CR-SHE: Collusion-Resilient Searchable Homomorphic Encryption for Multi-Cloud IoT Data Sharing

Md Mazharul Islam, Abrar Mohammed Tanzim Alam, and Mubasshir Ahmed

***Abstract*—Internet of Things (IoT) systems increasingly store their data across several cloud providers at once, which improves reliability and avoids dependence on any single vendor. This dispersal, however, creates a privacy risk that prior work has largely overlooked: if two or more providers secretly cooperate, they can combine what each observes and recover information the encryption was meant to protect, such as the keywords a user searches for, the records accessed, or their contents. To address this overlooked threat, we define an adversary of up to $n-1$ colluding cloud providers and present CR-SHE, a hybrid scheme that lets an authorized user search encrypted multi-cloud IoT data by keyword and compute over it homomorphically, without any provider ever decrypting it. The core idea is to split each query across the providers using a distributed point function, so that as long as at least one provider stays honest, the colluding group learns nothing about the query, the access pattern, or the result, while computations run directly on the homomorphically encrypted records. We prove these guarantees under standard cryptographic assumptions, and we implement a prototype, evaluated on a real IoT sensor-telemetry dataset, that confirms efficient keyword search, compact communication, and practical homomorphic aggregation.**

***Index Terms*—Cloud computing, distributed point function, homomorphic encryption, Internet of Things, searchable encryption.**

I. Introduction

THE proliferation of connected sensors, wearables, and cyber-physical systems has made the Internet of Things (IoT) one of the largest and fastest-growing sources of continuously generated data [1]. Bedside monitors, smart meters, traffic and environmental sensors, and industrial controllers emit measurements at volumes and velocities exceeding local storage and processing capabilities. The natural response is to offload this data to the cloud, and modern deployments increasingly distribute it across multiple providers. Multi-cloud operation has become an operational norm in IoT platforms: it supplies elastic storage and compute, improves availability through redundancy across independent infrastructures, and frees data owners from single-vendor lock-in [2]. Yet much of this data is acutely sensitive physiological telemetry in connected healthcare, mobility traces in smart cities, diagnostic streams from connected vehicles so it must be encrypted before leaving the IoT trust domain and stay encrypted in the cloud, even as multiple stakeholders need to query and analyze it.

(Corresponding author: Md Mazharul Islam.)
Md Mazharul Islam, Abrar Mohammed Tanzim Alam, and Mubasshir Ahmed are with the Department of Electrical and Computer Engineering, North South University, Dhaka, Bangladesh (e-mail: mazharul.islam1@northsouth.edu; abrar.alam01@northsouth.edu; mubasshir.ahmed@northsouth.edu).

Encrypting outsourced data with conventional schemes, however, neutralizes the very reason it was sent to the cloud: a provider holding only ciphertexts can neither locate the records a query needs nor compute over them, forcing either decryption in the cloud which reintroduces exposure or a wholesale download to the client, which defeats the purpose of outsourcing [3]. Two complementary lines of cryptography escape this dilemma for a single server. Searchable encryption (SE) lets a provider identify the records matching an encrypted keyword without learning the keyword or the plaintext [4]. Homomorphic encryption (HE) lets it evaluate functions directly on ciphertexts, producing aggregates and analytics without any decryption; leveled HE in particular supports circuits up to a chosen multiplicative depth at practical parameters [5]. Together these promise outsourced storage that is at once confidential, searchable, and computable, even against a curious provider that follows the protocol but inspects everything it sees.

These guarantees do not carry over to the multi-cloud setting, where distributing data across providers creates a new adversary: providers that collude [6]. Independent backends can correlate information, reconstructing search patterns, access patterns, and even record contents. The implied coalition adversary is rarely modeled formally, and existing constructions leave this intersection open: schemes that compute homomorphically offer weak keyword search; pattern-hiding searchable encryption assumes a single trust domain; and schemes combining search with computation remain single-cloud [7]. No prior construction simultaneously (i) defines a cross-cloud collusion adversary, (ii) provably bounds what a colluding coalition learns, and (iii) supports keyword retrieval and homomorphic analytics end to end. This is the gap we close.

