Hindawi
Security and Communication Networks
Volume 2024, Article ID 2764345, 11 pages
https://doi.org/10.1155/2024/2764345

Research Article

Exploring the Security Vulnerability in Frequency-Hiding Order-Preserving Encryption

JiHye Yang and Kee Sung Kim

Department of Computer Software, Daegu Catholic University, Daegu, Republic of Korea

Correspondence should be addressed to Kee Sung Kim; kee21@cu.ac.kr

Received 17 August 2023; Revised 24 January 2024; Accepted 21 February 2024; Published 29 February 2024

Academic Editor: Tom Chen

Frequency-hiding order-preserving encryption (FH-OPE) has emerged as an important tool in data security, particularly in cloud computing, because of its unique ability to preserve the order of plaintexts in their corresponding ciphertexts and enable efficient range queries on encrypted data. Despite its strong security model, indistinguishability under frequency analyzing ordered chosen plaintext attack (IND-FA-OCPA), our research identifies a vulnerability in its design, particularly the impact of range queries. In our research, we quantify the frequency of data exposure resulting from these range queries and present potential inference attacks on the FH-OPE scheme. Our findings are substantiated through experiments on real-world datasets, with the goal of measuring the frequency of data exposure resulting from range queries on FH-OPE encrypted databases. These results quantify the level of risk in practical applications of FH-OPE and reveal the potential for additional inference attacks and the urgency of addressing these threats. Consequently, our research highlights the need for a more comprehensive security model that considers the potential risks associated with range queries and underscores the importance of developing new range-query methods that prevent exposing these vulnerabilities.

1. Introduction

The significant rise in cloud computing used for data storage and processing necessitates new encryption methodologies to ensure data security. One such method is frequency-hiding order-preserving encryption (FH-OPE), which maintains the order of plaintexts in the ciphertexts, thereby allowing efficient range queries and sorting operations on the encrypted data. A significant aspect of FH-OPE is frequency hiding, a mechanism that conceals the frequency of individual values in the encrypted data, thereby mitigating some vulnerabilities of standard OPE. Such frequency hiding is important in a scenario where data usage patterns should remain confidential, like in medical database or financial records where repeated values could hint at certain medical conditions or repeated transactions, respectively.

While the security model for FH-OPE, known as indistinguishability under frequency analyzing ordered chosen plaintext attack (IND-FA-OCPA), is recognized as robust and well-structured, our work suggests that it might be incomplete. More specifically, the model was designed without considering that range queries could weaken FH-OPE's ability to hide plaintext distributions. We demonstrate in this paper that the security of FH-OPE can be affected not only by attackers directly performing these queries but also simply by observing their results, leading to potential frequency exposure.

We then transition our focus to quantifying frequency exposure, determining the number of range queries needed to reveal the frequency of all plaintexts in FH-OPE and the expected number of distinct ciphertexts exposed after executing a specific number of range queries. Our approach employs mathematical problems such as the coupon collector's problem and probabilistic analyses to illustrate these concepts effectively. We extend our research to other potential threats posed by more complex queries, such as join queries, as well as inference attacks conducted via association rule mining. Our findings demonstrate that these methods can use and potentially exacerbate the identified

weaknesses in the FH-OPE scheme. To support our theoretical analysis, we experiment with real-world datasets. The focus of these experiments was to measure the frequency of data exposure when executing range queries on an FH-OPE encrypted database. The results obtained quantified the risk associated with practical implementations of FH-OPE and reveal the possibility for further inference attacks.

Consequently, our study underlines the need for a more comprehensive FH-OPE security model. This model must consider the risks associated with range queries and their potential to expose vulnerabilities. Our research presents the importance of developing new methods for conducting range queries on encrypted data that are devoid of the vulnerabilities currently identified.

1.1. Related Work

1.1.1. OPE and FH-OPE. Over the years, numerous studies have been conducted on OPE [1–15], and FH-OPE [16–20]. The concept of OPE has garnered substantial attention due to its utility in database systems [21–23]. In recent research, a shift in focus has been noticeable, with most efforts centered on hiding frequency information and achieving ideal security for OPE and FH-OPE. The initial exploration of OPE was largely influenced by the work of Boldyreva et al. [2], which established the first formal definitions and security models for OPE and introduced the concept of indistinguishability under ordered chosen plaintext attack (IND-OCPA). This model primarily aimed to preserve the order of plaintext data, even in its encrypted form. As the field evolved, potential vulnerabilities, particularly concerning frequency exposure in traditional OPE schemes, became more apparent. Frequency-hiding order-preserving encryption was proposed in response. Kerschbaum's influential paper [16] marked a significant advance in this direction. Kerschbaum introduced the IND-FA-OCPA security model specifically designed for FH-OPE, aiming to mitigate frequency-related vulnerabilities and enhance data security. However, subsequent research by Maffei et al. [17] highlighted certain inadequacies in Kerschbaum's original security model. They presented a comprehensive critique of the model's structure and identified and executed an attack on it, thereby exposing an issue in the original security proof. Moreover, they proposed an impossibility result, demonstrating that Kerschbaum's security definition could not be achieved by any OPE scheme. Consequently, they introduced a new security definition that maintains the fundamental concept of frequency-hiding yet is practically achievable. They also demonstrated that their refined version of the security definition, which more accurately captured the concept of frequency-hiding, could indeed be realized.

1.1.2. Attacks on OPE and FH-OPE. Several papers [24–28] have presented various attack models targeting both OPE and FH-OPE. In general, deterministic OPE leaks both the order and the frequency of plaintexts. Such leakages lead to two primary types of attacks that are based on frequency: sorting attacks and frequency-revealing attacks.

(1) Sorting Attack. This attack, presented in [24], targets columns densely encrypted with OPE, such as those for age and disease severity, where each distinct plaintext is encrypted at least once. In such scenarios, there is a one-to-one correspondence between the distinct OPE ciphertexts and the distinct plaintexts. An adversary can exploit this to reconstruct the plaintexts by mapping the ordered distinct ciphertexts to their corresponding ordered distinct plaintexts. Therefore, the success of this attack is up to the adversary's prior knowledge of the plaintext distribution. Notably, the sorting attack only succeeds in columns with high density, where every element of the plaintext space is present; otherwise, it is likely to fail.

(2) Frequency-Revealing Attacks. This attack, presented in [24, 26], is specifically effective against datasets with a low-density plaintext space. In [24], their approach to frequency analysis is termed a cumulative attack. For columns encrypted with OPE, the adversary can infer the frequency of each ciphertext by constructing a histogram that reflects the pattern of the encrypted data. The adversary then exploits frequency and order leakages to correlate ciphertexts with plaintexts, aiming to closely match their distributions. The success of this cumulative attack hinges on the adversary having prior knowledge of the plaintext distribution with public auxiliary information. In addition, in [28], they introduced three novel ciphertext-only attacks for FH-OPE schemes [11, 16, 20]. For conducting frequency analysis, they assume that the adversary exploits leakages from the distribution of ciphertexts and the orders in which they are inserted. Specially, they presented that they recovered about 96% of plaintext frequencies for Kerschbaum's FH-OPE scheme in a nonuniform ciphertext distribution environment. In addition, they conducted a plaintext frequency attack on [11, 20] under the assumption that the attacker is aware of the ciphertext input order.

