Megumi Ando* and Marilyn George

On the Cost of Suppressing Volume for Encrypted Multi-maps

Abstract: Structured encryption (STE) schemes allow a client to store sensitive data on a semi-trusted remote server and query the data. STE schemes strike a balance between privacy and efficiency by leaking some information to the server. In particular, many STE schemes leak the *volume pattern* i.e., response lengths, and the *query equality pattern* i.e., if any queries are repeated. Recently discovered leakage-abuse attacks demonstrate that leaking the volume pattern can be unsafe; that is, the server can reconstruct parts of the database from this leakage. To address this leakage, Kamara and Moataz proposed a novel multi-map encryption scheme, AVLH, that hides query volumes by padding responses with parts of other responses (Eurocrypt 2019). AVLH was shown to be more storage-efficient than the naive approach to pad responses with dummy values to reach the maximum response length. Subsequently, Patel et al. provided an even more efficient volume-hiding multi-map scheme, dprfMM (CCS 2019). Despite these advances, the costs of fully suppressing query volumes are still unclear. In this paper, we provide the first lower bounds on STE schemes for multi-maps that leak at most the query equality pattern. Surprisingly, we find that in many cases, such STE schemes cannot be more storage-efficient than naively padding to the maximum length.

Keywords: structured encryption, searchable encryption, lower bounds, volume-hiding

DOI Editor to enter DOI
Received ..; revised ..; accepted ...

1 Introduction

Structured encryption (STE) is a cryptographic primitive that enables a client to both store and efficiently query large and confidential data on an external untrusted server. Confidential data can also be stored and queried using cryptographic primitives such as fully-homomorphic encryption (FHE) or oblivious RAM (ORAM); however, these solutions often incur high computational and/or communication costs. STE allows for a flexible trade-off between the level of security offered, the types of queries supported, and the practical efficiency of the system. This trade-off is possible because STE schemes allow for the quantification of "leakage" which, roughly, is any meaningful information the untrusted server learns about either the input data or the queries issued by the client.

For example, the multi-map is a commonly-used data structure which associates a label to a tuple of values. When a multi-map is queried for a label, the corresponding tuple of values is returned. Encrypted multi-maps can be used to build wide-ranging functionality such as search over encrypted document collections and SQL queries over encrypted relational databases, e.g., [10, 33], examples from [30].

While the server cannot directly decrypt an encrypted multi-map, it can observe all the actions performed on the encrypted structure, and therefore infer some information through the leakage of the STE scheme that could include: (i) the *volume pattern* (the response lengths; for a multi-map, the lengths of the tuples) and (ii) the *query equality pattern* (repetitions in queries, also referred to as the search pattern). These two types of leakages (i) and (ii) are common in multi-map encryption schemes and were, until recently, considered reasonable and (efficiently) unavoidable. However, a recent line of work on leakage attacks has demonstrated that these leakages could be used to compromise security, e.g., see [5, 11, 14, 16, 19–21, 23, 26, 28, 32].

Leakage attacks. Leakage-abuse attacks, first defined by Cash, Grubbs, Perry, and Ristenpart, are a class of attacks that exploit a specific leakage profile rather than a specific STE scheme [6]. For these attacks to work, it is often assumed that the adversary knows the query distribution from which the queries are sampled but not the queries in plaintext (e.g., [19, 23]). *Volumetric attacks* are a kind of leakage-abuse attack in which the adversarial server reconstructs the queries and/or the database by computing on the volume pat-

***Corresponding Author: Megumi Ando:** MITRE, E-mail: mando@mitre.org
Marilyn George: Brown University, E-mail: marilyn_george@brown.edu

tern (see e.g., [3, 6, 19]). A recent example is a paper by Kornaropoulos, Papamanthou, and Tamassia that demonstrates a new kind of attack; here, the adversary does not even know the query distribution a priori [22]. The growing number of attacks and prevalence of the leakage patterns prompted a new line of work looking to reduce the leakage of STE schemes.

Leakage suppression. All known attacks on STE schemes use either the query equality pattern or the volume pattern (included in the larger leakage profile known as the *access pattern*), or a combination of both. As such, given that these patterns occur frequently in STE schemes, there has been interest in trying to hide, or suppress, one or both of these leakage patterns.

In order to suppress the query equality, one could use ORAM or ORAM-like techniques as proposed by query equality suppression frameworks for multi-maps [12, 18]. However, ORAM techniques are subject to existing logarithmic lower bounds on communication complexity [4, 13, 24]. Similarly, for encrypted multi-maps, Patel, Persiano, and Yeo proved that there must be a logarithmic overhead to suppress the *decoupled key-equality* pattern, which includes the query equality pattern [29]. Additionally, ORAM techniques also require the modification of the server storage at query time, which prevents efficiency enhancements such as parallel query processing. Despite the large body of work in query equality suppression, to the best of our knowledge, only one existing attack by Liu et al. [25] makes use of primarily the query equality pattern, and their attack is only effective for certain query distributions.

On the other hand, the overwhelming majority of attacks on STE schemes exploit the access pattern leakage, which consists of the values returned by the server for a query, e.g., see [5, 11, 14, 16, 19, 20]. In particular, these attacks rely on two components of the access pattern leakage: (i) the intersections between queried tuples and (ii) the query volumes. It follows that suppressing these patterns would make existing attacks significantly harder, or even impossible. Any response-hiding STE scheme in the standard frameworks [1, 2], i.e., a scheme in which all the responses from the server are encrypted, would hide (i). In contrast, hiding (ii) is not as simple, and naive solutions are prohibitively expensive. This challenge of both hiding query volumes while also achieving efficient STE schemes motivated the study of techniques referred to in the literature as *volume-hiding*.

Volume-hiding. Kamara and Moataz first proposed hiding the volume pattern for STE schemes [17]. Their multi-map scheme, AVLH, suppressed query volumes by padding the encrypted responses using other encrypted responses. Patel, Persiano, Yeo, and Yung later proposed a formal definition of volume-hiding, as well as a more efficient volume-hiding construction, known as dprfMM [30]. Although there has been significant interest in designing volume-hiding techniques, there has not yet been a formal study of the costs of hiding the volume pattern.

We make the case that the volume pattern is the most crucial of the common leakage patterns, and that it warrants an individual study. With this motivation, our paper investigates the following question: What is the cost of leaking "minimally" i.e., only the query equality pattern, while suppressing the query volumes entirely?

1.1 Our Contributions

We consider a multi-map encryption scheme Σ that supports label queries to the encrypted data structure.

In this paper, Σ always consists of four algorithms: Setup, Token, Query, and Resolve. During setup, the client runs the algorithm Setup on the multi-map MM; this generates an encrypted data structure EMM for storage on the server-side. After setup, the client can query the encrypted data structure by sending tokens (i.e., encrypted queries) to the server; the client runs the algorithm Token to determine the token corresponding to each (plaintext) query. To respond to a token/query τ, the server runs the algorithm Query on τ and the encrypted data structure EMM; this produces an encrypted response. The client can then decrypt this response by running the algorithm Resolve.

Thus, our contributions are for static, response-hiding, non-interactive, and non-self-adjusting structured encryption schemes. By *static*, we mean that the scheme does not support any functionality for updating the data structure. The scheme is *non-interactive* if the server can compute the encrypted response from the search token without further interacting with the client. It is *non-self-adjusting* if the encrypted structure does not need to be modified in the server storage after initial setup, which is appropriate when studying the cost of suppressing the volume pattern without the need to use ORAM-like techniques.

Consistent with prior work on all known static, response-hiding, non-interactive, non-self-adjusting multi-map encryption schemes (see, e.g., the pad-and-split [2] and statistical independence frameworks [1] for multi-map STE schemes), we assume that the encrypted multi-map EMM output by Setup contains a set

of encrypted values, and that each encrypted response can be thought of as a subset of EMM.

Our results pertain to the total size of the encrypted data structure, or the number of values in the encrypted multi-map (i.e., storage), given some restrictions on the number of values in an encrypted response (i.e., read-efficiency).

1.1.1 Minimally-leaking schemes

We chose to take a game-based approach since this allowed us to make succinct arguments for the lower bounds presented in this paper. To that end, we first provide a simple game-based definition that captures what it means for an encrypted multi-map scheme to be "minimally leaking." Our notion is equivalent to standard $(\mathcal{L}_{\mathsf{S}}, \mathcal{L}_{\mathsf{Q}})$-security where the setup leakage $\mathcal{L}_{\mathsf{S}}$ outputs the *size* (i.e., the total number of values in the multi-map) and the maximum response length, and the query leakage $\mathcal{L}_{\mathsf{Q}}$ outputs the query equality pattern. The proof of equivalence follows from directly adapting the proof by CGKO [10] that indistinguishability and semantic security for searchable symmetric encryption (SSE) schemes are equivalent. See Definition 3 in Section 3 for the formal definition of minimally-leaking, and Appendix A for the formal definitions of $(\mathcal{L}_{\mathsf{S}}, \mathcal{L}_{\mathsf{Q}})$-security and simulation-based minimally-leaking.

As noted in [18, 30], if Σ is a minimally-leaking encrypted multi-map scheme, then the length of each encrypted response produced by Σ must be at least the maximum response length t. (If there existed a query with response length $< t$, the adversary would be able to exclude any multi-map such that the response to that query is of max length.) We refer to Σ as being "optimally read-efficient" if each encrypted response length is exactly t. It can be shown that the storage of any optimally read-efficient encrypted multi-map is at least mt, where m is the number of labels in the multi-map. (See Theorem 7 in Appendix B for a formal argument.) This implies that the naive approach that pads each response with as many information-less "dummy" values as needed for the padded length to equal t, is optimally storage-efficient among all optimally read-efficient strategies.

One natural question is, "Can we do better than this pad-to-max approach?" If we remove the restriction that the scheme is optimally read-efficient (perhaps the read-efficiency is an expected small constant factor worse), could we save on storage?

One way to reduce storage is to use parts of other responses instead of always padding with information-less dummy values [17]. That is, to extend the length of the response $c'_1 = \{c_{1,1}, \ldots, c_{1,\ell_1 < t}\}$ to the maximum length t, we can append values from other responses to c'_1. For example, we can obtain an extension of c'_1, $c_1 = \{c_{1,1}, \ldots, c_{1,\ell_1}, c_{2,1}, \ldots, c_{2,t-\ell_1}\}$, by using values from another response $c'_2 = \{c_{2,1}, \ldots, c_{2,\ell_2 \geq t-\ell_1}\}$. In this example, the values $c_{2,1}, \ldots, c_{2,t-\ell_1}$ show up in both c_1 and c_2: in the context of c_1, they are "real" values, and in the context of c_2, they are "non-real". To avoid confusion, we will always refer to the method of reusing encrypted responses as "overlapping", and to the method of using information-less dummies as "padding".

Randomly-overlapping STE schemes. For an STE scheme to be of practical use, it should be both easy to implement and easy to analyze. To that end, for studying whether overlapping can reduce the storage of minimally-leaking schemes, we focus on the scenario where the set of non-real values is chosen independently based only on the unencrypted response (and the expected length of each encrypted response is $t^+ \geq t$); this scenario captures a large class of STE schemes that are simple to implement and analyze. In fact, our result is for the slightly more general case where the sets of non-real values are pair-wise independent, which we call "randomly overlapping," see Definition 9. In particular, we do not consider techniques that assign non-real values based on other properties: e.g., correlations between the frequencies of response lengths. In large part, this is because such techniques require a lot of care to ensure that the volume pattern remains hidden.

For our first result, we show that if Σ is randomly-overlapping, even using an arbitrarily large (but polynomially bounded) number of information-less dummy values, it still cannot be minimally-leaking:

Theorem 1. *If Σ is a randomly-overlapping encrypted multi-map scheme, then Σ is not minimally-leaking.*

We show this to be the case by recasting the problem of randomly selecting encrypted values for overlapping as a "sampling marbles from bags" problem.

One implication of Theorem 1 is that modifying the Patel et al.'s dprfMM so that it becomes "stash-less" would break the scheme's security; without some data in the stash, the scheme would be randomly-overlapping and, therefore, not minimally-leaking. Finally, we analyze Kamara and Moataz's AVLH scheme. While AVLH is not randomly-overlapping, surprisingly, we found that

it is not volume-hiding (as defined by Patel et al.) and, therefore, not minimally-leaking either:

Theorem 2. *AVLH [17] is not minimally-leaking.*

1.1.2 Sampled Minimally-leaking schemes

The setting in papers attacking STE papers is different from the setting in the standard security definition: the attacker mounting a leakage-abuse attack operates without knowing the unencrypted queries, whereas the adversary in the security game *chooses* the queries. In this paper, we are also interested in determining the minimum costs for achieving the security notion that is analogous to minimally-leaking in the weaker model. This model, which we refer to as the "sampled setting," has the challenger sampling a sequence of queries uniformly at random from a fixed query distribution. Then "sampled minimally-leaking" is equivalent to the simulation-based security that leaks at most the size and the maximum response length in the sampled setting. We refer the reader to Section 3 and Appendix A for formal definitions.

In the sampled setting, when the adversary observes that some responses are shorter than the maximum response length, this does not immediately exclude some multi-maps. From this, it seems plausible that an encrypted multi-map encryption scheme that satisfies the weaker sampled definition can be more efficient than a fully secure one.

As explained earlier in Section 1.1.1, in the standard security setting where the adversary chooses the queries, each response must contain at least t encrypted values, where t is the maximum response length. In this paper, we show that the analogous lower bound in the sampled security setting is less restrictive: for all i, the i^{th} shortest response length $|\mathcal{E}_i|$ must be at least $\min(\lfloor \frac{N}{m-i+1} \rfloor, t)$, where N is the number of values in the multi-map, and m is the number of queries. Note that for $i < J \stackrel{\text{def}}{=} m - \lfloor \frac{N}{t} \rfloor + 1$, $\lfloor \frac{N}{m-i+1} \rfloor$ is strictly smaller than t. Let the *read-efficiency curve* be defined as follows:

$$\mathsf{RE}(i) \stackrel{\text{def}}{=} \begin{cases} \lfloor \frac{N}{m-i+1} \rfloor, & \text{for } 1 \le i < J \\ t, & \text{for } J \le i \le m; \end{cases}$$

Intuitively, we can think of $\mathsf{RE}(\cdot)$ as the curve that tightly bounds all possible multi-maps with the parameters N, m, and t.

Theorem 3. *With overwhelming probability, for every $i \in [m]$, the length of the i^{th} shortest encrypted response, denoted $|\mathcal{E}_i|$, is at least $\mathsf{RE}(i)$.*

This result is tight since padding up to the read-efficiency curve $\mathsf{RE}(\cdot)$ is a valid sampled minimally-leaking scheme. Thus, in the context of being sampled minimally-leaking, a multi-map encryption scheme is *optimally read-efficient* if for all i, $|\mathcal{E}_i| = \mathsf{RE}(i)$ with probability 1.