To address this, we propose CR-SHE (Collusion-Resilient Searchable Homomorphic Encryption), a hybrid scheme for multi-cloud IoT data sharing that unifies private keyword search and computation over encrypted data. The data owner builds an encrypted keyword index and replicates it to the n providers, together with the records' computable fields under leveled HE. To search, an authorized user issues one distributed point function (DPF) key per provider; each evaluates its key against its replicated index and returns an additive share, which the user recombines. Because the DPF splits the query into shares, a coalition of up to $t = n-1$ providers learns nothing about the queried keyword or the induced access pattern, while homomorphic aggregates run entirely on ciphertexts. CR-SHE's search is linear-work but symmetric-key only, with no public-key operation on the search path,

and embarrassingly parallel across providers and cores. The main contributions of this work are as follows:

- Formalizes the first t-of-n cross-cloud collusion adversary for searchable homomorphic IoT data sharing, with a leakage profile that separates what colluding providers observe from what the authorized combiner observes.
- Distributes a replicated pseudonymous index through a distributed point function and couples it with a leveled homomorphic-encryption layer, supporting keyword retrieval and homomorphic aggregates over the retrieved ciphertexts without any decryption at the providers.
- Proves correctness, $t = n - 1$ collusion leakage security, and content confidentiality under standard assumptions, complemented by an asymptotic analysis of storage, communication, and computation.
- Implements an open, reproducible prototype, evaluated on a real IoT telemetry dataset against plaintext and pure-FHE baselines, characterizing the regimes in which the hybrid is practical for IoT workloads.

The remainder of this paper is organized as follows. Section II reviews related work. Section III introduces the notation and cryptographic primitives. Sections IV and V present the system and threat models. Section VI details the CR-SHE construction. Section VII develops the security analysis. Section VIII analyzes asymptotic costs. Section IX describes the implementation, and Section X reports and discusses the results. Finally, Section XI concludes and outlines future work.

II. Related Work

CR-SHE draws together three lines of work: secure multi-cloud data sharing, computation over encrypted data, and searchable encryption. The first disperses trust so no single provider holds the data in the clear; the second lets a server compute on ciphertexts without decrypting them; the third lets it locate records matching a query without learning the query or plaintext. Each is mature alone, but the three are rarely combined, and almost never under a trust model anticipating providers acting together. We review each below, then position CR-SHE against the closest schemes.

A large body of work distributes data across providers to avoid placing trust in any single one, using hybrid cryptosystems, secret sharing, or attribute-based access control [8], with layered privacy-preserving architectures targeting sensitive domains such as healthcare [9], [10]. These designs strengthen confidentiality and availability across clouds, but they treat the providers as independent storage: they neither process queries over the ciphertexts nor model providers that collude across clouds, which is the adversary CR-SHE targets.

Homomorphic encryption enables computation directly on ciphertexts, and several works study its use for outsourced and multi-cloud analytics [11], [12]. SHAMC realizes secure database operations across multiple clouds by combining secure computation with partial homomorphic encryption [13], and other schemes apply homomorphic primitives to multi-keyword retrieval [14]. The recurring obstacle these works report is cost: full homomorphic evaluation remains expensive, which motivates the hybrid rather than pure FHE design that CR-SHE adopts.

Searchable encryption supports keyword retrieval over ciphertexts, with access and search-pattern leakage as the central residual risk [15], [16]. Recent schemes reduce this leakage or broaden expressiveness, including search pattern hiding symmetric searchable encryption in a multi-cloud setting [17], flexible and dynamic keyword search [18], and authorized searchable encryption with committee-based access control [19]. These schemes, however, either assume a single trust domain or scale their hiding overhead with the number of clouds, and none couples search with homomorphic computation over the retrieved records.

A complementary line ensures that outsourced data remains intact and that results are trustworthy, through provable data possession and verified public-key encryption with equality test [20], [21]. Within IoT specifically, privacy-preserving search has been studied for the Internet of Things [22] and the Internet of Vehicles [23], and federated multi-party computation with homomorphic encryption addresses the multi-owner case [24]. These works confirm the IoT relevance of encrypted search and computation but stop short of a unified, collusion-resilient guarantee across providers. Surveys of privacy-preserving query processing conclude that hybrid or layered designs offer the best security performance balance [25], while systematization of cryptography for multi-cloud storage notes the absence of standardized trust model definitions and calls for formally verified designs [26]. CR-SHE answers this call.