As shown in Table 1, the attacks initially presented by [24] considered only deterministic ciphertexts, focusing on the vulnerabilities inherent to OPE schemes. While effective, these attacks are limited to scenarios where plaintext distribution is known, which is a significant constraint. On the other hand, Cao et al. [28] expanded the scope by targeting FH-OPE schemes. Despite their progress, their methods had an applicability of "No," indicating that they were only applicable to specific schemes. Table 1 also shows that their work focused on schemes satisfying the IND-FA-OCPA security definition presented by [16]. However, Maffei et al. demonstrated its inadequacies, proving that the original IND-FA-OCPA could not be achieved in any FH-OPE scheme. In contrast, our attacks are conducted on the IND-FA-OCPA* (throughout this paper, "IND-FA-OCPA*" will be denoted simply as "IND-FA-OCPA"), a revised security model proposed by [17]. Despite these ongoing advancements and varied attacks, no research has yet shown that an adversary can conduct frequency exposure analysis without the need for additional information or constraints. Therefore, our attacks utilize only information that naturally occurs in OPE-encrypted databases. Moreover, current research noticeably lacks in exploring vulnerabilities in the

Table 1: Comparison of the proposed method with other competing methods.

Method	Target	Attack	Applicability	Security	Assumptions	
					Density	Auxiliary information
[24]	OPE	Sorting Frequency-revealing	Yes	IND-OCPA	High Low	Distribution of plaintexts
[28]	FH-OPE	Frequency-revealing	No	IND-FA-OCPA	—	Distribution of nonuniform ciphertexts Insertion order of ciphertexts
Ours	FH-OPE	Frequency-revealing	Yes	IND-FA-OCPA*	—	—

Applicability indicates whether the attack method can be applied across various encryption schemes. "Yes" signifies that the method is general and does not require knowledge of the specific encryption scheme, making it applicable to any OPE or FH-OPE system. "No" signifies that the method is scheme-specific, designed with full knowledge of a particular scheme's details, and can only be applied to that specific system. In the security, IND-FA-OCPA* denotes revised IND-FA-OCPA, addressing its original limitations. A high density means that all data in the plaintext space are encrypted. Conversely, low density implies that only a subset of the possible values in the plaintext space is encrypted. Auxiliary Information means the supplemental knowledge an adversary requires to successfully execute an attack.

FH-OPE security model. Our study aims to address this gap by highlighting issues within the FH-OPE security model, based on effectiveness of our attacks.

1.2. Our Contributions. Our research significantly enhances the current understanding of security vulnerabilities in FH-OPE, contributing to the existing field in several ways:

(1) We revisited the IND-FA-OCPA security model and identified significant vulnerabilities associated with range queries. This discovery questions the reliability of IND-FA-OCPA in encrypted systems, indicating a need to re-evaluate its design, especially in the context of range queries.

(2) We quantified the frequency of plaintext exposure with respect to the number of range queries executed on FH-OPE. This quantification offers insight into the risks associated with standard range queries.

(3) To validate our quantification formula, we conducted experiments with range queries on two real-world datasets, measuring the actual frequency of plaintext exposure. Furthermore, our research included the execution of an inference attack on the FH-OPE scheme.

1.3. Setting and Notations. The attacks we planned to execute are all based on a common framework, which is detailed below, accompanied by the necessary notation. Let n be the number of plaintexts to be encrypted and N be the number of distinct plaintexts. Let $X = \{x_1, x_2, \ldots, x_n\}$ denote the entire set of plaintexts, where $|X| = n$. Within this set, several plaintexts may be repeated. We then define another set $X' = \{x_1', x_2', \ldots, x_N'\}$ which includes only the distinct plaintexts from set X, where $|X'| = N$.

1.3.1. Range Queries. Suppose we have a plaintext range query $R(x_l, x_u)$ which requests all records in the range $x_l \le x < x_u$. In an FH-OPE encrypted database, the client transforms this query into an encrypted range query $R'(E(x_l), E(x_u))$ as follows:

(1) The client encrypts the range endpoints x_l and x_u with FH-OPE encryption function. Due to FH-OPE's frequency-hiding property, if x_l and x_u are duplicates in X, for each, this could produce multiple corresponding ciphertexts.

(2) The client selects the smallest encrypted value from the set of ciphertexts corresponding to x_l and the smallest encrypted value from the set of ciphertexts corresponding to x_u.

(3) The client makes the encrypted range query $R'(E_{\min}(x_l), E_{\min}(x_u))$, which requests all records in the range $E_{\min}(x_l) \le y < E_{\min}(x_u)$, where y is the ciphertext in the encrypted database.

(4) The client sends $R'(E_{\min}(x_l), E_{\min}(x_u))$ to the server.

(5) The server executes $R'(E_{\min}(x_l), E_{\min}(x_u))$ on the encrypted database and returns the result set of $R'(E_{\min}(x_l), E_{\min}(x_u))$ to the client.

(6) The client decrypts the result set to obtain the plaintext values that satisfy the original plaintext range query $R(x_l, x_u)$.

We consider the application of FH-OPE to an annual salary database. Given salary dataset $X = \{8000, 10000, 10000, 15000, 17000, 17000\}$ with its corresponding ciphertexts $Y = \{10, 20, 30, 40, 50, 60\}$ stored on the server, the server can execute queries generated by a client. For instance, we suppose a client is interested in determining the count of salaries falling within a specific range greater than or equal to 10000 but less than 17000. The client would generate a range query reflecting this interest but in the form of ciphertexts, such as $R'(20, 50)$, and send it to the server. Consequently, the ciphertext set, which may be utilized for a range search, has a size of N. This approach ensures that the plaintext's frequency remains concealed during the range query because the query leverages the minimum ciphertext corresponding to the lower range bound. Consequently, the server cannot ascertain the frequency of a particular salary in the data.

1.3.2. Adversarial Model. We adopt an adversarial model characterized by a persistent, passive adversary including the server acting as an honest-but-curious adversary. This adversary can continuously observe all interactions between the client and the server. Notably, our adversary model, unlike a snapshot adversary with only single instance access to the server's memory, continuously monitors the range queries executed by the client, identifying patterns and extracting meaningful information from these activities.

1.3.3. Mathematical Notation. In our notation, we will always use log to denote the natural rather than base 2 logarithm. The $n-$th harmonic number will be represented as H_n, such that $H_n = \sum_{k=1}^{n} 1/k$.

2. Preliminaries

In this section, we formally introduce the concept of OPE and explain its core security notions, specifically IND-OCPA and IND-FA-OCPA. These foundational definitions pave the way for our following discussion on the possible threats linked with OPE in the subsequent sections.

2.1. Formal Notion of OPE. A stateful OPE is a technique that keeps a record of past operations, which is essential for improving security. The unique adaptive nature of a stateful OPE provides the necessary layer of complexity for achieving IND-OCPA security.

Definition 1 (OPE). A stateful OPE scheme consists of the following three algorithms (K, E, D):

(1) $K(1^\lambda) \longrightarrow S$: The key generation algorithm takes as input a security parameter λ and initializes a state S.

(2) $E(x, S) \longrightarrow (y, S')$: The encryption algorithm takes as input a plaintext x and a state S. It outputs a ciphertext y and updates the state S to S'.