Balanced STE schemes. In the standard setting, the optimal storage among optimally read-efficient multi-map schemes is mt; see Theorem 7 in Appendix B. For our final result, we present a similar result to Theorem 7 in the sampled setting. In particular, we derive the optimal storage among a class of optimally read-efficient multi-map schemes, which we call "balanced." Below, we first define this class of schemes.

Let the range $[1, 2, \ldots, m]$ be the indices of the m queries, i.e., the x-axis for the read-efficiency curve. We can partition the range into two parts: the "lefthand-side" (indices $\mathsf{LHS} = [1, 2, \ldots, J-1]$) and the "righthand-side" (indices $\mathsf{RHS} = [J, J+1, \ldots, m]$). (See Figure 1 below for a pictorial depiction of lefthand-side and righthand-side.) In Lemma 1, we show that any non-real value in the righthand-side must be "covered by" a (real or information-less) value from the lefthand-side; that is, for every $i \in \mathsf{RHS}$, for each non-real value e for the i^{th} shortest response $\mathcal{E}_i$, there exists $j \in \mathsf{LHS}$ such that e is a real value only for $\mathcal{E}_j$, or else e is an information-less value.

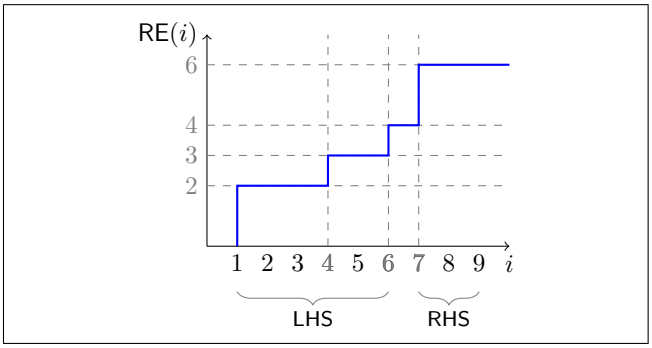


Fig. 1. The read-efficiency curve for $N = 18$, $m = 9$, $t = 6$. The indices in $\mathcal{I} = \{1, 4, 6, 7\}$ are considered "steps" since the value of the read-efficiency curve increases at each of the step indices. The indices $\mathsf{LHS} = \{1, \ldots, 7\}$ are those in the lefthand-side, and the indices $\mathsf{RHS} = \{8, 9\}$ are those in the righthand-side.

Consider the following two extremes: let MM_1 be the multi-map with as many "short" responses (i.e.,

of length $|\mathcal{E}_1|$) as possible, and let MM_2 be a multi-map with the maximum number of "long" responses (i.e., of length $|\mathcal{E}_m| = t$). If Σ is sampled minimally-leaking, then the observable properties of the encrypted data structure EMM_1 produced by running Setup on MM_1 is indistinguishable from those of EMM_2 produced from MM_2. These include the "overlap" between the lefthand-side and the righthand-side of the read-efficiency curve (i.e., the encrypted values that show up in both the shorter encrypted responses $\mathcal{E}_1, \ldots, \mathcal{E}_{J-1}$ and the longer ones $\mathcal{E}_J, \ldots, \mathcal{E}_m$). If Σ is optimally read-efficient, then from Lemma 1, it follows that the non-real values in the righthand-side of EMM_1 must be covered by real/information-less values in the lefthand-side. Moreover, since the overlap between the lefthand-side and the righthand-side is observable, it follows that the same overlap must exist when the multi-map is MM_2 instead.

By construction, this implies that in EMM_2, the real values in the righthand-side must cover values in the lefthand-side. Therefore the following is a natural idea: if there are more values on one side of the read-efficiency curve, then the extra values are used to cover the other side. We say that an encrypted multi-map scheme is "balanced" if this is always the case; that is, a similar restriction as Lemma 1 holds in the other direction: every non-real value in the lefthand-side must be covered by a (real or information-less) value from the righthand-side. Our bound is as follows:

Theorem 4. *Let Σ be balanced, and let $\sum_{i \in \mathsf{RHS}} \mathsf{RE}(i) = N$ where N is the size of the multi-map. With overwhelming probability, Σ's setup algorithm adds at least $\sum_{i \in \mathsf{LHS}} \mathsf{RE}(i) - N$ dummy values to the encrypted structure.*

In Section 5.3, we show experimentally that our bound in Theorem 4 essentially matches the scheme that naively pads up to the read-efficiency curve. This is further evidence that the class of balanced STE schemes is sufficiently general.

1.2 Related Work

Our work continues the line of investigation that investigates the security-efficiency trade-offs of STE schemes for multi-maps. Previous work has investigated efficiency trade-offs for multi-maps in the context of searchable symmetric encryption (SSE), using an underlying multi-map structure where each keyword maps to a tuple of search results. Cash and Tessaro first showed that a multi-map scheme must either pad to $\omega(N)$ in the total number of results N or have highly non-continuous reads [7]. Later, Asharov, Segev, and Shahaf proposed the pad-and-split framework and showed that multi-map schemes required $\Omega(N \log N)/\log L$ storage where the locality L was the maximum number of contiguous reads to retrieve any encrypted response [2]. Asharov, Naor, Segev, and Shahaf also proposed a second framework for SSE schemes known as the statistical independence framework and showed that there exist SSE schemes with both optimal storage and locality in this framework [1]. The existing bounds assume that both the access pattern (which includes the volumes) and the query equality pattern are leaked. In contrast to previous work of this nature, we would like to understand the additional efficiency costs to suppress the volume leakage for an encrypted multi-map, while still leaking the query equality. Our results pertain to the read efficiency and storage of schemes that leak (beyond the dimensions of the database) *at most the query equality.*

2 Preliminaries

Notation. For a natural number n, $[n]$ is the set $\{1, \ldots, n\}$. For a set Set, we denote the cardinality of Set by $|\mathsf{Set}|$, and $\mathsf{item} \leftarrow\!\!\$\, \mathsf{Set}$ is an item from Set chosen uniformly at random. If Dist is a probability distribution over Set, $\mathsf{item} \leftarrow \mathsf{Dist}$ is an item sampled from Set according to Dist. For an algorithm Algo, $\mathsf{output} \leftarrow \mathsf{Algo}(\mathsf{input})$ is the (possibly probabilistic) output from running Algo on input.

We say that a function $f : \mathbb{N} \mapsto \mathbb{R}$ is *negligible* in the parameter λ, written $f(\lambda) = \mathsf{negl}(\lambda)$, if for a sufficiently large λ, $f(\lambda)$ decays faster than any inverse polynomial in λ. If a function $f(\lambda)$ is non-negligible in the parameter λ, we denote this as $f(\lambda) = \mathsf{nonnegl}(\lambda)$. When λ is the security parameter, an event E_λ is said to occur *with (non-)negligible probability* if the probability of E_λ can(not) be bounded above by a function negligible in λ. An event occurs with overwhelming probability if its complement occurs *with negligible probability.*

Multi-maps. Let $\mathbb{MM}$ be a space of possible multi-maps, defined by a label space $\mathbb{L} = \{l_1, \ldots, l_m\}$. Any multi-map MM from the space $\mathbb{MM}$ will contain the same set of labels, which is fixed in advance. In particular, this means that a multi-map structure does not need to know how to respond to queries on labels outside of $\mathbb{L}$ to be correct. Each label l_i corresponds to a tuple of

values represented as $\mathsf{MM}(l_i)$. The correct response to a query for label l_i contains the values in $\mathsf{MM}(l_i)$. The *size* of the multi-map MM, denoted $|\mathsf{MM}|$, is the total number of values in MM, i.e., $|\mathsf{MM}| \stackrel{\text{def}}{=} \sum_{i\in\{1,\ldots,m\}} |\mathsf{MM}(l_i)|$.

3 Definitions

A structured encryption scheme [9] consists of the algorithms: Setup, Token, Query, and $\mathsf{Resolve}$. We describe the syntax of these algorithms below.

Definition 1 (Structured encryption)**.** *A structured encryption (STE) scheme Σ for the multi-map space $\mathbb{MM}$ and the label space $\mathbb{L}$ consists of the tuple $(\mathsf{Setup}, \mathsf{Token}, \mathsf{Query}, \mathsf{Resolve})$, where*

- Setup *is an efficient, i.e., probabilistic polynomial-time (p.p.t.), algorithm that takes as input the security parameter 1^λ and the multi-map $\mathsf{MM} \in \mathbb{MM}$, and outputs the key K, and the encrypted multi-map EMM, i.e., $(K, \mathsf{EMM}) \leftarrow \mathsf{Setup}(1^\lambda, \mathsf{MM})$.*
- Token *is an efficient algorithm that takes as input the key K and a query l, and outputs a query token τ for the label, i.e., $\tau \leftarrow \mathsf{Token}(K, l)$.*
- Query *is an efficient algorithm that takes as input the encrypted multi-map EMM and the token τ, and outputs the encrypted response c, i.e., $c \leftarrow \mathsf{Query}(\mathsf{EMM}, \tau)$.*
- $\mathsf{Resolve}$ *is an efficient algorithm that takes as input the key K, and the encrypted response c, and outputs the decrypted response r, i.e., $r \leftarrow \mathsf{Resolve}(K, c)$.*

There are other flavors of STE schemes: dynamic, interactive, self-adjusting, and/or response-revealing; as well as schemes that require the client to hold a small stash containing some data, but we will not consider these alternatives here. An STE scheme is *correct* if it returns the correct responses with overwhelming probability:

Definition 2 (Correctness)**.** *An STE scheme for the space of multi-maps $\mathbb{MM}$, and the label space $\mathbb{L}$ is correct if for any $\mathsf{MM} \in \mathbb{MM}$ and sequence of queries $q_1, \ldots, q_j \in \mathbb{L}$,*

$$\begin{aligned}&\Pr[(K, \mathsf{EMM}) \leftarrow \mathsf{Setup}(1^\lambda, \mathsf{MM});\\ &\quad r_1 \leftarrow \mathsf{Resolve}(K, \mathsf{Query}(\mathsf{EMM}, \mathsf{Token}(K, q_1)));\\ &\quad \vdots\\ &\quad r_j \leftarrow \mathsf{Resolve}(K, \mathsf{Query}(\mathsf{EMM}, \mathsf{Token}(K, q_j))) :\\ &r_1 \equiv \mathsf{MM}(q_1),\ \ldots,\ r_j \equiv \mathsf{MM}(q_j)] = 1 - \mathsf{negl}(\lambda), \quad (1)\end{aligned}$$

where "$a \equiv b$" means that a and b are equal up to a permutation. The scheme is perfectly *correct if for all values $\lambda \in \mathbb{N}$, the probability in* (1) *is* 1.

3.1 Security in the standard setting

Let $\mathbb{L} = \{l_1, \ldots, l_m\}$ be the fixed label space of the multi-map. The adversary $\mathcal{A}$ is assumed to be semi-honest; that is, we assume that $\mathcal{A}$ follows the encrypted multi-map scheme's query protocol in the role of the server.

Formally, we define security with respect to the following game. Let $q_1, \ldots, q_j \in \mathbb{L}$ be a sequence of queries to the multi-map, where each query corresponds to a label. The *query equality pattern* of the sequence $(q_1, \ldots, q_j)$ is the binary matrix indicating which queries are equal to each other.

The game $\mathsf{MLGame}^{\mathcal{A}}_{\Sigma}(1^\lambda)$ is parameterized by the security parameter 1^λ, the adversary $\mathcal{A}$, and the STE scheme $\Sigma = (\mathsf{Setup}, \mathsf{Token}, \mathsf{Query}, \mathsf{Resolve})$.

1. The adversary $\mathcal{A}$ chooses two multi-maps MM_0 and MM_1 of the same "dimensions" (i.e., size and maximum response length) and sends $(\mathsf{MM}_0, \mathsf{MM}_1)$ to the challenger $\mathcal{C}$.
2. $\mathcal{A}$ picks two sequences of queries, $\vec{q}_0 = (q_{0,1}, \ldots, q_{0,\mathsf{poly}(\lambda)})$ and $\vec{q}_1 = (q_{1,1}, \ldots, q_{1,\mathsf{poly}(\lambda)})$, with the same query equality pattern and sends $(\vec{q}_0, \vec{q}_1)$ to $\mathcal{C}$.
3. The challenger $\mathcal{C}$ samples a bit $b \leftarrow\!\!\$\, \{0,1\}$ uniformly at random, runs $\mathsf{Setup}(1^\lambda, \mathsf{MM}_b)$, and sends the resulting encrypted multi-map EMM to $\mathcal{A}$.
4. For each $q_{b,i}$, $\mathcal{C}$ runs $\mathsf{Token}(K, q_{b,i})$, where K is the key generated in step 1. $\mathcal{C}$ sends the tokens $(\tau_1, \ldots, \tau_{\mathsf{poly}(\lambda)})$ to $\mathcal{A}$.
5. Finally, $\mathcal{A}$ outputs a guess b' and wins if $b' = b$.

We define "minimally-leaking," a special case of $\mathcal{L}$-security [10], as follows:

Definition 3 (Minimally-leaking)**.** *The encrypted multi-map scheme Σ is minimally-leaking if for all p.p.t. adversaries $\mathcal{A}$, the advantage that $\mathcal{A}$ has in winning $\mathsf{MLGame}^{\mathcal{A}}_{\Sigma}(1^\lambda)$ is negligible in the security parameter λ, i.e., $\left|\Pr\left[\mathcal{A} \text{ wins } \mathsf{MLGame}^{\mathcal{A}}_{\Sigma}(1^\lambda)\right] - \frac{1}{2}\right| = \mathsf{negl}(\lambda)$.*

In the STE literature, security is defined with respect to two functions, the setup leakage $\mathcal{L}_\mathsf{S}$ and the query leakage $\mathcal{L}_\mathsf{Q}$, which together make up the *leakage profile*; see, for example, [17, 18] for a more thorough exposi-

tion on leakage profiles. Minimally-leaking is related to standard $(\mathcal{L}_{\mathsf{S}}, \mathcal{L}_{\mathsf{Q}})$-security as follows:

Theorem 5. *The encrypted multi-map scheme Σ is minimally-leaking if and only if it is non-adaptively $(\mathcal{L}_{\mathsf{S}}, \mathcal{L}_{\mathsf{Q}})$-secure, where the setup leakage $\mathcal{L}_{\mathsf{S}}$ reveals only the size and the maximum response length of the multi-map, and the query leakage $\mathcal{L}_{\mathsf{Q}}$ reveals only the query equality pattern.*

The proof follows from Curtmola et al.'s proof that indistinguishability for SSE schemes is equivalent to semantic security for SSE schemes [10].

3.2 Security in the "sampled" setting

In the standard setting, the adversary chooses a sequence of queries: $q_1, \ldots, q_j$. In this paper, we will also consider an alternative scenario, in which the adversary does not choose the queries. Instead, the challenger samples the queries uniformly at random, i.e., $q_1, \ldots, q_j \leftarrow\!\$\, \mathbb{L}$. This *sampled* security definition is inspired by the adversary models used in leakage attacks on STE schemes.

