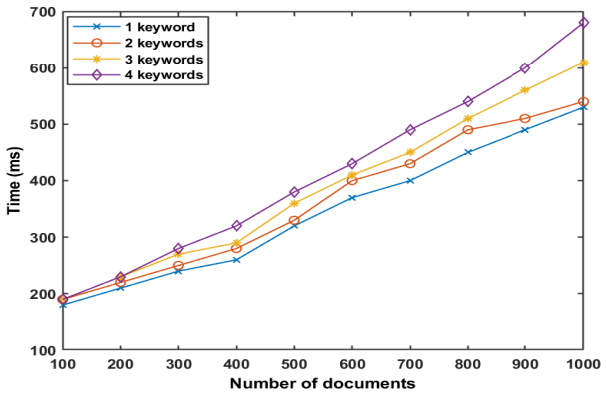


Figure 7: The process of retrieving a document from a device by a user via proxy server.

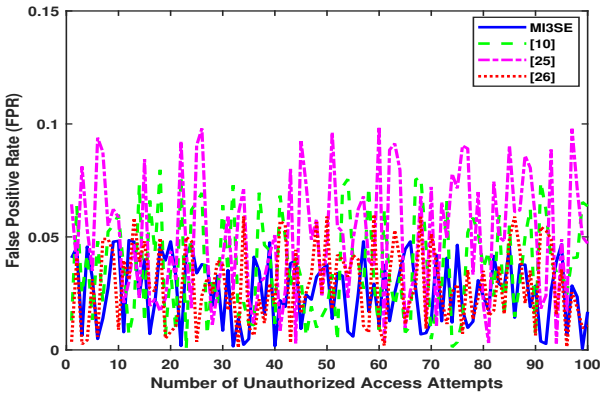


Figure 8: The Rate of Success of Unauthorized Users in Retrieving Documents.

size of the dataset affect the overall search performance. By systematically varying these parameters, we gained valuable insights into the scalability and efficiency of MI3SE under different search scenarios. As expected, the search time tends to increase as the number of documents in the dataset grows. Figure 7 illustrates this trend, where we observe a consistent rise in search time as the dataset size expands. Additionally, our analysis reveals that an increase in the number of keywords also correlates with higher search times. For instance, when considering a dataset containing 10000 documents, the search times for 1, 2, 3, and 4 keywords are recorded at 0.53 ms, 0.54 ms, 0.61 ms, and 0.68 ms, respectively. This observation underscores the impact of query complexity on search performance, highlighting the need for efficient indexing and retrieval mechanisms to handle larger datasets and more intricate search queries effectively.

As we can see in Figure 6 and Figure 7, while using a single keyword may lead to quicker search times, however it sacrifices accuracy. By incorporating two or more keywords, although the search time may increase slightly, the accuracy improves significantly. Therefore, considering both factors, MI3SE demonstrates optimal performance with two keywords. This balanced approach ensures efficient searches without compromising result accuracy, making MI3SE well-suited for practical applications where both speed and precision are essential.

6.1.4 *False Positive Rate (FPR)*

Figure 8 shows FPR across different schemes (MI3SE, [10], [25], and [26]), MI3SE exhibits a lower FPR compared to the other schemes, indicating its superior ability to minimize prevent unauthorized access attempts. A lower FPR suggests that MI3SE is more effective at accurately distinguishing between authorized and unauthorized access attempts. This is crucial for maintaining the security and integrity of data stored in IIoT devices, as a high FPR can result in unnecessary restrictions on legitimate users or an increased risk of unauthorized access. MI3SE achieves this lower FPR through its advanced security mechanisms and robust design. Unlike [10], [25], and [26], MI3SE utilizes a multi-user index-based security approach combined with a two-keyword query mechanism, enhancing its ability to accurately authenticate users and prevent false positives. This ensures that only authorized users are granted access to sensitive data, while unauthorized access attempts are effectively blocked. Furthermore, MI3SE's architecture and implementation are optimized to minimize false positives without compromising on security or performance. Its efficient handling of access requests and data retrieval operations contributes to its lower FPR, making it a reliable solution for securing data in IIoT environments. In summary, the lower FPR exhibited by MI3SE in Figure 8 its superiority in accurately identifying and preventing unauthorized access attempts, thereby ensuring the

confidentiality and integrity of data stored in IIoT devices.