(3) $D(y, S) \longrightarrow x$: The decryption algorithm takes as input a ciphertext y and a state S. It outputs a plaintext x.

Definition 2 (order-preserving). An OPE scheme is order-preserving if it maintains the order of plaintexts in their corresponding ciphertexts. This is for any two plaintexts x_1 and x_2 and their corresponding ciphertexts y_1 and y_2 produced by the OPE scheme, if $y_1 \geq y_2$, then $x_1 \geq x_2$.

2.2. Security Definitions. IND-OCPA security means that no efficient (bounded by polynomial time) adversary can distinguish between the ciphertexts of two sequences of plaintext that are equally ordered. This concept is illustrated through a simulation. The simulation $\mathrm{SIM}^{\lambda}_{\mathrm{IND-OCPA}}$ between adversary $\mathscr{A}$ and simulator $\mathscr{S}$ for security parameter λ proceeds as follows:

(1) The adversary $\mathscr{A}$ prepares two plaintext sequences $X_0 = \{x_{1,0}, \ldots, x_{n,0}\}$ and $X_1 = \{x_{1,1}, \ldots, x_{1,n}\}$ where $x_{i,0} < x_{j,0} \Longleftrightarrow x_{i,1} < x_{j,1}$, $1 \leq i, j \leq n$, and sends them to the simulator $\mathscr{S}$.

(2) The simulator $\mathscr{S}$ randomly chooses $b \longleftarrow \{0, 1\}$, executes $K(1^\lambda)$, and runs $(y_{i,b}, S_i) \longleftarrow (Ex_{i,b}, S_{i-1})$, for all $1 \leq i \leq n$. Then, the simulator $\mathscr{S}$ sends $y_{1 \leq i \leq n, b}$ to the adversary $\mathscr{A}$.

(3) The adversary $\mathscr{A}$ tries to infer which sequence has been encrypted and outputs b' as their guess for b.

Definition 3 (IND-OCPA). An OPE scheme has IND-OCPA security if the chance of a probabilistic polynomial time (PPT) adversary $\mathscr{A}$ correctly guessing whether a given ciphertext corresponds to a particular plaintext sequence is negligibly better than random guessing. Otherwise stated, the probability of $b' = b$ is $1/2 + \mathrm{negl}(\lambda)$, where $\mathrm{negl}(\lambda)$ is a negligible function in the security parameter λ.

Definition 4 (randomized order). Let us consider a sequence of not necessarily distinct plaintexts $X = \{x_1, \ldots, x_n\}$. We define a randomized order $\Gamma = \{\gamma_1, \gamma_2, \ldots, \gamma_n\}$, representing one of the possible permutation of the set $\{1, 2, \ldots, n\}$ that maintains the sequence of X. This means for every pair of indices i and j where $i \neq j$, $\gamma_i \neq \gamma_j$, and

$$\forall i, j. \left(x_i > x_j \Longrightarrow \gamma_i > \gamma_j\right) \wedge \left(\gamma_i > \gamma_j \Longrightarrow x_i \geq x_j\right). \quad (1)$$

For instance, we consider a plaintext sequence $X = \{2, 2, 1, 1\}$. It could be represented by the randomized orders $\Gamma_1 = \{4, 3, 2, 1\}$, $\Gamma_2 = \{3, 4, 2, 1\}$, $\Gamma_3 = \{4, 3, 1, 2\}$, or $\Gamma_4 = \{3, 4, 1, 2\}$. In this framework, the common randomized order Γ for X_0 and X_1 denotes the elements shared between the two randomized order sets of X_0 and X_1. So, for $X_0 = \{3, 3, 3, 2\}$ and $X_1 = \{3, 2, 2, 1\}$, the common randomized order Γ could be either $\{4, 3, 2, 1\}$ or $\{4, 2, 3, 1\}$. We utilize the notation $\Gamma_{\downarrow_i}$ to represent the order of the elements in the sequence up to γ_i. For example, if we consider the sequence $\Gamma = \{2, 4, 3, 1\}$, $\Gamma_{\downarrow_3}$ would refer to the order of the first three elements of Γ, resulting in the sequence $\{1, 3, 2\}$.

To satisfy the IND-FA-OCPA security notion and protect against frequency analysis attacks, Maffei et al. proposed an enhanced encryption method known as augmented order-preserving encryption. Therefore, we employ the augmented OPE scheme proposed by [17].

Definition 5 ((augmented) OPE). An augmented OPE scheme consists of the following three algorithms (K, E, D):

(i) $K(1^\lambda) \longrightarrow S$: The key generation algorithm takes as input a security parameter λ and initializes a state S.

(ii) $E(x, S, \Gamma) \longrightarrow (y, S')$: The encryption algorithm takes as input a plaintext x, a state S, and an order Γ. It outputs a ciphertext y and updates the state S to S'.

(iii) $D(y, S) \longrightarrow x$: The decryption algorithm takes as input a ciphertext y and a state S. It outputs a plaintext x.

In the context of FH-OPE, IND-FA-OCPA security extends the IND-OCPA definitions to withstand frequency analysis attacks. This is defined using the simulation $\mathrm{SIM}^{\lambda}_{\mathrm{IND-FA-OCPA}}$, where an adversary $\mathscr{A}$ interacts with a simulator $\mathscr{S}$. The simulation $\mathrm{SIM}^{\lambda}_{\mathrm{IND-FA-OCPA}}$ for security parameter λ proceeds as follows:

(1) The adversary $\mathscr{A}$ prepares two plaintext sequences $X_0 = \{x_{1,0}, \ldots, x_{n,0}\}$ and $X_1 = \{x_{1,1}, \ldots, x_{1,n}\}$. These sequences have at least one common randomized order Γ. These are sent to the simulator $\mathscr{S}$.

(2) The simulator $\mathscr{S}$ randomly chooses $b \longleftarrow \{0,1\}$ and one of Γ, executes $K(1^{\lambda})$, and runs $(y_{i,b}, S_i) \longleftarrow E(x_{i,b}, S_{i-1}, \Gamma_{\downarrow i})$, for all $1 \le i \le n$, based on the selected Γ. Then, the simulator $\mathscr{S}$ sends $y_{1 \le i \le n, b}$ to the adversary $\mathscr{A}$.

(3) The adversary $\mathscr{A}$ tries to infer which sequence has been encrypted and outputs b' as their guess of b.

Definition 6 (IND-FA-OCPA). An (augmented) OPE scheme has IND-FA-OCPA security if the chance of a probabilistic polynomial time (PPT) adversary $\mathscr{A}$ correctly guessing whether a given ciphertext corresponds to a particular plaintext sequence is only negligibly better than random guessing. Differently expressed, the probability of $b' = b$ is $1/2 + \mathrm{negl}(\lambda)$, where $\mathrm{negl}(\lambda)$ is a negligible function in the security parameter λ.

3. Exposing Vulnerability in the Security Model

In this section, we expose a potential vulnerability in the IND-FA-OCPA security model. This vulnerability arises when an adversary $\mathscr{A}$ merely observes the outcomes of range queries. Although these results are typically accessible to any system user and do not grant $\mathscr{A}$ any additional advantage, this seemingly harmless activity could still potentially weaken the security of IND-FA-OCPA.