Table I compares CR-SHE with the most closely related schemes: no prior construction simultaneously defines a cross-cloud collusion adversary, provably bounds what a colluding coalition learns, and supports keyword retrieval and homomorphic analytics end to end. CR-SHE occupies that intersection.

III. Preliminaries

This section establishes the notation and recalls the cryptographic primitives on which CR-SHE is built: pseudorandom functions, additive secret sharing, distributed point functions, and leveled homomorphic encryption. These primitives provide, respectively, the pseudonymous addressing of keywords, the splitting of query answers into individually meaningless shares, the private evaluation of a point query across the replicated index, and the computation of aggregates directly on ciphertexts the four ingredients CR-SHE combines to achieve collusion-resilient search and computation. The symbols used throughout the paper, together with their meanings, are summarized in Table II.

Pseudorandom functions. $F : \{0,1\}^\lambda \times \{0,1\}^* \to \{0,1\}^\lambda$ is a PRF if for every PPT distinguisher D, $\mathbf{Adv}^{\mathrm{prf}}_{F,D}(\lambda) = |\Pr[D^{F_K(\cdot)} = 1] - \Pr[D^{R(\cdot)} = 1]| \le \mathsf{negl}(\lambda)$, where R is a truly random function [27].

Additive secret sharing. An (n, n) additive sharing splits m into $(m_1, \ldots, m_n)$ with $\sum_j m_j = m$; any $n - 1$ shares are uniformly distributed and independent of m. CR-SHE uses additive sharing for the per-provider DPF answer shares.

Distributed point functions. An n-party DPF [28], [29] for point functions $f_{\alpha,\beta} : [N] \to \mathbb{G}$ (value β at α, else 0) is a pair $(\mathsf{Gen}, \mathsf{Eval})$: *(correctness)* $\sum_{j\in[n]} \mathsf{Eval}(j, \kappa_j, x) = f_{\alpha,\beta}(x)$

TABLE I
COMPARISON OF CR-SHE WITH THE MOST RELEVANT PRIOR SCHEMES.

Scheme	Year	Keyword search	Homomorphic computation	Multi-cloud	Cross-cloud collusion model	Pattern hiding	Verifiable results
SHAMC [13]	2020	○	✓	✓	○	×	×
Multi-cloud SSE [15]	2021	✓	×	✓	×	○	×
SE-EPOMFC [16]	2022	✓	○	○	×	×	×
Pattern-hiding SSE [17]	2023	✓	×	✓	○	✓	×
DCC-Auth. SE [19]	2023	✓	×	○	×	×	✓
HE+Prim search [14]	2023	✓	✓	×	×	×	×
Flexible KW search [18]	2024	✓	×	×	×	○	×
IoT searchable [22]	2019	✓	×	×	×	×	×
CR-SHE	2026	✓	✓	✓	✓	✓	×†

✓: supported; ○: partial; ×: not supported.

TABLE II
SUMMARY OF NOTATION.

Symbol	Meaning
λ	security parameter
PPT, negl	probabilistic poly. time; negligible function
$\approx_c$	computational indistinguishability
n, t	number of providers; collusion bound ($t = n-1$)
N, N_w, N_d	index domain size; keyword-universe bound; #records
Q	number of adaptive queries
F, K	pseudorandom function; its key
I	replicated pseudonymous inverted index
α_w	index address of keyword w ($\alpha_w = F_K(w)$)
$(\mathsf{Gen}, \mathsf{Eval})$	distributed point function (DPF)
κ_j	DPF key sent to provider CP_j
$\mathcal{S}_{\mathrm{DPF}}$	DPF simulator
pk, sk	HE public / secret key
L	HE multiplicative-depth bound
CT, ct_f	record ciphertexts; computed-result ciphertext
s_j, S	provider answer share; matched record set
$\mathcal{L}$	leakage profile (Setup / Search / Compute)
C	corrupted provider coalition ($\lvert C\rvert \le t$)
DO, DU, CP_j	data owner; data user; j-th cloud provider

TABLE III
ENTITIES, THEIR TRUST STATUS, AND WHAT THEY HOLD.