Consider the following modification to $\mathsf{MLGame}^{\mathcal{A}}_{\Sigma}(1^\lambda)$. The modification is boxed for better readability:

The game $\mathsf{SMLGame}^{\mathcal{A}}_{\Sigma}(1^\lambda)$ is parameterized by the security parameter 1^λ, the adversary $\mathcal{A}$, and the STE scheme $\Sigma = (\mathsf{Setup}, \mathsf{Token}, \mathsf{Query}, \mathsf{Resolve})$.

1. The adversary $\mathcal{A}$ chooses two multi-map MM_0 and MM_1 of the same "dimensions" (i.e., size and upper bound on the maximum response length) and sends $(\mathsf{MM}_0, \mathsf{MM}_1)$ to the challenger $\mathcal{C}$.
2. $\mathcal{C}$ samples $q_1, \ldots, q_{\mathsf{poly}(\lambda)}$ independently and uniformly at random from the query/label space $\mathbb{L}$.
3. The challenger $\mathcal{C}$ samples a bit $b \leftarrow\!\$\, \{0,1\}$ uniformly at random, runs $\mathsf{Setup}(1^\lambda, \mathsf{MM}_b)$. $\mathcal{C}$ sends the encrypted multi-map EMM to $\mathcal{A}$.
4. For each q_i, $\mathcal{C}$ runs $\mathsf{Token}(K, q_i)$, where K is the key generated in step 1, and sends the resulting tokens $(\tau_1, \ldots, \tau_{\mathsf{poly}(\lambda)})$ to $\mathcal{A}$.
5. Finally, $\mathcal{A}$ outputs a bit b' and wins if $b' = b$.

We define security in the "sampled" setting, which formalizes the security definition for papers attacking encrypted multi-map schemes, e.g., [19], as follows:

Definition 4 (Sampled minimally-leaking)**.** *The encrypted multi-map scheme Σ is sampled minimally-leaking if for all p.p.t. adversaries $\mathcal{A}$, the advantage that $\mathcal{A}$ has in winning $\mathsf{SMLGame}^{\mathcal{A}}_{\Sigma}(1^\lambda)$ is negligible in the security parameter λ, i.e., $\left|\Pr\left[\mathcal{A} \text{ wins } \mathsf{SMLGame}^{\mathcal{A}}_{\Sigma}(1^\lambda)\right] - \frac{1}{2}\right| = \mathsf{negl}(\lambda)$.*

We provide an equivalent simulation-based definition in Appendix A. We note here that minimally-leaking directly implies sampled minimally-leaking; see Appendix A.1 for the proof.

3.3 Efficiency of encrypted multi-map schemes

In this section, we present general terminology we will use in this paper to discuss the efficiency of an encrypted multi-map scheme. Let Σ be an encrypted multi-map scheme over the label space $\mathbb{L}$. For each label $l \in \mathbb{L}$, we assume l is associated with a tuple of values, denoted as $\mathsf{MM}(l)$. Following prior work [1, 2], we assume that the encrypted multi-map EMM output by Setup contains a set of encrypted values. Then each encrypted response c_i can be represented as a subset of this set of encrypted values. If a value occurs in more than one response, it is encrypted separately for each response.

The encrypted values in the response c_i can be further classified into *real* and *non-real* encrypted values (with respect to c_i). In the definitions below, let K be the key output by the Setup algorithm run on a multi-map MM, and for label $l_i \in \mathbb{L}$, let c_i be the encrypted response $c_i \leftarrow \mathsf{Query}(\mathsf{EMM}, \mathsf{Token}(K, l_i))$.

Definition 5 (Real encrypted value)**.** *An encrypted value e is real for the encrypted response c_i if the corresponding decrypted value $(\mathsf{val}, w) \leftarrow \mathsf{Resolve}(K, e)$ is an element of the tuple $\mathsf{MM}(l_i)$.*

Any additional values that are part of the encrypted response c_i are referred to as "non-real" for c_i. Then from our earlier assumption about repeated values being encrypted separately, an encrypted value is real for at most one encrypted response c_i, and non-real for every other encrypted response c_j. If a value is non-real for *all* encrypted responses c_i, then that value is considered "information-less." The encryption of such an empty value is referred to as a *dummy*.

We define "overlaps" between encrypted responses c_i and c_j as the encrypted values that are common to both encrypted responses.

Definition 6 (Overlap)**.** *An encrypted value e is an overlap for the encrypted responses c_i and c_j if it is an element of both the responses c_i and c_j.*

Then each overlapping encrypted value between any two responses c_i and c_j is either a dummy record, or real for either response c_i or response c_j.

We define the notions of expected read efficiency and storage overhead for encrypted multi-map schemes. Our definitions are inspired by Cash and Tessaro's definitions of the same name [8]; a notable difference is that read efficiency is the *additive* overhead in extraneous encrypted values read by the server (as opposed to the multiplicative overhead). This is to accommodate the fact that we allow for responses of length zero; when such a response exists, the multiplicative overhead in read-efficiency is necessarily unbounded.

Definition 7 (Additive read efficiency)**.** *The encrypted multi-map scheme Σ is r-read-efficient for the input space $\mathbb{MM}$ if for every $\lambda \in \mathbb{N}$, $\mathsf{MM} \in \mathbb{MM}$, and instance $(K, \mathsf{EMM}) \leftarrow \mathsf{Setup}(1^\lambda, \mathsf{MM})$, the average number of non-real encrypted values for the encrypted response c_i is at most r, i.e., $\mathbb{E}_{w \leftarrow\$ \mathbb{L}}[|c_i| - |\mathsf{Resolve}(K, c_i)|] \leq r$.*

We define the storage overhead of an encrypted multi-map scheme as the multiplicative increase in the number of values in the encrypted multi-map EMM as compared to the input multi-map MM.

Definition 8 (Storage overhead)**.** *The encrypted multi-map scheme Σ has s-storage overhead for the input space $\mathbb{MM}$ if for every $\lambda \in \mathbb{N}$, $\mathsf{MM} \in \mathbb{MM}$ and instance $(K, \mathsf{EMM}) \leftarrow \mathsf{Setup}(1^\lambda, \mathsf{MM})$, the total number of encrypted values present in the encrypted multi-map $|\mathsf{EMM}| \leq sN$, where $N = |\mathsf{MM}|$ for the input multi-map.*

4 Bounds on Minimally-leaking schemes

As noted in prior work [18, 30], if Σ is a volume-hiding encrypted multi-map scheme, then the encrypted responses it produces must be each at least as long as the maximum response length t. If there existed a label l_i such that the encrypted response c_i for l_i were shorter than the maximum length, this would reveal that $|\mathsf{MM}(l_i)|$ is not t. Since minimally-leaking implies volume-hiding, it follows that if Σ is minimally-leaking, then each encrypted response is at least t long. An encrypted multi-map scheme that is clearly minimally-leaking is simply padding every response up to the maximum response length using dummy values. Recent papers [18, 30] suggest that we can save on storage by padding with parts of other responses instead of using dummy values; we will refer to this latter technique as *overlapping*. We will reserve the term *padding* to mean using dummy values.

In this section, we first show that even if we relax the restriction that Σ is optimally read-efficient, choosing the overlaps "at random" cannot help to reduce the required storage (Theorem 1). Next, we show that AVLH is not minimally-leaking (Theorem 2).

To prove these theorems formally, we introduce the following setup: Let $\mathbb{L} = \{l_1, \ldots, l_m\}$ be the fixed set of queries. Let MM be any multi-map of size N and maximum response length t. Let Σ be any (correct) multi-map scheme for the label space $\mathbb{L}$. Let $(\mathsf{EMM}, c_1, \ldots, c_m) \leftarrow \mathsf{MLExp}_{\Sigma,\mathsf{MM}}(1^\lambda)$, where $\mathsf{MLExp}_{\Sigma,\mathsf{MM}}(1^\lambda)$ is defined below. Let S be the number of encrypted values in EMM. For each $i \in [m]$, let $\ell_i \stackrel{\text{def}}{=} |c_i|$ denote the length of the i^{th} encrypted response c_i. Without loss of generality, we will assume that the number of labels is at least two, i.e., $m \geq 2$.

Consider the following experiment, $\mathsf{MLExp}_{\Sigma,\mathsf{MM}}(1^\lambda)$, parameterized by the security parameter 1^λ, the encrypted multi-map scheme Σ, and the multi-map MM:

1. Run Σ's setup algorithm Setup on the multi-map MM to obtain the key K and the encrypted multi-map EMM, i.e., $(K, \mathsf{EMM}) \leftarrow \mathsf{Setup}(1^\lambda, \mathsf{MM})$.
2. For each label $l_i \in \mathbb{L}$:
   i. Run Σ's token algorithm Token on the key K and the label l_i; let τ_i be the output from running Token.
   ii. Then, run Σ's query algorithm Query on the encrypted multi-map EMM (from step 1) and the output τ_i to get the the encrypted response c_i, i.e., $\tau_i \leftarrow \mathsf{Token}(K, l_i)$; $c_i \leftarrow \mathsf{Query}(\mathsf{EMM}, \tau_i)$.
3. Output $(\mathsf{EMM}, c_1, \ldots, c_m)$.

4.1 Randomly-overlapping schemes cannot be Minimally-leaking

Recall that the naive approach to pad each response with dummy values so that each padded response is equal to the maximum response length is optimally storage-efficient among all optimally read-efficient

strategies. Thus, we have established that overlapping cannot reduce the storage of minimally-leaking and optimally read-efficient schemes.

Here, we show that even if the scheme Σ is not necessarily optimally read-efficient, overlapping may still not help achieve better storage efficiency. Specifically, we consider the case where the scheme chooses the overlaps for each response randomly from the set of all non-real encrypted values for the response and show that, in this case, Σ cannot be minimally-leaking.

We first provide a formal definition of what we mean by randomly-overlapping; to do this formally, we define the game $\mathsf{ROGame}_{\Sigma,\mathcal{A}}(1^\lambda)$, below: Recall that $(\mathsf{EMM}, c_1, \ldots, c_m) \leftarrow \mathsf{MLExp}_{\Sigma,\mathsf{MM}}(1^\lambda)$. For all $i, j \in [m]$, recall that the *overlap* between c_i and c_j is the intersection between c_i and c_j. For all $i \in [m]$, recall that the *real* encrypted values for c_i are the encrypted values in c_i whose decryptions are part of the response $\mathsf{MM}(l_i)$ (Definition 5). So, the *non-real* encrypted values for c_i are the encrypted values in c_i that are not real ones for c_i.

Consider the following game, $\mathsf{ROGame}_{\Sigma,\mathcal{A}}(1^\lambda)$, which is parameterized by the security parameter 1^λ, the encrypted multi-map scheme Σ, and the adversary $\mathcal{A}$:

1. First, the adversary $\mathcal{A}$ chooses a multi-map MM and indices $i, j \in [m]$ and sends it to the challenger $\mathcal{C}$.
2. The challenger $\mathcal{C}$ runs $\mathsf{MLExp}_{\Sigma,\mathsf{MM}}$. Let $(\mathsf{EMM}, c_1, \ldots, c_m)$ denote the output from running the experiment, i.e., $(\mathsf{EMM}, c_1, \ldots, c_m) \leftarrow \mathsf{MLExp}_{\Sigma,\mathsf{MM}}(1^\lambda)$. Let $\mathcal{E}$ denote the set of all encrypted values in EMM.
3. Next, $\mathcal{C}$ picks a random bit $b \leftarrow\$ \{0,1\}$. If $b = 1$, $\mathcal{C}$ modifies the encrypted responses as follows. For each $k \in \{i, j\}$, let $\mathsf{Reals}(c_k)$ and $\mathsf{Nonreals}(c_k)$ be the set of real encrypted values in c_k and the set of non-real encrypted values in c_k, respectively. $\mathcal{C}$ replaces the non-real encrypted values for c_k with a uniformly random size-$|\mathsf{Nonreals}(c_k)|$ sample from the set $\mathcal{E} \setminus \mathsf{Reals}(c_k)$.
4. After possibly modifying the encrypted responses in step 3, $\mathcal{C}$ sends (c_i, c_j) to $\mathcal{A}$.
5. Finally, $\mathcal{A}$ outputs a guess b' for b and wins if $b' = b$.

Recall that S is the number of encrypted values in the encrypted multi-map EMM.

Definition 9 (Randomly-overlapping)**.** *The encrypted multi-map scheme Σ is randomly-overlapping if two conditions are satisfied: (i) the length of each encrypted response is drawn independently from a fixed distribution, and (ii) for every p.p.t. adversary $\mathcal{A}$, $\left|\Pr\left[\mathcal{A} \text{ wins } \mathsf{ROGame}_{\Sigma,\mathcal{A}}(1^\lambda)\right] - \frac{1}{2}\right| = \mathsf{negl}(\lambda)$.*

Here, we show an impossibility result for when Σ is randomly-overlapping.

Theorem 1. *If Σ is a randomly-overlapping encrypted multi-map scheme, then Σ is not minimally-leaking.*

Proof. Let $i, j \neq i$ be two indices from $[m]$. Let MM be any multi-map such that $|\mathsf{MM}(l_i)| = |\mathsf{MM}(l_j)| = \ell$. Let $T_i = |c_i|$ and $T_j = |c_j|$.

We show that if Σ is randomly overlapping, then the expected number of overlaps between c_i and c_j is a function of the original response length ℓ. To see this, we study the following sampling problem; (see Figure 2):

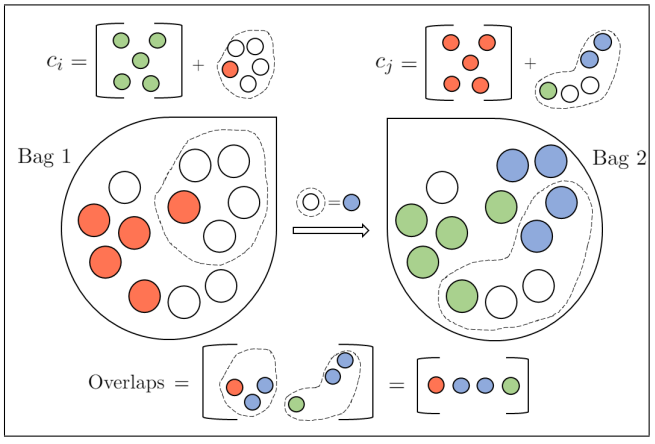


Fig. 2. For $\ell = 5$, $T_i = T_j = 10$, $S = 17$, the sampling experiment is as shown above. The green and red balls are the real values for the responses c_i and c_j, respectively. On the left, the sample (in dashed lines) becomes the non-real values for the response c_i. The white balls in the left sample are blue in the experiment on the right. Then the sample on the right are the non-real values for the response c_j. The total overlap is the set of red, green, and blue balls that are common between the two responses.

First, we select the encrypted values that will be a part of the encrypted response c_i. Since Σ is perfectly correct, all ℓ real encrypted values from the list $\mathsf{MM}(l_i)$ have to be part of the encrypted response c_i. Then there remain $T_i - \ell$ (non-real) encrypted values that must be part of the response.