Let X_0 and X_1 be two plaintext sequences. D_0 and D_1 are defined as sequences that represent the frequency of each unique plaintext in X_0 and X_1, respectively, when the unique plaintexts are sorted in ascending order. More formally, if U_i denotes the ith unique plaintext in ascending order in a sequence and $f_{X_j}(U_i)$ denotes the frequency of U_i in sequence X_j, then $D_j = \left\{f_{X_j}(U_1), f_{X_j}(U_2), \ldots, f_{X_j}(U_n)\right\}$ for $j \in \{0, 1\}$. We use n_j to denote $|D_j|$ and denote n^* as the greater value between n_0 and n_1. When D_0 and D_1 have different frequencies at some index t, we define $f_{X_j}(U_t)$ as the frequency of U_t in D_j for $t \in \{1, 2, \ldots, n_j\}$.

Theorem 7. *Given two plaintext sequences X_0 and X_1, which share a common randomized order Γ and have their frequency distributions D_0 and D_1, respectively, an adversary $\mathscr{A}$ that can observe the outcomes of range queries can distinguish which sequence has been selected by the simulator $\mathscr{S}$ with a probability of at least $p_f = 1 - (1 - 1/n^*)^{2k}$ after k queries have been processed, where the frequency distributions $D_0 \neq D_1$.*

Proof. Consider an adversary $\mathscr{A}$ that observes the outcomes of the range queries made by some entity (not necessarily $\mathscr{A}$). For each range query $R'(E(x_l), E(x_u))$, where $E(x_l)$ and $E(x_u)$ denote the lower and upper bounds of the encrypted range, respectively, there is a chance that either $E(x_l)$ or $E(x_u)$ matches an encrypted plaintext corresponding to $f_{X_j}(U_t)$ in the frequency distribution D_j. If the ciphertext corresponding to $f_{X_j}(U_t)$ is included in the lower and upper bounds of the range query, then the number of ciphertexts returned as a result of the range query is different for each of the sequences X_0 and X_1. The adversary $\mathscr{A}$ can distinguish the sequence X_b selected by the simulator by observing this difference in the number of returned ciphertexts. The probability p_f of $\mathscr{A}$ being able to distinguish X_0 from X_1 after observing k queries can be computed as follows: The probability of a range query not including the ciphertext corresponding to $f_{X_j}(U_t)$ in either of the two selected values in a query is given by $p = 1 - 1/n^*$. Therefore, the probability of the ciphertext not being included in any of the $2k$ selections is $p_{2k} = p^{2k} = (1 - 1/n^*)^{2k}$. The probability p_f of the ciphertext being included in at least one of the $2k$ selections is then given by $p_f = 1 - p_{2k} = 1 - (1 - 1/n^*)^{2k}$. Hence, with k queries (equivalent to $2k$ selections), $\mathscr{A}$ can distinguish between the sequences with probability p_f.

Theorem 7 is illustrated with a practical example. Suppose we have two plaintext sequences: $X_0 = \{1, 1, 2, 2\}$ and $X_1 = \{1, 2, 2, 3\}$. The frequency distributions of X_0 and X_1 are represented as $D_0 = \{2, 2\}$ and $D_1 = \{1, 2, 1\}$, respectively. Here, the frequencies of the distinct plaintext "1" are different in X_0 and X_1, i.e., $f_{X_0}(1) = 2$ and $f_{X_1}(1) = 1$, so we consider the index $t = 1$ where D_0 and D_1 differ. These sequences share a common randomized order $\Gamma = \{1, 2, 3, 4\}$. We assume that the simulator $\mathscr{S}$ selects the sequence X_0 for encryption, resulting in the ciphertext sequence $Y = \{10, 20, 30, 40\}$. When an entity issues a range query that includes the ciphertext corresponding to plaintext "1" as either the upper or lower bound, the adversary $\mathscr{A}$, who can observe the results of such queries, will see two ciphertexts being returned. From this observation, $\mathscr{A}$ can infer that plaintext "1" appears twice in the selected sequence. By comparing this frequency with the frequency distributions D_0 and D_1, $\mathscr{A}$ can correctly conclude that X_0 is the sequence encrypted by the simulator is X_0. Thus, the adversary successfully distinguishes the chosen plaintext sequence. □

Remark 8. This theorem underscores a key limitation of the IND-FA-OCPA security model. While it provides security against adversaries attempting to derive information from individual plaintext-ciphertext pairs, it does not fully protect the frequency patterns of these pairs. Consequently, it does not ensure frequency hiding. An adversary, just by observing the outcomes of range queries, could potentially infer the frequency of specific plaintexts within a given

range. This vulnerability highlights that the IND-FA-OCPA security model may still be susceptible to frequency analysis attacks.

4. Quantifying Frequency Exposure

We now shift focus to quantifying frequency exposure. The objective of this section is not merely to measure the degree of frequency exposure of plaintexts in an FH-OPE scheme through range queries but to highlight the urgent need for further research into novel range-query methods that could mitigate this exposure.

4.1. Demonstrating Frequency Exposure Using the Coupon Collector's Problem. Due to its complexity, quantifying the level of frequency exposure in FH-OPE is a challenge. However, we can leverage mathematical problems, such as the coupon collector's problem, to measure this exposure. This problem, where the goal is to collect distinct coupons through independent trials, parallels the process of conducting range queries on an encrypted database. In both scenarios, the aim is to acquire unique elements (or data) from a larger set. A mathematical model, inspired by the coupon collector's problem that estimates the number of range queries required to expose the frequency of all plaintexts in an FH-OPE scheme, is demonstrated below.

Theorem 9. *To reveal the frequency of all plaintexts in our scenario with FH-OPE, the required number of range queries, denoted as Q, is given by the approximation:*

$$Q \approx \frac{N \log N}{2}. \tag{2}$$

Proof. In the coupon collector's problem, the expected number of trials for collecting all N distinct coupons is given by the formula:

$$\mathbb{E}[N] = N \times H_N = N \log N, \tag{3}$$

where H_N represents the $N-$th harmonic number, roughly equal to $\log N$.

Comparing our scenario with FH-OPE to the coupon collector's problem, we consider each distinct ciphertext in $E(X^{'})$ as a unique coupon and our range queries as trials to collect these coupons. A range query, denoted as $R^{'}(E_{\min}(x_l), E_{\max}(x_u))$, parallels the action of selecting two coupons simultaneously.

Therefore, the required number of range queries, denoted as Q, is approximated by $N \log N/2$. □

4.2. Probabilistic Analysis of Plaintext Frequency Exposure. In the preceding subsection, we examined the frequency exposure of plaintext through a method similar to the coupon collector's problem. Now, we shift our perspective to a probabilistic analysis of the frequency exposure of plaintext, focusing on the expected number of unique plaintexts revealed after executing a specific number of range queries.