7 Conclusion

In conclusion, we have presented a secure searchable symmetric encryption scheme, "MI3SE," which effectively addresses the security and privacy concerns associated with data exchanged among entities and stored in the outdoor IIoT devices and dedicated storage servers. The proposed SSE scheme, MI3SE, is built upon a two-keyword query approach. This scheme keeps secure and private, *data-at-rest*, the data stored in the device's storage. It has been proved that MI3SE is secure and robust against various security threats even if the device is physically stolen. Furthermore, we have compared our scheme with some benchmarks in terms of accuracy, and search time and the obtained results indicated the superiority of MI3SE. As a future plan, we are going to improve our work by integrating a trust model to this work. The trust model guarantees the trustworthiness and reliability of data measured by indoor/outdoor smart IoT devices.

Acknowledgment

This work was supported in part by the U.K. Engineering and Physical Sciences Research Council under Grant EP/ P03456X /1 and EP/X013162/1.

References

[1] S. A. Soleymani, S. Goudarzi, M. H. Anisi, H. Cruickshank, A. Jindal, and N. Kama, "TRUTH: Trust and authentication scheme in 5G-IIoT," *IEEE Transactions on Industrial Informatics*, 2022.

[2] C. Xenofontos, I. Zografopoulos, C. Konstantinou, A. Jolfaei, M. K. Khan, and K.-K. R. Choo, "Consumer, commercial, and industrial IoT (In) security: Attack taxonomy and case studies," *IEEE Internet of Things Journal*, vol. 9, no. 1, pp. 199–221, 2021.

[3] M. Wazid, A. K. Das, and S. Shetty, "BSFR-SH: Blockchain-enabled security framework against ransomware attacks for smart healthcare," *IEEE Transactions on Consumer Electronics*, vol. 69, no. 1, pp. 18–28, 2022.

[4] D. Javeed, M. S. Saeed, I. Ahmad, P. Kumar, A. Jolfaei, and M. Tahir, "An intelligent intrusion detection system for smart consumer electronics network," *IEEE Transactions on Consumer Electronics*, vol. 69, no. 4, pp. 906–913, 2023.

[5] Y. Liu, K.-F. Tsang, C. K. Wu, Y. Wei, H. Wang, and H. Zhu, "IEEE p2668-compliant multi-layer IoT-DDoS defense system using deep reinforcement learning," *IEEE Transactions on Consumer Electronics*, vol. 69, no. 1, pp. 49–64, 2022.

[6] M. Barbareschi, V. Casola, A. De Benedictis, E. La Montagna, and N. Mazzocca, "On the adoption of physically unclonable functions to secure IIoT devices," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 11, pp. 7781–7790, 2021.

[7] S. F. Tan and A. Samsudin, "Recent technologies, security countermeasure and ongoing challenges of industrial internet of things (IIoT): A survey," *Sensors*, vol. 21, no. 19, p. 6647, 2021.

[8] M. R. Ali, D. Pal, A. Das, and D. R. Chowdhury, "HARPOCRATES: An approach towards efficient encryption of data-at-rest," *IEEE Transactions on Emerging Topics in Computing*, 2024.

[9] R. Gupta, N. K. Jadav, H. Mankodiya, M. D. Alshehri, S. Tanwar, and R. Sharma, "Blockchain and onion-routing-based secure message exchange system for edge-enabled IIoT," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 2, pp. 1965–1976, 2022.

[10] G. S. Poh, P. Gope, and J. Ning, "Privhome: Privacy-preserving authenticated communication in smart home environment," *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 3, pp. 1095–1107, 2019.

[11] D. Cash, S. Jarecki, C. Jutla, H. Krawczyk, M.-C. Roşu, and M. Steiner, "Highly-scalable searchable symmetric encryption with support for boolean queries," in *Advances in Cryptology–CRYPTO 2013: 33rd Annual Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2013. Proceedings, Part I*. Springer, 2013, pp. 353–373.

[12] J. Bader and A. L. Michala, "Searchable encryption with access control in industrial internet of things (IIoT)," *Wireless Communications and Mobile Computing*, vol. 2021, pp. 1–10, 2021.