Let B be a bag with ℓ red marbles and $S - 2\ell$ white marbles. (The red marbles represent the ℓ real

encrypted values for c_j, while the white marbles represent encrypted values in $\mathsf{EMM} \setminus (c_i \cup c_j)$.)

Choosing the non-real encrypted values for c_i is equivalent to choosing a random sample of size $T_i - \ell$ (without replacement) from B; let X be the number of red marbles in the sample, and so $T_i - \ell - X$ is the number of white marbles in the sample. (The red marbles in the sample represent the real encrypted values for c_j that are randomly chosen to "cover" part of c_i.)

Next, we sample the encrypted values that will be part of the encrypted response c_j. Similar to the previous case, the ℓ real values corresponding to $\mathsf{MM}(l_j)$ will be part of the response c_j, and the remaining $(T_j - \ell)$ will be sampled.

Then, let B' be a bag with ℓ green marbles, $T_i - \ell - X$ blue marbles, and $S - \ell - T_i + X$ white marbles. (The green marbles are the real encrypted values for c_i, the blue ones are non-real encrypted values for c_i which are not real values for c_j, and the white marbles are all other encrypted values in $\mathsf{EMM} \setminus (c_i \cup c_j)$.)

Choosing the non-real encrypted values for c_j is now equivalent to choosing a random sample of size $T_j - \ell$ (without replacement) from B'; let X' be the number of green or blue marbles in the sample.

(The green marbles in the sample are those that are real for c_i, and the blue ones are those that are non-real for c_i.)

Given both the above samples, the sum $X + X'$ represents the number of overlaps between c_i and c_j. The expectation of $X + X'$ can be expressed as follows:

$$
\begin{aligned}
&\mathbb{E}[X + X'] \\
&= \mathbb{E}\left[\mathbb{E}_X[X] + \mathbb{E}_X[X']\right] && (2)\\
&= \mathbb{E}\left[\frac{(T_i - \ell)\ell}{S - \ell} + \sum_{x=0}^{\ell} \Pr[X = x] \frac{(T_j - \ell)(T_i - x)}{S - \ell}\right] && (3)\\
&= \mathbb{E}\left[\frac{T_j - \ell}{S - \ell}\left(\frac{(T_i - \ell)\ell}{T_j - \ell} + \sum_{x=0}^{\ell} \Pr[X = x](T_i - x)\right)\right] \\
&= \mathbb{E}\left[\frac{T_j - \ell}{S - \ell}\left(\frac{(T_i - \ell)\ell}{T_j - \ell} + T_i - \sum_{x=0}^{\ell} \Pr[X = x] \cdot x\right)\right] \\
&= \mathbb{E}\left[\frac{T_j - \ell}{S - \ell}\left(\frac{(T_i - \ell)\ell}{T_j - \ell} + T_i - \frac{(T_i - \ell)\ell}{S - \ell}\right)\right]. && (4)
\end{aligned}
$$

Equation (2) is true from the linearity of expectation. (3) and (4) follow from plugging in the expression of the mean value, $\frac{nK}{N}$, of a hypergeometric distribution with parameters: total population size N, total number K of success states in population, and sample size n.

From (4), $\mathbb{E}[X + X']$ is equal to:

$$
\begin{aligned}
&= \mathbb{E}\left[\frac{\ell T_i - \ell^2}{S - \ell} + \frac{T_i T_j - \ell T_i}{S - \ell} - \frac{(T_i - \ell)(T_j - \ell)\ell}{(S - \ell)^2}\right] \\
&= \frac{\mathbb{E}[T_i T_j]}{S - \ell} - \frac{\ell \mathbb{E}[T_i T_j]}{(S - \ell)^2} + \frac{2\ell^2 \mathbb{E}[T_i]}{(S - \ell)^2} - \frac{\ell^2 S}{(S - \ell)^2} \\
&= \frac{(S - 2\ell)\,\mathbb{E}[T_i]\,\mathbb{E}[T_j]}{(S - \ell)^2} + \frac{2\ell^2 \mathbb{E}[T_i]}{(S - \ell)^2} - \frac{\ell^2 S}{(S - \ell)^2}
\end{aligned}
$$

Since T_i, T_j are sampled from a fixed distribution with bounded expectation $\mathbb{E}[T]$, it follows that $\mathbb{E}[T_i]\mathbb{E}[T_j] = \mathbb{E}[T]^2$ is also a bounded constant. (In fact, we only require that T_i, T_j are sampled from fixed distributions with the same expectation; the distributions need not be identical.) Thus, $\mathbb{E}[X + X']$ varies non-negligibly as a function of ℓ, and so Σ cannot be minimally-leaking since overlaps are visible to the adversary. □

Interestingly, Patel et al.'s dprfMM *is* minimally-leaking despite being almost randomly-overlapping; at setup, the scheme selects $2(t - \ell)$ random overlaps for a response of length ℓ. However, the total storage space is fixed, and any encrypted value that cannot be stored due to collisions is moved to a client stash. (See Patel et al.'s paper for more details on how this scheme works [30].) Theorem 1 does not apply here since because of the stash. In fact, the theorem implies that the stash is necessary; modifying the scheme so that it becomes randomly-overlapping, for example, by taking a Las Vegas-style approach to setup, would not be secure.

What about the other scheme that suppresses query volumes, namely, AVLH [17]? During setup, AVLH organizes the encrypted values into random groups: For each label l, a random size-$|\mathsf{MM}(l)|$ sample X is chosen from the set $\mathcal{X}$ of all bins. Each real encrypted value is placed into a bin in X so that each bin contains exactly one real encrypted value. Then, a second sample Y of size $t - |\mathsf{MM}(l)|$ is chosen from the remaining bins $\mathcal{X} \setminus X$. When the label l is queried (later on), the server returns the contents (including dummy values) of all the bins in either X or Y. (For more details on this, we refer the reader to the original paper detailing AVLH [17].) It follows that AVLH is not randomly-overlapping because the non-real encrypted values in c_i are not independent of the non-real values in c_j. Even so, we show that it is in fact not volume-hiding [30] and so not minimally-leaking.[1]

1 We note here that AVLH preceded the volume-hiding definition, and thus no claim was made that the scheme satisfied the definition.

Theorem 2. *AVLH [17] is not minimally-leaking.*

Proof. Recall that N is the size of the multi-map, and t is the upper bound on the maximum response length. Let $B_1, \ldots, B_n$ be the n "bins."

Let MM_0 be any multi-map such that $|\mathsf{MM}_0(l_1)| = 0$, and $|\mathsf{MM}_0(l_2)| = t$.

Let MM_1 be any multi-map such that $|\mathsf{MM}_1(l_1)| = |\mathsf{MM}_1(l_2)| = \frac{t}{2}$, and for all $2 < i \leq m$, $|\mathsf{MM}_1(l_i)| = |\mathsf{MM}_0(l_i)|$.

First, consider what happens when the multi-map is MM_0. Recall that $\mathcal{E}$ is the set of all encrypted values in EMM. Let $\mathcal{C}$ be the random variable representing the set of real encrypted values that are either in c_1 or c_2. For each $i \in [n]$, let α_i be the random variable representing the number of encrypted values in $\mathcal{E} \setminus \mathcal{C}$ that are randomly mapped to the bin B_i during setup; and let $\alpha_{\max} \stackrel{\text{def}}{=} \max_{1 \leq i \leq n} \alpha_i$. For each $i \in [n]$, let $\beta_{0,i}$ be the random variable representing the number of encrypted values in $\mathcal{E}$ that are randomly mapped to the bin B_i during setup; and let $\beta_{0,\max} \stackrel{\text{def}}{=} \max_{1 \leq i \leq n} \beta_{0,i}$. Clearly, there is no chance that $\beta_{0,\max} - \alpha_{\max}$ is greater than one.

Compare this with the scenario in which the multi-map is MM_1. Let $(\alpha_1, \ldots, \alpha_n)$ be defined as before. By construction, this is distributed identically to the tuples when the multi-map is MM_0. For each $i \in [n]$, let $\beta_{1,i}$ be the random variable representing the number of encrypted values in $\mathcal{E}$ that are randomly mapped to the bin B_i during setup; and let $\beta_{1,\max} \stackrel{\text{def}}{=} \max_{1 \leq i \leq n} \beta_{1,i}$. Here, there is a non-negligible probability that $\beta_{1,\max} - \alpha_{\max} = 2$; this occurs whenever there exists a bin B_i such that $\alpha_i = \alpha_{\max}$, and a real encrypted value for c_1 and a real encrypted value for c_2 are both randomly mapped to the bin B_i. In other words, $\Pr[\beta_{1,\max} - \alpha_{\max} = 2] \geq \left(\frac{t}{2n}\right)^2 = \mathsf{nonnegl}(\lambda)$.

Since AVLH pads the bins to the size of the largest bin, the adversary can distinguish between the scenario in which the multi-map is MM_0 from that when it is MM_1 with non-negligible advantage, just from the number of dummy values, which can be inferred from the storage. See Figure 3. □

Our results imply that the setup leakage of AVLH would have to include the size of the bins, which is not entirely independent of the input multi-map. It might be possible to set a constant bin size and use a Las Vegas approach to setup in AVLH in order to make it minimally-leaking. However, there arises the question of how to pick the bin size. Our proof shows that for any pair of multi-maps with only two differing volumes, the setup will create bins of differing sizes with non-negligible probability. Therefore, the bin size must account for every such pair of multi-maps in order to be minimally-leaking.

Remark 1. *Note that while dprfMM is similar to AVLH in that it uses overlaps to reduce storage, it has a fixed storage size on the server regardless of the underlying multi-map. Any value that "overflows" is stored in the client-side stash; this makes it possible for the overlaps observable on the server-side to be both independent of response lengths and identically distributed for all multi-maps.*

5 Bounds on Sampled Minimally-leaking STE schemes

Recall that the adversary's view includes the encrypted multi-map EMM, which is the output of the scheme's setup algorithm, as well as the server's states, messages, and computations during querying.

In the "sampled" security game, the adversary does not choose the queries. Instead, the challenger picks the queries which are "encrypted" into search tokens and, thus, hidden from the adversary. It is worth noting that it may be the case that the adversary never learns which label corresponds to which encrypted query. Still, the adversary might glean private information: for example, after some queries occur, the adversary can order the encrypted responses in ascending order and count the number of overlaps between the ordered encrypted responses. Recall that an overlap means that the encrypted value is also part of another response (Definition 6). So, the overlap between the i^{th} shortest response $\mathcal{E}_i$ and the j^{th} shortest response $\mathcal{E}_j$ is the set of encrypted values that are part of both $\mathcal{E}_i$ and $\mathcal{E}_j$.

If Σ is sampled minimally-leaking, it follows that the adversary's view when the scheme is run on multi-map MM_0 is indistinguishable to that when run on a different multi-map MM_1 with the same "dimensions" (i.e., size and upper bound on the maximum response length). Specifically, the observable statistics on overlaps when the input is MM_0 ought to be indistinguishable from the same statistics when the input is MM_1 instead. We will use this fact to show that overlapping doesn't help to reduce storage by very much even in the weaker "sampled" security setting; we prove a tight lower bound on read efficiency for sampled minimally-leaking encrypted multi-map schemes (Theorem 3 in Section 5.1) and a lower

bound on storage for sampled minimally-leaking and optimally read-efficient encrypted multi-map schemes (Theorem 4 in Section 5.2).

To do this formally, we first introduce the experiment, $\mathsf{SMLExp}_{\mathsf{MM}}(1^\lambda)$, below and describe the setting for all the theorem and lemma statements in this section: Let $\mathbb{L} = \{l_1, \ldots, l_m\}$ be the label space.

Consider the following experiment, $\mathsf{SMLExp}_{\mathsf{MM}}(1^\lambda)$, parametrized by the security parameter 1^λ and the multi-map MM:

1. Run the encrypted multi-map scheme's setup algorithm Setup on the multi-map MM to obtain the key K and the encrypted multi-map EMM, i.e., $(K, \mathsf{EMM}) \leftarrow \mathsf{Setup}(1^\lambda, \mathsf{MM})$.
2. For each label $l_i \in \mathbb{L}$:
   i. Run Σ's token algorithm Token on the key K and the label l_i; let τ_i be the output from running Token.
   ii. Then, run Σ's query algorithm Query on the encrypted multi-map EMM (from step 1) and the output τ_i to get the the encrypted response c_i, i.e., $\tau_i \leftarrow \mathsf{Token}(K, l_i)$ and $c_i \leftarrow \mathsf{Query}(\mathsf{EMM}, \tau_i)$.
3. Recall that each encrypted response c_i is a set of encrypted values. Order the encrypted responses from smallest to largest set; ties are broken by the value of the smallest encrypted value in the response; let $\mathcal{E}_1, \ldots, \mathcal{E}_m$ denote this ordered list so that $\mathcal{E}_i$ is the i^{th} shortest encrypted response. (Note: $\mathcal{E}_i = c_j$ for some j that is not necessarily i.)
4. Output $(K, \mathcal{E}_1, \ldots, \mathcal{E}_m)$.

Let MM be the multi-map; and let N be the size of MM, and let t be the upper bound on the maximum response length. Let $J \stackrel{\text{def}}{=} m - \left\lfloor \frac{N}{t} \right\rfloor + 1$. Let LHS (short for "lefthand-side") be the range $[1, \ldots, J-1]$, and let RHS (short for "righthand-side") be the range $[J, \ldots, m]$. (See Figure 1 in the introduction for a picture of what we mean by the lefthand-side and righthand-side.)

Let $\Sigma = (\mathsf{Setup}, \mathsf{Token}, \mathsf{Query}, \mathsf{Resolve})$ be any encrypted multi-map scheme for the label space $\mathbb{L}$ that is correct and sampled minimally-leaking. Let λ denote the security parameter for Σ.

In all of the theorems and lemmas below, let $(K, \mathcal{E}_1, \ldots, \mathcal{E}_m) \leftarrow \mathsf{SMLExp}_{\mathsf{MM}}(1^\lambda)$ be the output from running $\mathsf{SMLExp}_{\mathsf{MM}}(1^\lambda)$ (above). For every pair of indices $i, j \in [m]$, let the overlap between the i^{th} shortest response $\mathcal{E}_i$ and the j^{th} shortest response $\mathcal{E}_j$, denoted $\mathsf{OL}_{\mathsf{MM}}(i, j)$, be the encrypted values that show up in both $\mathcal{E}_i$ and $\mathcal{E}_j$, i.e., $\mathsf{OL}_{\mathsf{MM}}(i, j) = \mathcal{E}_i \cap \mathcal{E}_j$. For *an encrypted value* e and an index $i \in [m]$, let $\mathsf{OL}_{\mathsf{MM}}(e, i)$ denote the intersection between the singleton $\{e\}$ and $\mathcal{E}_i$, i.e., $\mathsf{OL}_{\mathsf{MM}}(e, i) = \{e\} \cap \mathcal{E}_i$.