Theorem 10. *The expected number of distinct ciphertexts selected after k queries (or 2k selections) is given by the equation:*

$$\mathbb{E}[2k] = N \times \left(1 - \left(1 - \frac{1}{N}\right)^{2k}\right). \tag{4}$$

Proof. We begin by noting that the probability of not selecting a particular ciphertext in a selection can be represented as

$$p = \left(1 - \frac{1}{N}\right). \tag{5}$$

Then, the probability of not selecting a particular ciphertext in any of the $2k$ sections is

$$p_{2k} = p^{2k} = \left(1 - \frac{1}{N}\right)^{2k}. \tag{6}$$

Hence, the probability of selecting the ciphertext in at least one of the $2k$ selections is

$$p_f = 1 - p_{2k} = 1 - \left(1 - \frac{1}{N}\right)^{2k}. \tag{7}$$

Substituting this into the equation for the expected number of distinct ciphertexts selected after k queries (or $2k$ selections), we get our result:

$$\mathbb{E}[2k] = N \times p_f = N \times \left(1 - \left(1 - \frac{1}{N}\right)^{2k}\right). \tag{8}$$

Now, we consider an example where $N = 100$ and $k = 10$. The expected number of distinct ciphertexts selected after $k = 10$ range queries is $E[20] = 100 \times (1 - (1 - 1/100)^{20}) \approx 18.13$. These interpretations help us probabilistically quantify the vulnerability of plaintext frequencies in FH-OPE after executing a specific number of range queries. □

5. Experiments

To demonstrate the potential vulnerability within the context of FH-OPE in practical settings, we performed extensive range queries on real-world datasets. These experiments were conducted on a desktop computer with AMD Ryzen 7 PRO 4750G (3.60 GHz, 16 GB RAM) running Windows 11. All of the experimental code was written in Python 3.9.7.

5.1. Datasets and Preprocessing. For our experiments, we employed two real-world datasets: Dataset 1 and Dataset 2. Dataset 1, sourced from Allegheny County Employee Salaries (https://catalog.data.gov/dataset/allegheny-county-employee-salaries), contains salary information that is frequently encrypted in real-world scenarios to ensure privacy. This dataset has 6280 entries with 1677 distinct values,

i.e., $N = 1677$. On the other hand, Dataset 2 contains weight information of Women's National Basketball Association (WNBA) (https://www.kaggle.com/datasets/jinxbe/wnba-player-stats-2017) players. It consists of 143 data points with 40 distinct values, i.e., $N = 40$. We encrypted both datasets using the FH-OPE scheme [17] to preserve privacy, enabling us to measure the frequency of different data points without risking data leakage.

5.2. Experiment Setup. We designed our experiments to evaluate the FH-OPE scheme under two different query scenarios. The first scenario is characterized by a uniformly random range query, where all data are equally likely to be queried. In contrast, the second approach involves a weighted range-query model that assumes a Gaussian distribution, thereby giving preferential consideration to areas likely to be queried more frequently by users.

5.3. Experiment Results

5.3.1. Uniform Random Range Query. In the uniform random range-query scenario, we aimed to verify the theoretical results obtained in previous sections, with a particular focus on those associated with equation (2) and experimented on Dataset 1. Figure 1 illustrates the plaintext frequency exposure corresponding to a varying number of range queries. Q_n is defined as the number of range queries corresponding to different proportions of equation (2). Here, Q_1 corresponds to $0.05 \times (N \log N/2)$, Q_2 corresponds to $0.1 \times (N \log N/2)$, Q_3 corresponds to $0.2 \times (N \cdot N \log N/2)$, Q_4 corresponds to $0.5 \times (N \log N/2)$, and Q_5 corresponds to $(N \log N/2)$. As can be observed from Figure 1, with a number of queries equivalent to equation (2), we effectively reveal almost all plaintext frequencies. In addition, Figure 1 also shows the fact that executing fewer queries than the size of equation (2) still leads to a considerable degree of plaintext exposure.

5.3.2. Gaussian Range Query. To generate Gaussian range queries, we utilized the properties of Dataset 2, which naturally follows a Gaussian distribution. The mean and standard deviation of Dataset 2 are approximately 79.02 and 10.96, respectively. Utilizing these parameters, we generated synthetic data through random sampling from a Gaussian distribution. This procedure yielded a set of representative data points that could be used to execute a range of queries reflecting the natural distribution of the dataset. By applying equation (4) to the Gaussian range query, we calculated the expected number of distinct ciphertexts selected as a result of the chosen k.

From Figure 2, it is clear that even with a smaller number of queries, a significant degree of plaintext exposure is possible. Also, despite the differences in query distribution between uniform random range queries and Gaussian range queries, the outcomes do not exhibit a significant disparity. Our results underline the need for improved methods of query processing in FH-OPE to minimize such data exposure and enhance its overall security.

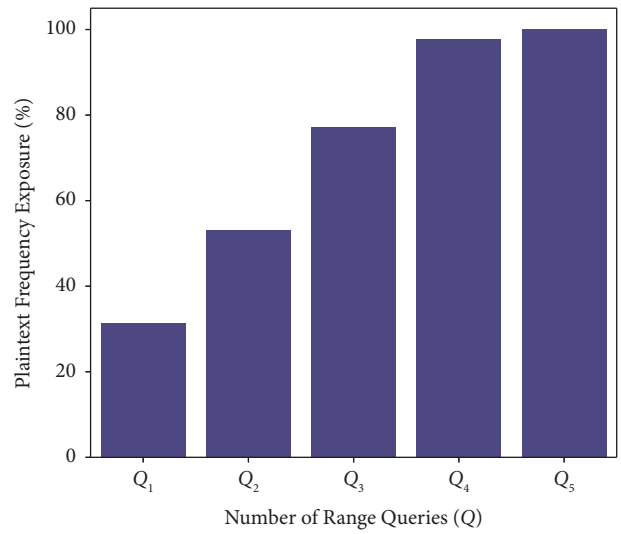

Figure 1: Average plaintext frequency exposure with varying numbers of range queries for dataset 1. Each value represents the average result from multiple experimental runs.

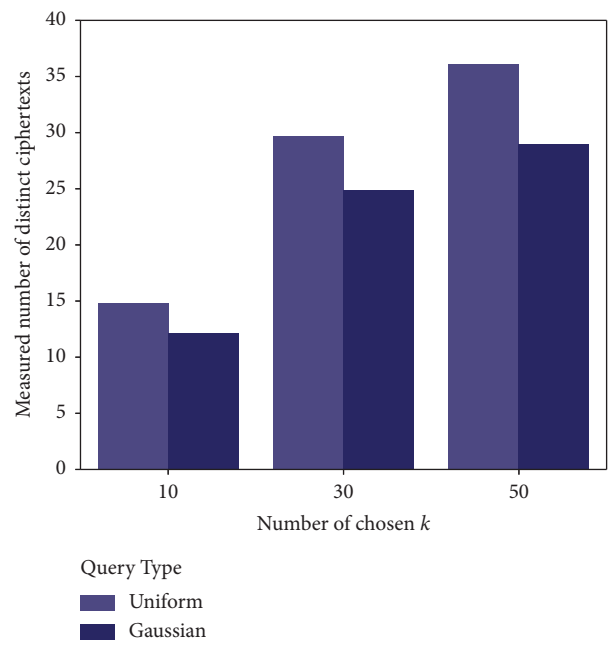

Figure 2: Average measured number of distinct ciphertexts with varying numbers of range queries for dataset 2. Each value represents the average result from multiple experimental runs.