[13] D. X. Song, D. Wagner, and A. Perrig, "Practical techniques for searches on encrypted data," in *Proceeding 2000 IEEE Symposium on Security and Privacy. S&P 2000*. IEEE, 2000, pp. 44–55.

[14] K. Kurosawa and Y. Ohtaki, "UC-secure searchable symmetric encryption," in *International Conference on Financial Cryptography and Data Security*. Springer, 2012, pp. 285–298.

[15] Y. Ohtaki and K. Kurosawa, "How to update documents verifiably in searchable symmetric encryption," in *International Conference on Cryptology and Network Security*. Springer, 2013, pp. 309–328.

[16] D. Cash, S. Jarecki, C. Jutla, H. Krawczyk, M.-C. Roşu, and M. Steiner, "Highly-scalable searchable symmetric encryption with support for boolean queries," in *Annual cryptology conference*. Springer, 2013, pp. 353–373.

[17] S. Kamara and T. Moataz, "Boolean searchable symmetric encryption with worst-case sub-linear complexity," in *Annual International Conference on the Theory and Applications of Cryptographic Techniques*. Springer, 2017, pp. 94–124.

[18] R. Zhang, R. Xue, T. Yu, and L. Liu, "Dynamic and efficient private keyword search over inverted index–based encrypted data," *ACM Transactions on Internet Technology (TOIT)*, vol. 16, no. 3, pp. 1–20, 2016.

[19] L. Wu, B. Chen, K.-K. R. Choo, and D. He, "Efficient and secure searchable encryption protocol for cloud-based internet of things," *Journal of Parallel and Distributed Computing*, vol. 111, pp. 152–161, 2018.

[20] J. Li, Y. Huang, Y. Wei, S. Lv, Z. Liu, C. Dong, and W. Lou, "Searchable symmetric encryption with forward search privacy," *IEEE Transactions on Dependable and Secure Computing*, 2019.

[21] Z. Liu, Y. Huang, X. Song, B. Li, J. Li, Y. Yuan, and C. Dong, "Eurus: Towards an efficient searchable symmetric encryption with size pattern protection," *IEEE Transactions on Dependable and Secure Computing*, 2020.

[22] S. A. Soleymani, S. Goudarzi, M. H. Anisi, Z. Movahedi, A. Jindal, and N. Kama, "PACMAN: Privacy-preserving authentication scheme for managing cybertwin-based 6G networking," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 7, pp. 4902–4911, 2021.

[23] Y. Liu, J. Yu, J. Fan, P. Vijayakumar, and V. Chang, "Achieving privacy-preserving dsse for intelligent iot healthcare system," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 3, pp. 2010–2020, 2021.

[24] Q. Feng, D. He, M. Luo, X. Huang, and K.-K. R. Choo, "Eprice: An efficient and privacy-preserving real-time incentive system for crowdsensing in industrial internet of things," *IEEE Transactions on Computers*, 2023.

[25] W. Wang, P. Xu, D. Liu, L. T. Yang, and Z. Yan, "Lightweighted secure searching over public-key ciphertexts for edge-cloud-assisted industrial iot devices," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 6, pp. 4221–4230, 2019.

[26] H. Yin, W. Zhang, H. Deng, Z. Qin, and K. Li, "An attribute-based searchable encryption scheme for cloud-assisted IIoT," *IEEE Internet of Things Journal*, 2023.

[27] T. Hoang, A. A. Yavuz, F. B. Durak, and J. Guajardo, "Oblivious dynamic searchable encryption on distributed cloud systems," in *IFIP Annual Conference on Data and Applications Security and Privacy*. Springer, 2018, pp. 113–130.

[28] R. Zhang, R. Xue, T. Yu, and L. Liu, "Dynamic and efficient private keyword search over inverted index–based encrypted data," *ACM Transactions on Internet Technology (TOIT)*, vol. 16, no. 3, pp. 1–20, 2016.

[29] J. Ghareh Chamani, D. Papadopoulos, C. Papamanthou, and R. Jalili, "New constructions for forward and backward private symmetric searchable encryption," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, 2018, pp. 1038–1055.