5.1 Optimal Read-efficiency

Let the "*read-efficiency curve*," denoted $\mathsf{RE} : [m] \mapsto \mathbb{N}$, be the following function mapping from the set $[m]$ of indices to the set of non-negative integers,

$$\mathsf{RE}(i) = \begin{cases} \left\lfloor \frac{N}{m-i+1} \right\rfloor, & \text{for } i < J \\ t, & \text{for } J \leq i \leq m. \end{cases} \tag{5}$$

Intuitively, the curve tightly bounds all possible multi-maps with the parameters N, m, and t. See Figure 1 for an example. Here, we show that the necessary and sufficient number of bits that the server reads is dictated by the read-efficiency curve as follows:

Theorem 3. *With overwhelming probability, for every $i \in [m]$, the length of the i^{th} shortest encrypted response, denoted $|\mathcal{E}_i|$, is at least $\mathsf{RE}(i)$.*

To prove Theorem 3, we show that for each i, there exists a multi-map whose i^{th} shortest (unencrypted) response is of length $\mathsf{RE}(i)$.

Proof. Let I be the set of indices i in the set $\{2, \ldots, m\}$ such that $\mathsf{RE}(i) > \mathsf{RE}(i-1)$; that is, $I \stackrel{\text{def}}{=} \{i \in \{2, \ldots, m\} \mid \mathsf{RE}(i) > \mathsf{RE}(i-1)\}$. Let $\mathcal{I} = I \cup \{1\}$. For each $i \in \mathcal{I}$, let MM_i be a multi-map where for each $j \in \{i, \ldots, m\}$, the response $|\mathsf{MM}(l_j)| \geq \mathsf{RE}(i) = \left\lfloor \frac{N}{m-i+1} \right\rfloor$, i.e.,

$$\mathsf{MM}_i(j) \geq \begin{cases} 0, & \text{for } j < i \\ \mathsf{RE}(i), & \text{for } i \leq j \leq m; \end{cases}$$

i.e., MM_i is a "rectangular" multi-map with all tuple lengths either $\mathsf{RE}(i)$ or 0. Let $(\mathsf{MM}_i, \mathcal{E}_{\mathsf{MM}_i,1}, \ldots, \mathcal{E}_{\mathsf{MM}_i,m})$ be the result of running $\mathsf{SMLExp}_{\mathsf{MM}_i}$. The sequence of encrypted responses $(\mathcal{E}_{\mathsf{MM}_i,1}, \ldots, \mathcal{E}_{\mathsf{MM}_i,m})$ is part of the adversarial view. Note that $\mathcal{E}_{\mathsf{MM}_i,j}$ does not necessarily correspond to the encrypted response for query l_j, and that the adversary is not told which encrypted response is the result of which label. However, the adversary can deduce that the longer encrypted responses (those that are at least of length $\mathsf{RE}(i)$) correspond to labels $l_i, \ldots, l_m$. This is because Σ is correct, and so every encrypted response must be at least as long as its decryption, and a response that is at least $\mathsf{RE}(i)$ values

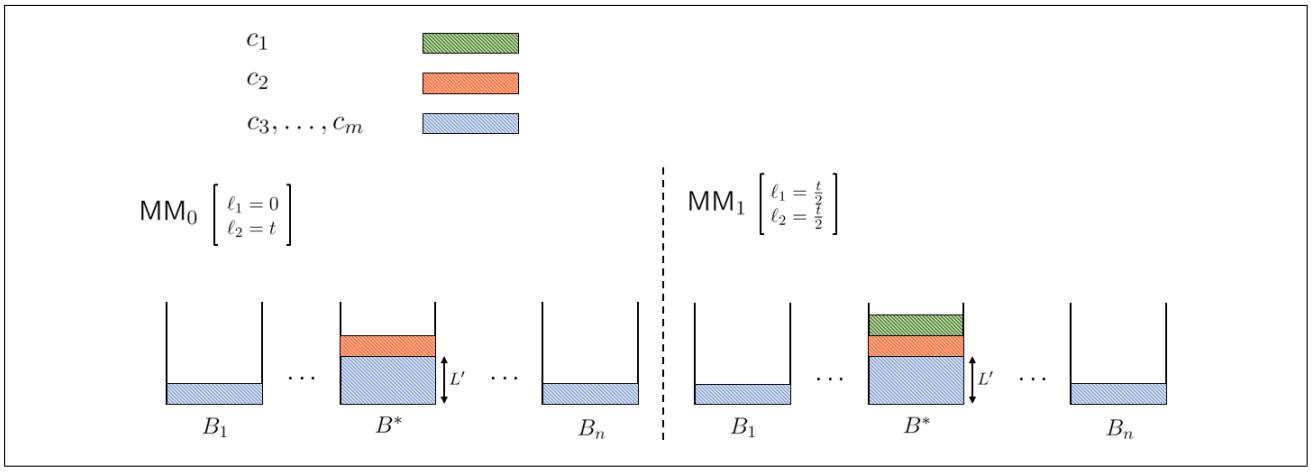


Fig. 3. For any one run of setup, let load L' of bin B^* be the maximum load after values from labels $l_3, \ldots, l_m$ have been mapped to bins and let L^* be the maximum load after all the values have been mapped to bins. On the left, we show the maximum load distribution for MM_0. The maximum load L^* can only be either L' or $L'+1$ after mapping labels l_0, l_1. On the right, for MM_1, we see that L^* is $L'+2$ whenever both l_0 and l_1 are mapped to bin B^*, which happens with non-negligible probability.

long cannot "fit" into a shorter encrypted response. It follows that for all $j \in \{i, \ldots, m\}$, $|\mathcal{E}_{\mathsf{MM}_i, j}| \geq \mathsf{RE}(i)$.

Since Σ is sampled minimally-leaking, with overwhelming probability, for all $i \in [m]$, $|\mathcal{E}_i| \geq \mathsf{RE}(i)$. □

Remark 2. *The bound in Theorem 3 is tight since the encrypted multi-map scheme that pads the response lengths to the read-efficiency curve is correct and sampled minimally-leaking. See Appendix C for a formal description.*

5.2 Required Storage for Optimal Read-efficiency

The results in this section are for when Σ is correct, sampled minimally-leaking, and optimally read-efficient; in the sampled security setting, by *optimally read-efficient*, we mean that with probability 1, for all $i \in [m]$, $|\mathcal{E}_i| = \mathsf{RE}(i)$.

Lemma 1. *For every $i, j \in \mathsf{RHS}$, the probability that $|\mathsf{OL}_{\mathsf{MM}}(i, j)| > 0$ is negligible.*

Proof. Let $\mathsf{MM}_{\blacksquare}$ be a multi-map of size N and maximum response t such that for every $i \in \mathsf{RHS}$, the length of the response $\mathsf{MM}_{\blacksquare}(l_i)$ is t. Then $\mathsf{MM}_{\blacksquare}$ corresponds to the "rectangle" where all the real encrypted values are in righthand-side. i.e.,

$$|\mathsf{MM}_{\blacksquare}(l_i)| = t \qquad \forall i \in \mathsf{RHS}.$$

Since Σ is correct and optimally read-efficient,

$$\Pr\big[|\mathsf{OL}_{\mathsf{MM}_{\blacksquare}}(i, j)| > 0\big] = 0 \qquad \forall i, j \in \mathsf{RHS}.$$

Overlaps between encrypted responses are part of the adversary's view. Thus the fact that Σ is also sampled minimally-leaking implies that for any MM with the same size and upper bound on the maximum response length, for all $i, j \in \mathsf{RHS}$, $\Pr[|\mathsf{OL}_{\mathsf{MM}}(i, j)| > 0] = \mathsf{negl}(\lambda)$. □

From Lemma 1, we also get the additional lemma:

Lemma 2. *For every $i \in \mathsf{LHS}$, for every encrypted value $e \in \mathcal{E}_i$, the probability that $\sum_{j \in \mathsf{RHS}} |\mathsf{OL}_{\mathsf{MM}}(e, j)| > 1$ is negligible.*

Proof. For the sake of reaching a contradiction, assume that there exists $i \in \mathsf{LHS}$ such that there exists $e \in \mathcal{E}_i$ such that $\Pr\left[\sum_{j \in \mathsf{RHS}} |\mathsf{OL}_{\mathsf{MM}}(e, j)| > 1\right] = \mathsf{nonnegl}(\lambda)$. For each $j \in \mathsf{RHS}$, $|\mathsf{OL}_{\mathsf{MM}}(e, j)|$ is either zero or one; otherwise, Σ wouldn't be correct. Thus, the probability that there exist two indices $j, j' \in \mathsf{RHS}$ such that $|\mathsf{OL}_{\mathsf{MM}}(e, j)| = |\mathsf{OL}_{\mathsf{MM}}(e, j')| = 1$ is non-negligible. This implies that the responses for j and j' overlap in the encrypted value e with non-negligible probability, which contradicts Lemma 1. □

We will make use of Lemma 2 to prove the main lemma, Lemma 3, below. For each $i \in [m]$, let ℓ_i denote the length of the result of running Σ's $\mathsf{Resolve}$ algorithm on input the key K and the encrypted response $\mathcal{E}_i$, i.e., $\ell_i \stackrel{\text{def}}{=} |\mathsf{Resolve}(K, \mathcal{E}_i)|$. In the following results, we assume that the upper bound on the maximum response length

t evenly divides the total number of values N in the multi-map MM i.e. $\sum_{i\in\mathsf{RHS}} \mathsf{RE}(i) = N$.

An implication of Lemma 1 is that every non-real value in the righthand-side must be covered by a (real or dummy) value in the lefthand-side. That is, if e is a non-real value for $\mathcal{E}_i$ for $i \in \mathsf{RHS}$, then there exists $j \in \mathsf{LHS}$ such that e is real only for $\mathcal{E}_j$, or e is a dummy record. Let the STE be *balanced* if values in the righthand-side are covered by values in the lefthand-side, and vice versa.

Lemma 3. *Let Σ be balanced, and let N, m, and t be such that $\sum_{i\in\mathsf{RHS}} \mathsf{RE}(i) = N$. Let $\eta \stackrel{def}{=} \sum_{i\in\mathsf{LHS}} \mathsf{RE}(i) - N$. For each $d \in \{0,\ldots,\eta\}$, when Σ's setup algorithm incorporates a total of d dummy values into the encrypted multi-map, with overwhelming probability,*

$$\sum_{i\in\mathsf{LHS}} \sum_{j\in\mathsf{RHS}} |\mathsf{OL}^{\rightarrow}_{\mathsf{MM}}(j,i)| \geq \sum_{i\in\mathsf{LHS}} (\mathsf{RE}(i) - \ell_i) - d.$$

where $\mathsf{OL}^{\rightarrow}_{\mathsf{MM}}(j,k)$ denotes the real encrypted values in the encrypted response $\mathcal{E}_{\mathsf{MM}_i,j}$ that overlap the encrypted response $\mathcal{E}_{\mathsf{MM}_i,k}$, and $\ell_i \stackrel{def}{=} |\mathsf{Resolve}(K,\mathcal{E}_i)|$.

Proof. The proof is by cases. In the first case, setup adds no dummy values. In the second case, setup adds any number of dummy values between 1 and η.

Case 1: $d = 0$. Since there are no dummy values, the non-real encrypted values in any encrypted response have to be real encrypted values in some other encrypted response. In order to characterize these overlaps, we define a series of multi-maps MM_i. Intuitively, each MM_i is a multi-map such that the response lengths $|\mathsf{MM}(l_j)|$ for $j \geq i$ are all exactly $\mathsf{RE}(i)$. See Figure 4 for an example. A multi-map MM_i is defined for each "step" index i, where the value of $\mathsf{RE}(i)$ increases, as shown in the figure.

Formally, let I be the set of indices $i \in \{2,\ldots,m\}$ such that $\mathsf{RE}(i) > \mathsf{RE}(i-1)$. For each $i \in I$, we define the multi-map MM_i as follows. Let $f : [m] \mapsto \mathbb{N}$ be a fixed function such that for all $1 \leq j \leq m$, $f(j) \leq \mathsf{RE}(j)$, and $\sum_{1\leq j<i} f(j) = N - \sum_{i\leq j\leq m} \mathsf{RE}(i)$.

$$|\mathsf{MM}_i(j)| = \begin{cases} f(j), & \text{for } 1 \leq j < i \\ \mathsf{RE}(i), & \text{for } i \leq j \leq m. \end{cases}$$

We define the additional multi-map MM_1. Let $i_{\min}$ be the smallest value in I. Let $g : [m] \mapsto \mathbb{N}$ be a fixed function such that for all $1 \leq j \leq m$, $\mathsf{RE}(1) \leq g(j) \leq \mathsf{RE}(j)$, and $\sum_{i_{\min}\leq j\leq m} g(j) = N - \sum_{1\leq j<i_{\min}} \mathsf{RE}(j)$.

$$|\mathsf{MM}_1(j)| = \begin{cases} \mathsf{RE}(j), & \text{for } j < i_{\min} \\ g(j), & \text{for } i_{\min} \leq j \leq m. \end{cases}$$

For each $i \in I$, consider the multi-map MM_i:

Let $(K_{\mathsf{MM}_i}, \mathcal{E}_{\mathsf{MM}_i,1}, \ldots, \mathcal{E}_{\mathsf{MM}_i,m}) \leftarrow \mathsf{SMLExp}_{\mathsf{MM}_i}(1^\lambda)$. For each index $j \in [m]$, let $\mathcal{R}_{\mathsf{MM}_i,j} \subseteq \mathcal{E}_{\mathsf{MM}_i,j}$ denote the part of $\mathcal{E}_{\mathsf{MM}_i,j}$ that when decrypted equals the response $\mathsf{Resolve}(K_{\mathsf{MM}_i}, \mathcal{E}_i)$; these are the real encrypted values in $\mathcal{E}_{\mathsf{MM}_i,j}$. For indices $j,k \in [m]$, let $\mathsf{OL}^{\rightarrow}_{\mathsf{MM}}(j,k)$ be the encrypted values in $\mathcal{R}_{\mathsf{MM}_i,j}$ that "cover" part of $\mathcal{E}_{\mathsf{MM}_i,k}$, i.e., $\mathsf{OL}^{\rightarrow}_{\mathsf{MM}_i}(j,k) = \mathcal{R}_{\mathsf{MM}_i,j} \cap \mathcal{E}_{\mathsf{MM}_i,k}$.

Since Σ is correct, for any $i \leq j \leq m$, the response to l_j has to be long enough to hold all the $\mathsf{RE}(i)$ values. Therefore, the response to l_j cannot be one of the $(i-1)$ shortest responses. This implies that for every $i \leq j \leq m$, the j^{th} shortest encrypted response $\mathcal{E}_{\mathsf{MM}_i,j}$ includes exactly $\mathsf{RE}(i)$ real encrypted values.

From Lemma 1, every non-real encrypted value in the righthand-side must be "covered by" a real encrypted value in the lefthand-side. That is, with overwhelming probability, for every $e \in \mathcal{E}_{\mathsf{MM}_i,j}$ such that $j \in \mathsf{RHS}$ and $e \notin \mathcal{R}_{\mathsf{MM}_i,j}$, there exists $k \in \mathsf{LHS}$ and $e' \in \mathcal{R}_{\mathsf{MM}_i,k}$ such that $e = e'$. The non-real encrypted value must be covered by something, and it cannot be covered by a real encrypted value in the righthand-side because this would violate Lemma 1.