6. Further Exploitations: Inference Attacks Using Join Queries and Association Rule Mining

The frequency-exposure vulnerability of FH-OPE and the weakness of the IND-FA-OCPA security model have been demonstrated through range queries. However, beyond range queries, more complex types of queries, such as join queries, may also potentially expose additional vulnerabilities in FH-OPE. While individual columns encrypted using FH-OPE might appear secure in isolation, joining multiple

tables can expose new relationships between data items, thereby allowing sensitive information to be deduced. To illustrate, we consider a hypothetical scenario involving a healthcare database encrypted with FH-OPE, consisting of two main tables:

(1) Patients: this table includes fields like "PatientID," "Age Group," and "Gender"

(2) Treatments: this table contains fields such as "PatientID," "Lengths of Stay," and "Diagnosis Code"

The sensitive numerical fields "Lengths of Stay" and "Diagnosis Code" are encrypted using the FH-OPE scheme. We assume an adversary has knowledge about the frequency distribution of patients' lengths of stay from their initial range queries and targets a specific "age group," for instance, "30–49." The adversary could execute the following SQL join query:

(1) SELECT Patients.Age_Group, Patients.Gender, Treatments.Lengths_of_Stay, Treatments.Diagnosis_Code

(2) FROM Patients

(3) JOIN Treatments ON Patients.PatientID = Treatments.PatientID

(4) WHERE Patients.Age_Group = "30–49"

The result of this join query allows the adversary to generate frequency distributions of the encrypted "Lengths of Stay" and "Diagnosis Code" for the targeted age group. Identifying correlations between "Lengths of Stay" and "Diagnosis Codes" may enable the adversary to infer that certain diseases correspond to longer hospital stays for patients within this age group. If the adversary has prior knowledge or assumptions about disease prevalence and average hospital stays, they could potentially deduce further information from the encrypted diagnosis codes, revealing a risk to privacy even within securely encrypted fields.

Moreover, adversaries can use the ordered nature of encrypted data in FH-OPE to perform advanced inference attacks, such as association rule mining [29]. This machine learning technique can reveal relations between different fields in a database that are not directly linked but share common attributes. For example, given the adversary's knowledge about the frequency distribution of patients' ages and lengths of stay, they could potentially discover rules like "if a patient is in the "X" age group and stays for "Y" days, they likely have a "Z" diagnosis." These associations can be found even when the diagnosis is encrypted using FH-OPE, emphasizing the serious privacy implications. To provide a deeper understanding of these potential threats, we present further extensive experiments in which our goal was to demonstrate that association rule mining attacks on real-world datasets can reveal sensitive information within a healthcare context, even when data are encrypted using the FH-OPE scheme.

6.1. Experiments on Further Exploitations

6.1.1. Datasets and Preprocessing. We utilized a real-world dataset originating from the New York State Department of Health, specifically, the "Hospital Inpatient Discharges (SPARCS De-Identified)" dataset available on the https://health.data.ny.gov website. This dataset includes a wealth of patient information, such as patient ID, age group, gender, lengths of stay, and CCS diagnosis Code. The data relating to lengths of stay and CCS diagnosis code have been encrypted using the FH-OPE scheme. Using join queries on this dataset, we focused on a subset of patient data. Specifically, we obtained the patient IDs for those in the long-term inpatients category, patients with lengths of stay exceeding 120 days. We assumed that an adversary had executed several range queries, thereby acquiring knowledge about the frequency distribution of lengths of stay. Figure 3 includes histograms that depict the distribution of lengths of stay and the CCS diagnosis code for patients within the 120+ days category.

Subsequently, we undertook a series of experiments on these encrypted datasets, each characterized by a different number of range queries executed on encrypted data. Specifically, we applied different volumes of range queries ($k = 400$, $k = 200$, $k = 100$, and $k = 50$) to the CCS diagnosis code, leading to distinct datasets with varying levels of exposed ciphertexts being generated. The primary purpose of our experiment is to assess the impact of varying levels of exposed ciphertexts on the quality of association rule mining. Therefore, we employed association rule mining on each of these datasets, as detailed in Table 2, and contrasted the outcomes.

6.1.2. Association Rule Mining Results on Datasets. To explore the impact of varying levels of exposed ciphertexts on the quality of association rule mining, we applied the Apriori algorithm [29] in Python. We applied this algorithm to each of the datasets (DS_{400}, DS_{200}, DS_{100}, and DS_{50}) defined in Table 2. Indeed, DS_{400} can essentially be considered the same as a plaintext dataset, as the frequency of ciphertexts is 100% exposed. Table 3 displays the rules yielded by applying the Apriori algorithm to the Hospital Inpatient Discharges (SPARCS De-Identified) dataset. For these results, the minimum support was set to 0.01, and the minimum confidence threshold was set to 0.6. These rules provide a baseline with which we can compare the association rule mining results for the remaining datasets.

Figures 4(a)–4(d) show the outcomes of association rule mining the datasets (DS_{400}, DS_{200}, DS_{100}, and DS_{50}) with the Apriori algorithm, maintaining a minimum support of 0.01 and a minimum confidence threshold of 0.05. Each figure represents a scatterplot graph displaying the distribution of the rule metrics: support, confidence, and lift. The resultant graphs for DS_{400}, DS_{200}, and DS_{100} depict strikingly similar distributions, indicating that the rules generated for these

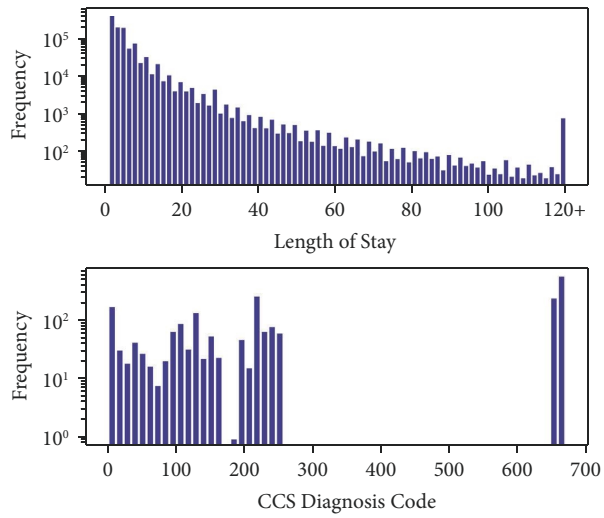

Figure 3: Distribution of length of stay and CCS diagnosis code for patients within the 120+ days category.

Table 2: Datasets representing different levels of distinct ciphertext exposure for varying numbers of range queries (k) on the CCS diagnosis code.

Dataset name	k	$\mathbb{E}[2k]$	$\mathbb{M}[2k]$	Exposure (%)
DS_{400}	400	155	156	100
DS_{200}	200	144	143	92
DS_{100}	100	113	113	72
DS_{50}	50	74	73	49

The total number of data points is 1894 ($n = 1894$), with 156 distinct values ($N = 156$). $\mathbb{E}[2k]$ represents the expected number of distinct ciphertexts after k queries according to equation (4), $\mathbb{M}[2k]$ represents the measured number of distinct ciphertexts after k queries, and exposure (%) is the corresponding percentage of $\mathbb{M}[2k]$ over the total distinct values N. All values in the above table are rounded to one decimal place.

Table 3: Rules resulting from the Apriori algorithm applied to the Hospital Inpatient Discharges (SPARCS De-identified) dataset.