Moreover, from Lemma 2, the covered encrypted value is *unique*; a real encrypted value from the lefthand-side cannot cover two non-real encrypted values in the righthand-side because this would violate Lemma 2.

From geometry and our assumption that $\sum_{i\in\mathsf{RHS}} \mathsf{RE}(i) = N$, the number of real encrypted values in the lefthand-side equals the number of non-real encrypted values in the righthand-side; both of these quantities sum to $N - \sum_{j\in\mathsf{RHS}} \mathcal{R}_{\mathsf{MM}_i,j}$. It follows that each of the real encrypted values in the lefthand-side covers a distinct non-real encrypted value in the righthand-side. Each $i \leq j < J$ has exactly $\mathsf{RE}(i)$ real encrypted values, and so, with overwhelming probability, the total overlap with the righthand-side $\sum_{k\in\mathsf{RHS}} |\mathsf{OL}_{\mathsf{MM}_i}(j,k)| \geq \mathsf{RE}(i)$.

Using a similar argument and multi-map MM_1, we can show that with overwhelming probability, for each $1 \leq j < i_{\min}$, $\sum_{k\in\mathsf{RHS}} |\mathsf{OL}_{\mathsf{MM}_1}(j,k)| \geq \mathsf{RE}(j)$.

Since Σ is sampled minimally-leaking and the number of overlaps are observable by the adversary, the overlaps for the multi-maps MM_i must be indistinguishable from the overlaps for any other multi-map MM with the same parameters N, m, and t. It follows that, with overwhelming probability, regardless of the multi-map, for all $i \in \mathsf{LHS}$, $\sum_{j\in\mathsf{RHS}} |\mathsf{OL}_{\mathsf{MM}}(i,j)| \geq \mathsf{RE}(i)$. Since for any MM and $i \in \mathsf{LHS}$, the i^{th} response only has ℓ_i real encrypted values, this implies that the rest of the over-

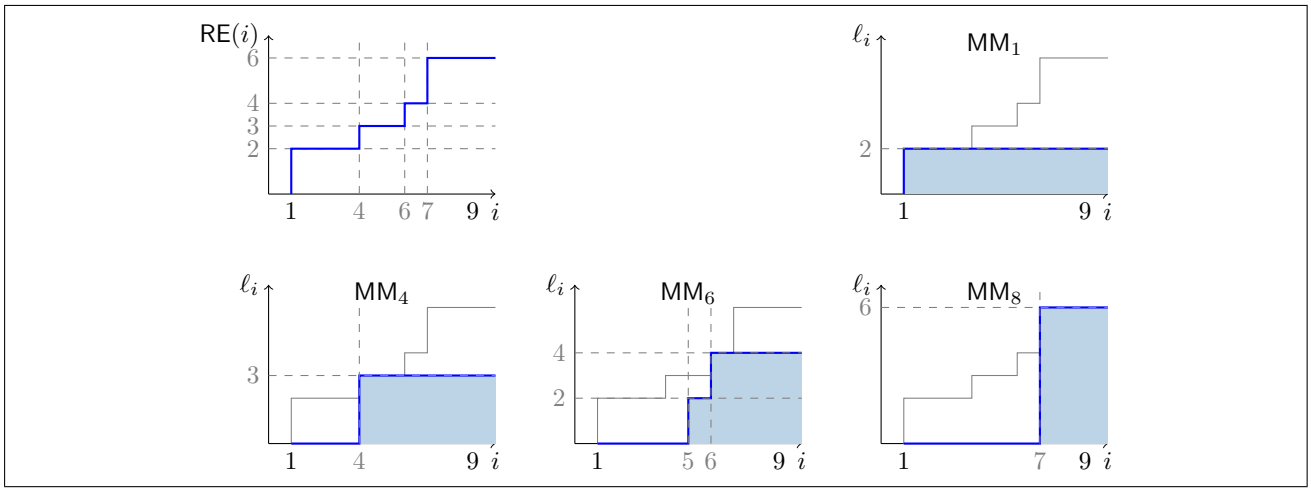


Fig. 4. A series of multi-maps $\{\mathsf{MM}_i\}$ in blue for $N = 18$, $m = 9$, $t = 6$. Each multi-map MM_i is defined at one of the step indices i such that $\mathsf{RE}(i) > \mathsf{RE}(i-1)$.

lap must be covered by real encrypted values from the righthand-side. Then with overwhelming probability, $\sum_{i\in\mathsf{LHS}}\sum_{j\in\mathsf{RHS}}|\mathsf{OL}^{\rightarrow}_{\mathsf{MM}}(j,i)| \geq \sum_{i\in\mathsf{LHS}}(\mathsf{RE}(i)-\ell_i)$.

Case 2: $1 \leq d \leq \eta$**.** When a dummy is added to the righthand-side, the number of non-real encrypted values on the lefthand-side that need to be covered by something in the righthand-side remains the same. However, when a dummy is added to the lefthand-side, the number of non-real encrypted values in the lefthand-side decreases by one. Then the total number of non-real encrypted values in the lefthand-side that must be covered by the righthand-side decreases *at most* by the number of dummies d. Thus, by a similar argument to that of case 1, it follows that for each $1 \leq d \leq \eta$, when Σ's setup algorithm incorporates a total of d dummy values into the encrypted multi-map, with overwhelming probability, $\sum_{i\in\mathsf{LHS}}\sum_{j\in\mathsf{RHS}}|\mathsf{OL}^{\rightarrow}_{\mathsf{MM}}(j,i)| \geq \sum_{i\in\mathsf{LHS}}(\mathsf{RE}(i)-\ell_i)-d$.

This completes our proof of Lemma 3. □

We are now ready to prove the main result of this section:

Theorem 4. *Let Σ be balanced, and let $\sum_{i\in\mathsf{RHS}}\mathsf{RE}(i) = N$ where N is the size of the multi-map. With overwhelming probability, Σ's setup algorithm adds at least $\sum_{i\in\mathsf{LHS}}\mathsf{RE}(i)-N$ dummy values to the encrypted structure.*

Proof. For the sake of reaching a contradiction, assume that with non-negligible probability, the setup algorithm adds d dummy values where d is strictly less than $\eta \stackrel{\text{def}}{=} \sum_{i\in\mathsf{LHS}}\mathsf{RE}(i) - N$.

From Lemma 3, with overwhelming probability,

$$\begin{aligned}
&\sum_{i\in\mathsf{LHS}}\sum_{j\in\mathsf{RHS}}|\mathsf{OL}^{\rightarrow}_{\mathsf{MM}}(j,i)| \\
&\geq \sum_{i\in\mathsf{LHS}}(\mathsf{RE}(i)-\ell_i)-d \\
&> \sum_{i\in\mathsf{LHS}}\mathsf{RE}(i) - \left(N - \sum_{i\in\mathsf{RHS}}\ell_i\right) - \left(\sum_{i\in\mathsf{LHS}}\mathsf{RE}(i) - N\right) \\
&= \sum_{i\in\mathsf{RHS}}\ell_i. \qquad (6)
\end{aligned}$$

From (6), there are more non-real encrypted values in the lefthand-side that are covered by a real encrypted file in the righthand-side than there are real encrypted values in the righthand-side. This implies that there exists a real encrypted value in the righthand-side that covers two encrypted values e_1 and e_2 in the lefthand-side. However, since Σ is balanced, this would reveal to the adversary that e_1 and e_2 are non-real encrypted values, and Σ would not be sampled minimally-leaking.

It follows that $s \geq (\eta + N)/N = \sum_{i\in\mathsf{LHS}}\mathsf{RE}(i)/N$. □

5.3 Experiments

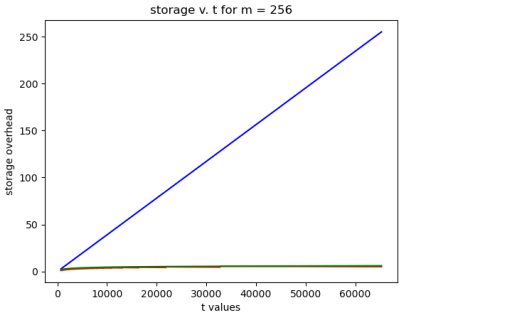
We now present a brief experimental analysis of our bounds on the storage efficiency of optimally read-efficient minimally-leaking schemes. At a high level, we look to answer the following questions: (i) is our lower bound in the sampled setting, which only applies to balanced schemes, too restrictive? and, (ii) is there a significant difference between the storage lower bounds in

the standard and sampled settings for minimally-leaking STE schemes?

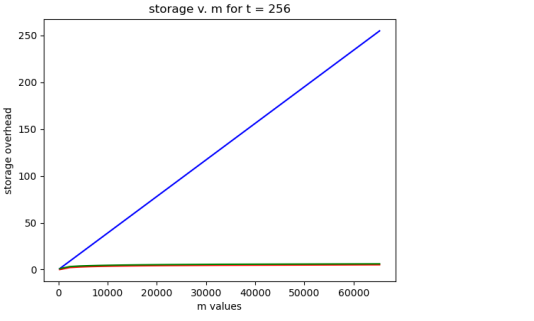
For (i), we compare our lower bound to the naive upper bound scheme that pads every encrypted response to the read-efficiency curve. Our experimental results show that our lower bound is very close to the upper bound for the sampled setting. This suggests that any tighter upper or lower bound would have to utilize substantially more complicated techniques.

For (ii), we see, as expected, that the lower bound on storage for the standard setting dominates the lower bound for the sampled setting. We conclude that the sampled setting does indeed allow for significantly more storage-efficient schemes than in the standard STE model.

The results for both (i) and (ii) are more pronounced for larger values of N, m, and t. We show our results for $N = 65536$ in Figure 5 and refer the interested reader to Appendix C for further details.



(a) $N = 65536, m = \sqrt{N}$



(b) $N = 65536, t = \sqrt{N}$

Fig. 5. $N = 65536$, and $m = \sqrt{N}$ or $t = \sqrt{N}$. In each plot, the blue curve represents the lower bound on storage overhead for the minimally-leaking setting. The green curve represents the storage overhead of the naive sampled minimally-leaking scheme, and the red curve represents our lower bound for the sampled minimally-leaking setting.

6 Conclusion and Future Work

This paper presents the first storage and read efficiency bounds for STE schemes for multi-maps that leak at most the query equality pattern.

While overlapping can significantly reduce the required storage (e.g., dprfMM [30] does this), this is sometimes impossible and, in general, tricky to achieve. For a correct (and stash-less) encrypted multi-map scheme Σ: if Σ is minimally-leaking and optimally read-efficient, it is impossible; if Σ is sampled minimally-leaking and optimally read-efficient scheme, it is impossible using a balanced technique; and if Σ is randomly-overlapping, it cannot be minimally-leaking. Currently, the only known minimally-leaking scheme that is much more efficient than pad-to-max is dprfMM that stores some of the answers on the client-side. This stash is crucial to both security and correctness of the scheme, as we discussed in Section 4.

We conjecture that the problem of arranging the overlaps to find a solution with minimal storage costs in a correct, stash-less, and minimally-leaking manner may be computationally intractable. Even finding and analyzing a non-optimal solution may be difficult. We leave this as an open problem for future research.

7 Acknowledgements

We would like to thank the anonymous reviewers for their helpful feedback. This work has been partially supported by a grant from the MITRE corporation.

References

[1] Gilad Asharov, Moni Naor, Gil Segev, and Ido Shahaf. Searchable symmetric encryption: optimal locality in linear space via two-dimensional balanced allocations. In Daniel Wichs and Yishay Mansour, editors, *48th ACM STOC*, pages 1101–1114. ACM Press, June 2016.

[2] Gilad Asharov, Gil Segev, and Ido Shahaf. Tight tradeoffs in searchable symmetric encryption. In Shacham and Boldyreva [31], pages 407–436.

[3] Laura Blackstone, Seny Kamara, and Tarik Moataz. Revisiting leakage abuse attacks. In *27th Annual Network and Distributed System Security Symposium, NDSS 2020, San Diego, California, USA, February 23-26, 2020*. The Internet Society, 2020.

[4] Elette Boyle and Moni Naor. Is there an oblivious ram lower bound? In *Proceedings of the 2016 ACM Conference on*

Innovations in Theoretical Computer Science, ITCS '16, page 357–368, New York, NY, USA, 2016. Association for Computing Machinery.

[5] David Cash, Paul Grubbs, Jason Perry, and Thomas Ristenpart. Leakage-abuse attacks against searchable encryption. In Indrajit Ray, Ninghui Li, and Christopher Kruegel, editors, *ACM CCS 2015*, pages 668–679. ACM Press, October 2015.

[6] David Cash, Paul Grubbs, Jason Perry, and Thomas Ristenpart. Leakage-abuse attacks against searchable encryption. In *Proceedings of the 22nd ACM SIGSAC conference on computer and communications security*, pages 668–679, 2015.

[7] David Cash and Stefano Tessaro. The locality of searchable symmetric encryption. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 351–368. Springer, 2014.

[8] David Cash and Stefano Tessaro. The locality of searchable symmetric encryption. In Phong Q. Nguyen and Elisabeth Oswald, editors, *EUROCRYPT 2014*, volume 8441 of *LNCS*, pages 351–368. Springer, Heidelberg, May 2014.

[9] Melissa Chase and Seny Kamara. Structured encryption and controlled disclosure. In Masayuki Abe, editor, *ASIACRYPT 2010*, volume 6477 of *LNCS*, pages 577–594. Springer, Heidelberg, December 2010.

[10] Reza Curtmola, Juan A. Garay, Seny Kamara, and Rafail Ostrovsky. Searchable symmetric encryption: improved definitions and efficient constructions. In Ari Juels, Rebecca N. Wright, and Sabrina De Capitani di Vimercati, editors, *ACM CCS 2006*, pages 79–88. ACM Press, October / November 2006.

[11] Francesca Falzon, Evangelia Anna Markatou, David Cash, Adam Rivkin, Jesse Stern, and Roberto Tamassia. Full database reconstruction in two dimensions. In *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*, pages 443–460, 2020.

[12] Marilyn George, Seny Kamara, and Tarik Moataz. Structured encryption and dynamic leakage suppression. Springer-Verlag, 2021.

[13] Oded Goldreich and Rafail Ostrovsky. Software protection and simulation on oblivious rams. *Journal of the ACM (JACM)*, 43(3):431–473, 1996.

[14] Paul Grubbs, Marie-Sarah Lacharité, Brice Minaud, and Kenneth G. Paterson. Learning to reconstruct: Statistical learning theory and encrypted database attacks. In IEEE S&P 2019 [15], pages 1067–1083.

[15] *2019 IEEE Symposium on Security and Privacy*. IEEE Computer Society Press, May 2019.

[16] Mohammad Saiful Islam, Mehmet Kuzu, and Murat Kantarcioglu. Inference attack against encrypted range queries on outsourced databases. In *Proceedings of the 4th ACM conference on Data and application security and privacy*, pages 235–246, 2014.

[17] Seny Kamara and Tarik Moataz. Computationally volume-hiding structured encryption. In Yuval Ishai and Vincent Rijmen, editors, *EUROCRYPT 2019, Part II*, volume 11477 of *LNCS*, pages 183–213. Springer, Heidelberg, May 2019.

[18] Seny Kamara, Tarik Moataz, and Olga Ohrimenko. Structured encryption and leakage suppression. In Shacham and Boldyreva [31], pages 339–370.

[19] Georgios Kellaris, George Kollios, Kobbi Nissim, and Adam O'Neill. Generic attacks on secure outsourced databases. In Edgar R. Weippl, Stefan Katzenbeisser, Christopher Kruegel, Andrew C. Myers, and Shai Halevi, editors, *ACM CCS 2016*, pages 1329–1340. ACM Press, October 2016.

[20] Evgenios M. Kornaropoulos, Charalampos Papamanthou, and Roberto Tamassia. Data recovery on encrypted databases with k-nearest neighbor query leakage. In IEEE S&P 2019 [15], pages 1033–1050.

[21] Evgenios M. Kornaropoulos, Charalampos Papamanthou, and Roberto Tamassia. The state of the uniform: Attacks on encrypted databases beyond the uniform query distribution. Cryptology ePrint Archive, Report 2019/441, 2019. https://eprint.iacr.org/2019/441.

[22] Evgenios M. Kornaropoulos, Charalampos Papamanthou, and Roberto Tamassia. Response-hiding encrypted ranges: Revisiting security via parametrized leakage-abuse attacks. *IACR Cryptol. ePrint Arch.*, 2021:93, 2021.

[23] Marie-Sarah Lacharité, Brice Minaud, and Kenneth G. Paterson. Improved reconstruction attacks on encrypted data using range query leakage. In *2018 IEEE Symposium on Security and Privacy*, pages 297–314. IEEE Computer Society Press, May 2018.

[24] Kasper Green Larsen and Jesper Buus Nielsen. Yes, there is an oblivious RAM lower bound! In Hovav Shacham and Alexandra Boldyreva, editors, *CRYPTO 2018, Part II*, volume 10992 of *LNCS*, pages 523–542. Springer, Heidelberg, August 2018.

[25] Chang Liu, Liehuang Zhu, Mingzhong Wang, and Yu-An Tan. Search pattern leakage in searchable encryption: Attacks and new construction. *Inf. Sci.*, 265:176–188, may 2014.

[26] Evangelia Anna Markatou and Roberto Tamassia. Full database reconstruction with access and search pattern leakage. In *International Conference on Information Security*, pages 25–43. Springer, 2019.

[27] University of Glasgow. *Glasgow IR datasets*, (accessed January 07, 2021). http://ir.dcs.gla.ac.uk/resources/test_collections/.

[28] Simon Oya and Florian Kerschbaum. Hiding the access pattern is not enough: Exploiting search pattern leakage in searchable encryption. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 127–142. USENIX Association, August 2021.

[29] Sarvar Patel, Giuseppe Persiano, and Kevin Yeo. Lower bounds for encrypted multi-maps and searchable encryption in the leakage cell probe model. Springer-Verlag, 2020.

[30] Sarvar Patel, Giuseppe Persiano, Kevin Yeo, and Moti Yung. Mitigating leakage in secure cloud-hosted data structures: Volume-hiding for multi-maps via hashing. In Lorenzo Cavallaro, Johannes Kinder, XiaoFeng Wang, and Jonathan Katz, editors, *ACM CCS 2019*, pages 79–93. ACM Press, November 2019.

[31] Hovav Shacham and Alexandra Boldyreva, editors. *CRYPTO 2018, Part I*, volume 10991 of *LNCS*. Springer, Heidelberg, August 2018.

[32] Yupeng Zhang, Jonathan Katz, and Charalampos Papamanthou. All your queries are belong to us: The power of file-injection attacks on searchable encryption. In *25th USENIX Security*, pages 707–720, 2016.

[33] Zheguang Zhao, Seny Kamara, Tarik Moataz, and Stan Zdonik. Encrypted databases: From theory to systems. In *11th Conference on Innovative Data Systems Research, CIDR 2021, Virtual Event, January 11-15, 2021, Online Proceedings*. www.cidrdb.org, 2021.

A Simulation-based definitions

In this section, we provide formal simulation-based definitions that are equivalent to minimally-leaking (Definition 3) and sampled minimally-leaking (Definition 4).

Minimally-leaking. Let $\mathbb{L} = \{l_1, \ldots, l_m\}$ be the label/query space. The *leakage profile* $\mathcal{L} = (\mathcal{L}_\mathsf{S}, \mathcal{L}_\mathsf{Q})$ is a pair of functions: the setup leakage function $\mathcal{L}_\mathsf{S}$ and the query leakage function $\mathcal{L}_\mathsf{Q}$. It defines the information about the multi-map MM that is allowed to leak during setup and querying. The idea is that the adversary should not be able to tell whether it is interacting in the real setting in which the challenger is running the STE scheme's algorithms, or in the ideal setting in which the simulator knows only the setup leakage, $\mathcal{L}_\mathsf{S}(\mathsf{MM})$, and the query leakages, $\mathcal{L}_\mathsf{Q}(\mathsf{MM}, (q_1)), \ldots, \mathcal{L}_\mathsf{Q}(\mathsf{MM}, (q_1, \ldots, q_\ell))$.

Formally, we define security with respect to the following experiments.

The experiment $\mathsf{Real}^{\mathcal{A}}_{\Sigma}(1^\lambda)$ is parameterized by the security parameter 1^λ, the adversary $\mathcal{A}$, and the STE scheme $\Sigma = (\mathsf{Setup}, \mathsf{Token}, \mathsf{Query}, \mathsf{Resolve})$.

1. The adversary $\mathcal{A}$ chooses the multi-map MM and sends MM to the challenger $\mathcal{C}$.
2. The challenger $\mathcal{C}$ runs $\mathsf{Setup}(1^\lambda, \mathsf{MM})$ and sends the resulting encrypted multi-map EMM to the $\mathcal{A}$.
3. $\mathcal{A}$ picks a query q and sends it to $\mathcal{C}$.
4. $\mathcal{C}$ runs $\mathsf{Token}(K, q)$, where K is the key generated in step 1, and sends the resulting token τ to $\mathcal{A}$.
5. Steps 3-4 are repeated a polynomial (in λ) number of times.
6. Finally, $\mathcal{A}$ outputs a bit b.

The experiment $\mathsf{Ideal}^{\mathcal{A},\mathcal{S}}_{\mathcal{L}}(1^\lambda)$ is parameterized by the security parameter 1^λ, the adversary $\mathcal{A}$, the simulator $\mathcal{S}$, and the leakage profile $\mathcal{L} = (\mathcal{L}_\mathsf{S}, \mathcal{L}_\mathsf{Q})$.

1. The adversary $\mathcal{A}$ chooses the multi-map MM and sends MM to the challenger $\mathcal{C}$.
2. The challenger $\mathcal{C}$ computes the setup leakage $\mathcal{L}_\mathsf{S}(\mathsf{MM})$ from MM and sends just the leakage to the simulator $\mathcal{S}$. The simulator $\mathcal{S}$ computes the encrypted multi-map EMM and returns EMM to $\mathcal{A}$ (via $\mathcal{C}$).
3. $\mathcal{A}$ picks a query q and sends it to $\mathcal{C}$.
4. $\mathcal{C}$ computes the query leakage $\mathcal{L}_\mathsf{Q}(\mathsf{MM}, \vec{q})$ from MM and the sequence $\vec{q}$ of queries so far, and sends just the leakage to $\mathcal{S}$. $\mathcal{S}$ computes the token τ and send τ to $\mathcal{A}$.
5. Steps 3-4 are repeated a polynomial (in λ) number of times.
6. Finally, $\mathcal{A}$ outputs a bit b.

In the literature (e.g., [17, 18]), security is defined with respect to these experiments as follows:

Definition 10 ($\mathcal{L}$-security)**.** *The STE scheme Σ is $\mathcal{L}$-secure for the leakage profile $\mathcal{L} = (\mathcal{L}_\mathsf{S}, \mathcal{L}_\mathsf{Q})$ if there exists a p.p.t. simulator $\mathcal{S}$ such that every p.p.t. adversary $\mathcal{A}$ can distinguish whether it is interacting in the real setting $\mathsf{Real}^{\mathcal{A}}_{\Sigma}(1^\lambda)$, or in the ideal setting $\mathsf{Ideal}^{\mathcal{A},\mathcal{S}}_{\mathcal{L}}(1^\lambda)$, with only negligible advantage. Σ is adaptively $\mathcal{L}$-secure if in step 3 of the experiments, $\mathcal{A}$ chooses the queries adaptively; that is, $\mathcal{A}$ chooses the next query based on the encrypted multi-map and interactions from prior queries.*

Definition 11 (Simulation-based minimally-leaking)**.** *The STE scheme Σ is minimally-leaking if it is non-adaptively $(\mathcal{L}_\mathsf{S}, \mathcal{L}_\mathsf{Q})$-secure, where the setup leakage $\mathcal{L}_\mathsf{S}$ reveals only the size and the upper bound on the maximum response length of the multi-map, and the query leakage $\mathcal{L}_\mathsf{Q}$ reveals only the query equality pattern.*

Sampled minimally-leaking. Simulation-based, sampled minimally-leaking is defined with respect to the following modifications to $\mathsf{Real}^{\mathcal{A}}_{\Sigma}(1^\lambda)$ and $\mathsf{Ideal}^{\mathcal{A},\mathcal{S}}_{\mathcal{L}}(1^\lambda)$. The modifications are boxed for better readability:

The experiment $\mathsf{SampledReal}^{\mathcal{A}}_{\Sigma,W}(1^\lambda)$ is parameterized by the security parameter 1^λ, the adversary $\mathcal{A}$, the STE scheme $\Sigma = (\mathsf{Setup}, \mathsf{Token}, \mathsf{Query}, \mathsf{Resolve})$, and the query distribution W.

1. The adversary $\mathcal{A}$ chooses the multi-map MM and sends MM to the challenger $\mathcal{C}$.
2. The challenger $\mathcal{C}$ runs $\mathsf{Setup}(1^\lambda, \mathsf{MM})$ and sends the resulting encrypted multi-map EMM to the $\mathcal{A}$.
3. $\boxed{\mathcal{C} \text{ samples a query } q \leftarrow \mathbb{L}.}$
4. $\mathcal{C}$ runs $\mathsf{Token}(K, q)$, where K is the key generated in step 1, and sends the resulting token τ to $\mathcal{A}$.
5. Steps 3-4 are repeated a polynomial (in λ) number of times.
6. Finally, $\mathcal{A}$ outputs a bit b.

The experiment $\mathsf{SampledIdeal}^{\mathcal{A},\mathcal{S}}_{\mathcal{L},W}(1^\lambda)$ is parameterized by the security parameter 1^λ, the adversary $\mathcal{A}$, the simulator $\mathcal{S}$, the leakage profile $\mathcal{L} = (\mathcal{L}_\mathsf{S}, \mathcal{L}_\mathsf{Q})$, and the query distribution W.

1. The adversary $\mathcal{A}$ chooses the multi-map MM and sends MM to the challenger $\mathcal{C}$.
2. The challenger $\mathcal{C}$ computes the setup leakage $\mathcal{L}_\mathsf{S}(\mathsf{MM})$ from MM and sends just the leakage to the simulator $\mathcal{S}$. The simulator $\mathcal{S}$ computes the encrypted multi-map EMM and returns EMM to $\mathcal{A}$ (via $\mathcal{C}$).
3. $\boxed{\mathcal{C} \text{ samples a query } q \leftarrow \mathbb{L}.}$
4. $\mathcal{C}$ computes the query leakage $\mathcal{L}_\mathsf{Q}(\mathsf{MM}, q)$ from MM and q and sends just the leakage to $\mathcal{S}$. $\mathcal{S}$ computes the token τ and send τ to $\mathcal{A}$.
5. Steps 3-4 are repeated a polynomial (in λ) number of times.
6. Finally, $\mathcal{A}$ outputs a bit b.

Definition 12 (W-sampled $\mathcal{L}$-security)**.** *The STE scheme Σ is W-sampled $\mathcal{L}$-secure for the leakage profile $\mathcal{L} = (\mathcal{L}_\mathsf{S}, \mathcal{L}_\mathsf{Q})$ if there exists a p.p.t. simulator $\mathcal{S}$ such that every p.p.t. adversary $\mathcal{A}$ can distinguish whether it is interacting in the real setting $\mathsf{SampledReal}^{\mathcal{A}}_{\Sigma,W}(1^\lambda)$, or in the ideal setting $\mathsf{SampledIdeal}^{\mathcal{A},\mathcal{S}}_{\mathcal{L},W}(1^\lambda)$, with only negligible advantage.*

Definition 13 (Sim-based sampled minimally-leaking)**.** *The STE scheme Σ is sampled minimally-leaking if it is $\mathsf{Uniform}(\mathbb{L})$-sampled $(\mathcal{L}_\mathsf{S}, \mathcal{L}_\mathsf{Q})$-secure, where $\mathsf{Uniform}(\mathbb{L})$ is the uniform distribution over the set $\mathbb{L}$ of labels, the setup leakage $\mathcal{L}_\mathsf{S}$ reveals only the size and the upper bound on the maximum response length of the multi-map, and the query leakage $\mathcal{L}_\mathsf{Q}$ doesn't reveal anything.*

A.1 Minimally-leaking implies sampled minimally-leaking

Theorem 6. *If $\Sigma = (\mathsf{Setup}, \mathsf{Token}, \mathsf{Query}, \mathsf{Resolve})$ is a minimally-leaking STE scheme, then it is sampled minimally-leaking.*

Proof. Suppose that Σ is not sampled minimally-leaking, i.e., there exists a p.p.t. adversary $\mathcal{A}$ that wins $\mathsf{SMLGame}^{\mathcal{A}}_{\Sigma}(1^\lambda)$ with non-negligible advantage. From $\mathcal{A}$, we can construct a p.p.t. reduction $\mathcal{B}$ that wins $\mathsf{MLGame}^{\mathcal{A}}_{\Sigma}(1^\lambda)$ with non-negligible advantage as follows:

Let $\mathcal{C}$ be the challenger in $\mathsf{MLGame}^{\mathcal{A}}_{\Sigma}(1^\lambda)$.