No	Antecedents	Consequents	Antecedent support	Consequent support	Support	Confidence	Lift	Leverage	Conviction
1	(653)	(70 or Older)	0.025343	0.186906	0.016895	0.666667	3.566855	0.012159	2.439282
2	(109, M)	(50 to 69)	0.013200	0.359556	0.010032	0.760000	2.113715	0.005286	2.668515
3	(109)	(50 to 69)	0.021119	0.359556	0.012672	0.600000	1.668722	0.005078	1.601109
4	(233)	(M)	0.022770	0.167932	0.012144	0.884615	1.561474	0.004367	3.756776
5	(5)	(M)	0.026927	0.566526	0.019007	0.705882	1.245984	0.003752	1.473812
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

Each column is explained as follows: "Antecedents" are the items that precede, and "Consequents" are the items that follow. "Antecedent Support" and "Consequent Support" show the proportion of transactions in the data that contain the antecedent and consequent, respectively. "Support" indicates the frequency of the antecedent and consequent appearing together, while "Confidence" shows the conditional probability of the consequent given the antecedent. "Lift" measures how much more likely the antecedent and consequent are to occur together than if they were statistically independent, "Leverage" computes the difference between the observed frequency of the antecedent and consequent appearing together and what would be expected if they were independent, and "Conviction" indicates the dependency of the consequent on the antecedent.

datasets are nearly identical. This similarity suggests that despite reducing the amount of exposed ciphertext (from 100% for DS_{400} to 92% for DS_{200} and further down to 72% for DS_{100}), the quality of association rule mining did not significantly degrade. The rules mined under these conditions are, therefore, almost as insightful as those obtained from plaintext data. In contrast, the results for DS_{50}, where the frequency exposure is less than 50%, are clearly differentiated from the previous dataset. Despite the dissimilar graph shape, strong rules that were found in the more exposed datasets were also identified to some extent in the DS_{50} dataset.

Through our experimentation, we found that despite a reduction in ciphertext exposure from 100% to 49%, an adversary can still extract significant information from encrypted data. This insight is critical, as it shows that FH-OPE is vulnerable to inference attacks by enabling attackers to discern patterns and gather meaningful information. Notably, the derived association rules remained informative, regardless of the reduced data exposure. Furthermore, our findings definitively proved that even a limited number of range queries can allow an adversary to launch successful attacks, thus compromising the security of FH-OPE encrypted data.

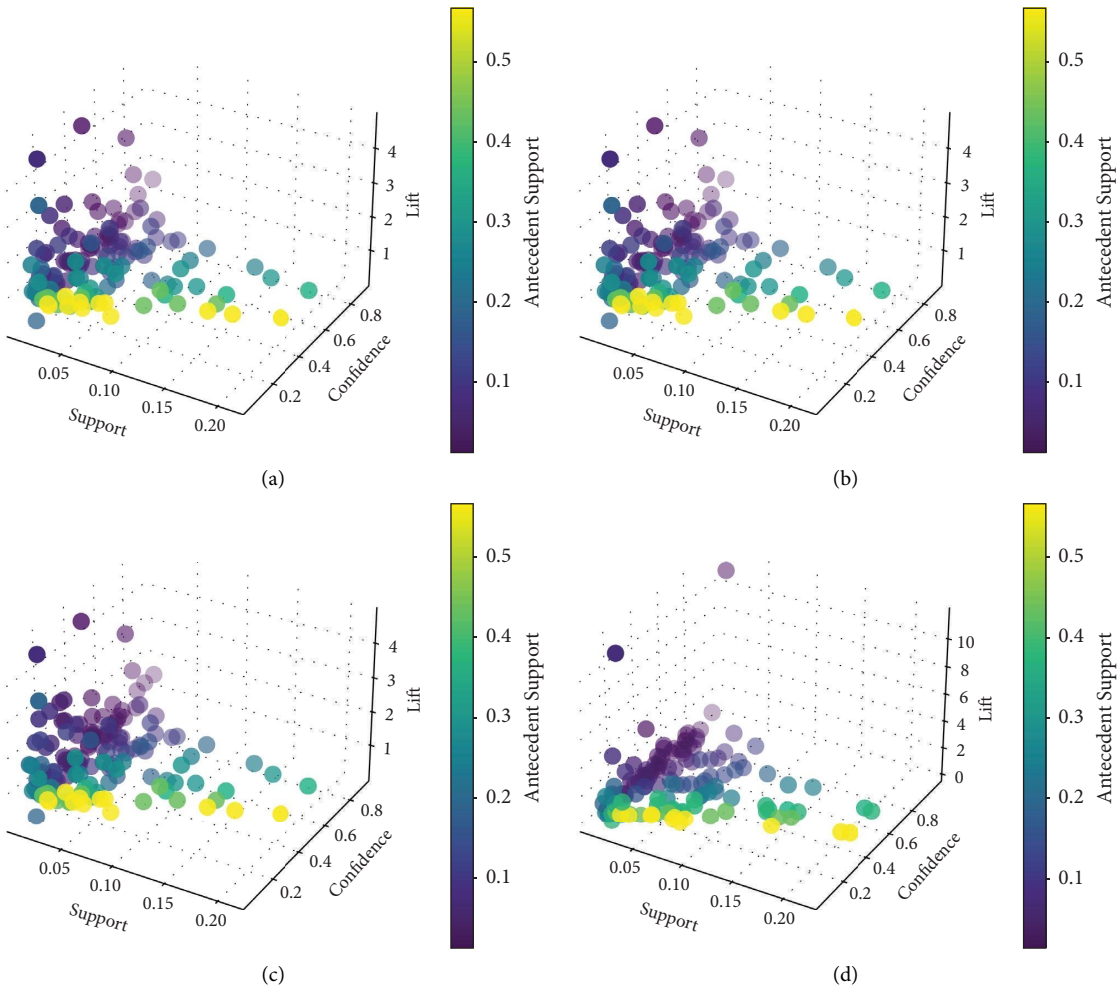

Figure 4: Distribution of support, confidence, and lift for association rules derived from the DS_{400}, DS_{200}, DS_{100}, and DS_{50} datasets using the Apriori algorithm. The color intensity of each point reflects the level of antecedent support, indicating the frequency of the rule's conditions in the respective datasets. (a) DS_{400}. (b) DS_{200}. (c) DS_{100}. (d) DS_{50}.

7. Conclusion

This study presented a comprehensive analysis of FH-OPE, specifically its vulnerability to frequency exposure through range queries. Our findings provide an overlooked aspect of the IND-FA-OCPA security model: the absence of consideration for vulnerabilities introduced by range queries. We have quantified frequency exposure using principles from the coupon collector's problem and probabilistic analyses to determine the number of range queries necessary to reveal the frequency of all plaintexts and estimate the expected number of unique ciphertexts revealed after executing a given number of range queries. Furthermore, our exploration goes beyond straightforward threats, considering more complex query types such as join queries and advanced attacks through association rule mining. Experimental analyses on real-world datasets have provided concrete evidence of these vulnerabilities, highlighting the risks associated with practical implementations of FH-OPE. Our results quantify the level of exposure risk and present the potential for inference attacks. Ultimately, our findings highlight that FH-OPE requires a more comprehensive security model that adequately addresses the risks posed by range queries. Further research is needed to develop new range-query methods that resist the vulnerabilities identified in this study. In conclusion, we hope our research contributes to a deeper understanding of the security challenges associated with FH-OPE.