1. First, $\mathcal{A}$ picks two multi-maps MM_0 and MM_1 of the same dimensions (i.e., size and upper bound on the maximum response length) and sends $(\mathsf{MM}_0, \mathsf{MM}_1)$ to $\mathcal{B}$; $\mathcal{B}$ forwards the multi-maps to $\mathcal{C}$.
2. Then, $\mathcal{C}$ samples a bit $b \leftarrow\!\!\$\, \{0,1\}$ and runs the STE scheme's setup algorithm Setup on the security parameter and the randomly chosen multi-map MM_b and replies to $\mathcal{B}$ with the generated encrypted multi-map EMM; $\mathcal{B}$ forwards EMM to $\mathcal{A}$.
3. Recall that $\mathsf{Uniform}(\mathbb{L})$ is the uniform distribution over the set $\mathbb{L}$ of labels. $\mathcal{B}$ samples the queries $q_1, \ldots, q_{\mathsf{poly}(\lambda)}$ independently and uniformly at random from $\mathsf{Uniform}(\mathbb{L})$. $\mathcal{B}$ sends $((q_1, \ldots, q_{\mathsf{poly}(\lambda)}), (q_1, \ldots, q_{\mathsf{poly}(\lambda)}))$ to $\mathcal{C}$.
4. $\mathcal{C}$ computes the tokens $(\tau_1, \ldots, \tau_{\mathsf{poly}(\lambda)})$ by running the STE scheme's token algorithm Token on the key K from step 1 and each query q_i, and sends the tokens to $\mathcal{B}$; $\mathcal{B}$ forwards the tokens to $\mathcal{A}$.
6. When $\mathcal{A}$ outputs a guess b', $\mathcal{B}$ outputs the same guess b'.

Analysis of $\mathcal{B}$. The reduction works because (i) it runs in polynomial time, (ii) the transcript between $\mathcal{A}$ and $\mathcal{B}$ is identically distributed to what $\mathcal{A}$ expects to see "in the wild," and (iii) $\mathcal{B}$ wins whenever $\mathcal{A}$ wins. □

B Optimal storage for optimal read efficiency

Suppose that Σ is minimally-leaking and optimally read-efficient; by *optimally read-efficient*, we mean that with probability 1, $L_1 = \cdots = L_m = t$ where L_i denotes the length of the i^{th} encrypted response c_i. We show

that, in this case, Σ necessarily adds at least as many dummy values as pad-to-max. That is, each response must be "padded" with sufficiently many dummy values so that its length is t.

Theorem 7. *If Σ is minimally-leaking and optimally read-efficient, then with overwhelming probability, the scheme stores at least mt values.*

Proof. For the sake of reaching a contradiction, assume that there exists a multi-map MM and indices $i, j \in [m]$ such that the size of the "overlap" (i.e., the intersection) between the i^{th} encrypted response c_i and the j^{th} encrypted response c_j is at least one with non-negligible probability, i.e., $\Pr[|c_i \cap c_j| \geq 1] = \mathsf{nonnegl}(\lambda)$. Since Σ is correct, this would reveal to an adversary $\mathcal{A}$ that MM cannot be a multi-map such that both $|\mathsf{MM}(l_i)|$ and $|\mathsf{MM}(l_j)|$ equal t. That is, this would reveal to $\mathcal{A}$ that at least one of $|\mathsf{MM}(l_i)|$ or $|\mathsf{MM}(l_j)|$ is less than t, and so Σ would not be minimally-leaking. □

C Experiments

In this section, we show experimentally that our lower bound on storage overhead for the sampled minimally-leaking setting (Theorem 4) matches the naive "pad to the read-efficiency curve" scheme for large values of N, m, and t. We now formally describe the naive sampled minimally-leaking scheme:

Given a multi-map MM with labels $\mathbb{L} = \{l_1, \ldots, l_m\}$ and their corresponding response lengths $\vec{\ell} = (\ell_1, \ldots, \ell_m)$, a dictionary encryption scheme $\mathsf{STE_{DX}} = (\mathsf{Setup}, \mathsf{Get})$, and symmetric encryption scheme $\mathsf{SKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$, and a pseudo-random function $F_K(\cdot)$. Let N, m and t be the public parameters of the scheme.

We define the STE scheme $\mathsf{STE}^{\mathsf{ML}}_{\mathsf{MM}}(\mathsf{Setup}, \mathsf{Token}, \mathsf{Query}, \mathsf{Resolve})$ as follows:

- $\mathsf{Setup}(\mathsf{MM}, 1^\lambda)$: Sample a key $K \leftarrow \{0,1\}^\lambda$. Run $K_1 \leftarrow \mathsf{SKE.Gen}(1^\lambda)$. Sort the response lengths $\ell_1, \ldots, \ell_m$ in ascending order. For each label l_j in sorted order:
  1. Compute $\mathsf{RE}(j)$. Recall that in the sorted order, $\ell_j \leq \mathsf{RE}(j)$ for any multi-map. Add dummy values to the response $\mathsf{MM}(l_j)$ till the total response length is $\mathsf{RE}(j)$.
  2. Encrypt each value with key K_1 and store the encrypted response at the next available location in the encrypted structure EMM. Let this location be L_j.
  3. Add the entry $(F_K(l_j), (L_j, j))$ to the dictionary DX.

  Finally, setup the encrypted dictionary $(K_2, \mathsf{EDX}) \leftarrow \mathsf{STE_{DX}.Setup}(1^\lambda, \mathsf{DX})$. Output keys (K, K_1, K_2) to the client and $(\mathsf{EMM}, \mathsf{EDX})$ to the server.
- $\mathsf{Token}(K, l)$: Output $\tau = F_K(l)$.
- $\mathsf{Query}(\mathsf{EMM}, \tau)$: The client and server execute $(L_k, k) \leftarrow \mathsf{STE_{DX}.Get}(\tau, K_2)$. Output the $\mathsf{RE}(k)$ encrypted values stored at L_k in EMM.
- $\mathsf{Resolve}$: The client uses key K_1 to decrypt all the values in the encrypted response, and discards the empty values.

Intuitively, this scheme is secure in the sampled minimally-leaking setting because: (1) the adversary's view for setup is identical for all multi-maps with the same parameters N, m, and t; and (2) there are no overlaps in the encrypted responses. Then the adversarial view for the sampled query phase is identically-distributed on any multi-map. In particular, the simulator can simulate the adversary's view knowing only the parameters N, m, and t. We formally prove the security of this scheme in Appendix D.

Our experiments indicate that our lower bound matches the storage overhead of the scheme closely for large values of N, m, and t. In particular, when m or t are $O(\sqrt{N})$, our bound is essentially tight.[2] Additionally, as expected, the naive sampled minimally-leaking scheme has asymptotically better storage overhead with respect to the lower bound for the minimally-leaking setting (Theorem 7).

2 From the collection of information retrieval datasets held at the University of Glasgow [27] we see that $m = O(\sqrt{N})$ for almost all the datasets.

Observations. The plots in Figure 6 show our observations when $N \in \{2^{18}, 2^{20}\}$. We notice that the storage overhead of the minimally-leaking scheme asymptotically dominates that of the sampled minimally-leaking scheme in all settings. Further, our lower bound on the storage overhead becomes tighter for larger (more realistic) values of N.

D Proof of security of scheme $\mathsf{STE}^{\mathsf{ML}}_{\mathsf{MM}}$

Theorem 8. *If STE_{DX} is (N_{DX}, resp)-secure, then scheme $\mathsf{STE}^{ML}_{\mathsf{MM}}$ is sampled minimally-leaking.*

Proof. Let $\mathcal{S}_{\mathsf{DX}}$ be the simulator guaranteed to exist by the security of the dictionary encryption scheme $\mathsf{STE}_{\mathsf{DX}}$. To prove security we construct a simulator $\mathcal{S}$ such that a computationally-bounded adversary $\mathcal{A}$ cannot distinguish between $\mathsf{SampledReal}^{\mathcal{A}}_{\Sigma,W}(1^\lambda)$ and $\mathsf{SampledIdeal}^{\mathcal{A},\mathcal{S}}_{\mathcal{L},W}(1^\lambda)$ where $\Sigma = \mathsf{STE}^{\mathsf{ML}}_{\mathsf{MM}}$, W is the uniform distribution on $\mathbb{L}$ and $\mathcal{L} = (\mathcal{L}_{\mathsf{S}}, \mathcal{L}_{\mathsf{Q}})$ where $\mathcal{L}_{\mathsf{S}} = (N, t)$ and $\mathcal{L}_{\mathsf{Q}}$ is empty.

Simulating Setup. Given $\mathcal{L}_{\mathsf{S}}$ and the (public) label space $\mathbb{L}$, $\mathcal{S}$ computes the curve $\mathsf{RE}(N, m, t)$, and samples a key K from $\{0,1\}^\lambda$.

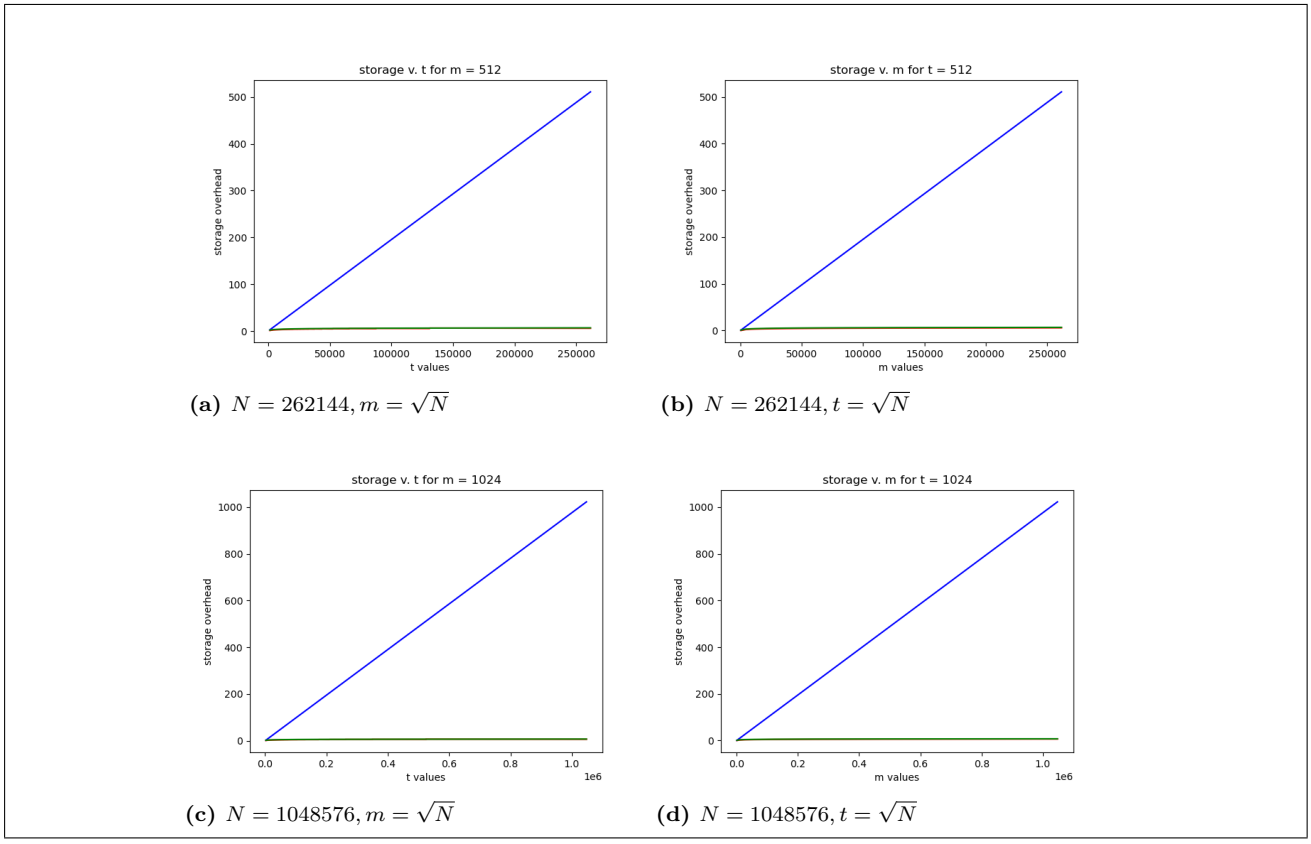
- $\mathcal{S}$ generates a key $K_1 \leftarrow \mathsf{SKE.Gen}(1^\lambda)$ and creates encryptions of $\perp$ for each response length $\mathsf{RE}(i)$.
- $\mathcal{S}$ generates encrypted array EMM by placing the encrypted responses in ascending order of length, $\mathsf{RE}(i)$.
- $\mathcal{S}$ generates $\mathsf{EDX}' \leftarrow \mathcal{S}_{\mathsf{DX}}(m)$.
- $\mathcal{S}$ sends $(\mathsf{EMM}, \mathsf{EDX}')$ to the adversary $\mathcal{A}$.

Simulating Query. Given the number of queries n from the challenger $\mathcal{C}$, the simulator samples uniformly $l_j \leftarrow\!\!\$\, \mathbb{L}$, $j = 1, \ldots, n$. For each l_j, it runs $\mathcal{S}_{\mathsf{DX}}(L_j, j)$ to simulate a Get on EDX'.

Now let Game_0 be the same as the $\mathsf{SampledReal}^{\mathcal{A}}_{\Sigma,W}(1^\lambda)$ experiment with $\Sigma = \mathsf{STE}^{\mathsf{ML}}_{\mathsf{MM}}$ and W the uniform distribution on $\mathbb{L}$, for some multi-map MM of size N, and upper bound on the maximum response length t. Consider the following experiments:

- Game_1: Same as Game_0 except all the encrypted values are replaced with encryptions of $\perp$. The probabilities of Game_1 and Game_0 outputting 1 are negligibly close by the security of the encryption scheme SKE.
- Game_2: Same as Game_1 except EDX is replaced by $\mathcal{S}_{\mathsf{DX}}(m)$ and every Get on EDX is replaced by running $\mathcal{S}_{\mathsf{DX}}(L_j, j)$ for the corresponding l_j. The probabilities of Game_2 and Game_1 outputting 1 are negligibly close by the security of the dictionary encryption scheme $\mathsf{STE}_{\mathsf{DX}}$.
- Game_3: Same as Game_2 except that the queries in the multi-map are permuted such that the ascending order of response lengths is also the lexicographical order. The probability of Game_3 outputting 1 is distributed identically to that of Game_2 because: (1) the adversary's view during Setup is identical; and (2) during Query, the uniform distribution on $\mathbb{L}$ is identical to the uniform distribution on any permutation of $\mathbb{L}$.
- Game_4: Same as Game_3 except the n queries are sampled by the simulator $\mathcal{S}$ instead of the challenger $\mathcal{C}$. The output of Game_4 is identically distributed to that of Game_3.

We finally note that the probability of Game_4 outputting 1 is exactly that of $\mathsf{SampledIdeal}^{\mathcal{A},\mathcal{S}}_{\mathcal{L},W}(1^\lambda)$ outputting 1 and our proof is complete. □



(a) $N = 262144, m = \sqrt{N}$ **(b)** $N = 262144, t = \sqrt{N}$

(c) $N = 1048576, m = \sqrt{N}$ **(d)** $N = 1048576, t = \sqrt{N}$

Fig. 6. $N = 262144, 1048576$, and $m = \sqrt{N}$ or $t = \sqrt{N}$. In each plot, the blue curve represents the lower bound on storage overhead for the minimally-leaking setting. The green curve represents the storage overhead of the naive sampled minimally-leaking scheme, and the red curve represents our lower bound for the sampled minimally-leaking setting.