Data Availability

The data that support the findings of this study are openly available in Allegheny County Employee Salaries at https://catalog.data.gov/dataset/allegheny-county-employee-salaries and Women's National Basketball Association at https://www.kaggle.com/datasets/jinxbe/wnba-player-stats-2017.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. NRF-2022R1F1A1062693).

References

[1] R. Agrawal, J. Kiernan, R. Srikant, and Y. Xu, "Order preserving encryption for numeric data," in *Proceedings of the 2004 ACM SIGMOD International Conference on Management of Data*, pp. 563–574, Paris, France, June 2004.

[2] A. Boldyreva, N. Chenette, Y. Lee, and A. O'neill, "Order-preserving symmetric encryption," in *Proceedings of the Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 224–241, Springer, Paris, France, April 2009.

[3] A. Boldyreva, N. Chenette, and A. O'Neill, "Order-preserving encryption revisited: improved security analysis and alternative solutions," in *Proceedings of the Annual Cryptology Conference*, pp. 578–595, Springer, Barbara, CA, USA, October 2011.

[4] S. Lee, T.-J. Park, D. Lee, T. Nam, and S. Kim, "Chaotic order preserving encryption for efficient and secure queries on databases," *IEICE- Transactions on Info and Systems*, vol. 92, no. 11, pp. 2207–2217, 2009.

[5] H. Kadhem, T. Amagasa, and H. Kitagawa, "Mv-opes: multivalued-order preserving encryption scheme: a novel scheme for encrypting integer value to many different values," *IEICE- Transactions on Info and Systems*, vol. 93, no. 9, pp. 2520–2533, 2010.

[6] L. Xiao, I.-L. Yen, and D. T. Huynh, "Extending order preserving encryption for multi-user systems," *IACR Cryptology ePrint Archive*, vol. 2012, p. 192, 2012.

[7] L. Xiao and I.-L. Yen, "A note for the ideal order-preserving encryption object and generalized order-preserving encryption," *IACR Cryptology ePrint Archive*, vol. 2012, p. 350, 2012.

[8] D. Liu and S. Wang, "Nonlinear order preserving index for encrypted database query in service cloud environments," *Concurrency and Computation: Practice and Experience*, vol. 25, no. 13, pp. 1967–1984, 2013.

[9] R. A. Popa, F. H. Li, and N. Zeldovich, "An ideal-security protocol for order-preserving encoding," in *Proceedings of the 2013 IEEE Symposium on Security and Privacy*, pp. 463–477, IEEE, San Francisco, CA, USA, February 2013.

[10] F. Kerschbaum and A. Schröpfer, "Optimal average-complexity ideal-security order-preserving encryption," in *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, pp. 275–286, New York, NY, USA, November 2014.

[11] D. S. Roche, D. Apon, S. G. Choi, A. Yerukhimovich, and Pope, "Partial order preserving encoding," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pp. 1131–1142, Vienna, Austria, October 2016.

[12] K. S. Kim, "New construction of order-preserving encryption based on order-revealing encryption," *Journal of Information Processing Systems*, vol. 15, no. 5, pp. 1211–1217, 2019.

[13] N. Shen, J.-H. Yeh, H.-M. Sun, and C.-M. Chen, "A practical and secure stateless order preserving encryption for outsourced databases," in *Proceedings of the 2021 IEEE 26th Pacific Rim International Symposium on Dependable Computing (PRDC)*, pp. 133–142, IEEE, Perth, Australia, December 2021.

[14] A. Tueno and F. Kerschbaum, "Efficient secure computation of order-preserving encryption," in *Proceedings of the 15th ACM Asia Conference on Computer and Communications Security*, pp. 193–207, Singapore, April 2020.

[15] J. Yang and K. S. Kim, "An efficient update algorithm for mutable order-preserving encryption," *IEEE Access*, vol. 10, pp. 102 009–102 018, 2022.

[16] F. Kerschbaum, "Frequency-hiding order-preserving encryption," in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pp. 656–667, Denver, CO, USA, October 2015.

[17] M. Maffei, M. Reinert, and D. Schröder, "On the security of frequency-hiding order-preserving encryption," in *Proceedings of the International Conference on Cryptology and Network Security*, pp. 51–70, Springer, Abu Dhabi, UAE, November 2017.

[18] J. Yang and K. S. Kim, "Practical frequency-hiding order-preserving encryption with improved update," *Security and Communication Networks*, vol. 2021, Article ID 1160305, 8 pages, 2021.

[19] S. Chen, L. Li, W. Zhang, X. Chang, Z. Han, and Bope, "Boundary order-preserving encryption scheme in relational database system," *IEEE Access*, vol. 9, pp. 30–124, 2021.

[20] D. Li, S. Lv, Y. Huang et al., "Frequency-hiding order-preserving encryption with small client storage," *Proceedings of the VLDB Endowment*, vol. 14, no. 13, pp. 3295–3307, 2021.

[21] R. A. Popa, C. M. Redfield, N. Zeldovich, and H. Balakrishnan, "Cryptdb: protecting confidentiality with encrypted query processing," in *Proceedings of the Twenty-Third ACM Symposium on Operating Systems Principles*, pp. 85–100, Cascais, Portugal, October 2011.

[22] S. L. Tu, M. F. Kaashoek, S. R. Madden, and N. Zeldovich, *Processing Analytical Queries over Encrypted Data*, Massachusetts Institute of Technology, Cambridge, MA, USA, 2013.

[23] F. Taigel, A. K. Tueno, and R. Pibernik, "Privacy-preserving condition-based forecasting using machine learning," *Journal of Business Economics*, vol. 88, no. 5, pp. 563–592, 2018.

[24] M. Naveed, S. Kamara, and C. V. Wright, "Inference attacks on property-preserving encrypted databases," in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pp. 644–655, Denver, CO, USA, October 2015.

[25] F. B. Durak, T. M. DuBuisson, and D. Cash, "What else is revealed by order-revealing encryption?" in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pp. 1155–1166, Vienna, Austria, October 2016.

[26] P. Grubbs, K. Sekniqi, V. Bindschaedler, M. Naveed, and T. Ristenpart, "Leakage-abuse attacks against order-revealing encryption," in *Proceedings of the 2017 IEEE Symposium on Security and Privacy (SP)*, pp. 655–672, IEEE, San Jose, CA, USA, May 2017.

[27] V. Bindschaedler, P. Grubbs, D. Cash, T. Ristenpart, and V. Shmatikov, "The tao of inference in privacy-protected databases," *Proceedings of the VLDB Endowment*, vol. 11, no. 11, pp. 1715–1728, 2018.

[28] X. Cao, J. Liu, Y. Shen, X. Ye, and K. Ren, "Frequency-revealing attacks against frequency-hiding order-preserving encryption," *Proceedings of the VLDB Endowment*, vol. 16, no. 11, pp. 3124–3136, 2023.

[29] R. Agarwal and R. Srikant, "Fast algorithms for mining association rules," in *Proceedings of the 20th International Conference on Very Large Data Basese*, p. 499, San Francisco, CA, USA, September 1994.