Entity	Role	Trust	Holds
Data Owner	builds/outsources index; authorizes	trusted	K, sk
Data Producers	generate records	trusted (in domain)	raw data
Edge relays	transport/buffering	untrusted w/ secrets	nothing secret
Cloud Providers	store; search; HE-evaluate	semi-honest ($\le t$ collude)	replicated I, CT, pk, κ_j
Data Users	query; combine; decrypt	authorized/ trusted	queries, delegated sk

for all x; *($(n-1)$-privacy)* there is a PPT simulator $\mathcal{S}_{\mathrm{DPF}}$ such that for every coalition $C \subseteq [n]$ with $|C| \le n-1$, $\{\kappa_j\}_{j\in C} \approx_c \mathcal{S}_{\mathrm{DPF}}(1^\lambda, C, N)$; we write $\mathbf{Adv}^{\mathrm{dpf}}$ for the advantage in distinguishing real keys from simulated. The two-party instantiation has keys of size $O(\lambda \log N)$; n-party constructions with $(n-1)$-privacy incur larger keys, part of the cost characterized in Section X. Intuitively, a DPF lets the data user split a single "which address" query into n keys so that each provider, holding only its key, evaluates the query against the replicated index and returns an additive share of the answer; recombining the shares recovers the result, while no coalition of up to $n-1$ providers can tell which address hence which keyword was asked.

Leveled homomorphic encryption. $(\mathsf{KGen}, \mathsf{Enc}, \mathsf{Eval}, \mathsf{Dec})$ is correct for arithmetic circuits of multiplicative depth $\le L$ and IND-CPA secure with advantage $\mathbf{Adv}^{\mathrm{cpa}}$ [30], [31]. In CR-SHE the providers hold only pk; sk never leaves the DO/DU. Opaque payloads use an IND-CPA symmetric scheme.

IV. SYSTEM MODEL

CR-SHE spans an IoT trust domain and an untrusted multi-cloud back-end, across five classes of entities summarized in Table III. The **Data Owner (DO)** an IoT-domain authority such as a hospital or fleet operator is the root of trust, holding the index key K and the HE secret key sk. **Data Producers** and **Edge relays** lie inside the DO trust domain, generating and transporting raw records with no secret material. The n **Cloud Providers** $\mathrm{CP}_1, \ldots, \mathrm{CP}_n$ are independent and semi-honest: each holds only the public key and a replicated copy of the index and ciphertexts, executing search and homomorphic evaluation but never decrypting. Authorized **Data Users (DU)** issue queries, recombine the per-provider answers, and hold delegated decryption.

The workflow proceeds in five stages (Fig. 1). In setup, the DO derives the keys, builds the pseudonymous index placing each keyword at a PRF-derived address, encrypts the computable fields under leveled HE, and replicates the index and ciphertexts to all n providers. To query, an authorized DU sends one DPF key per provider; each evaluates its key against its index and returns an additive share of the matching record set, which the DU recombines into S. The requested aggregate is then evaluated homomorphically over the ciphertexts of the records in S, entirely on the providers, and the DU decrypts the returned ciphertext with sk. Because the per-provider DPF keys are individually meaningless, every network-visible message is independent of the queried keyword, so a coalition of up to $t = n-1$ colluding providers learns nothing query-dependent.

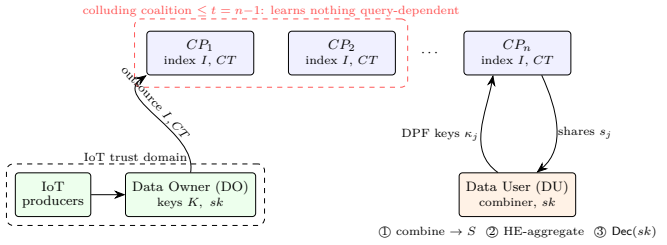

Fig. 1. CR-SHE architecture and query workflow across n providers.

V. Threat Model

The adversary $\mathcal{A}$ is probabilistic polynomial-time, static, and semi-honest: every corrupted party follows the protocol faithfully but pools and analyzes everything it observes to learn what it is not entitled to. Before execution, $\mathcal{A}$ fixes a corruption set $C \subseteq [n]$ with $|C| \leq t$; the headline guarantee is $t = n-1$, so security holds as long as at least one provider stays out of the coalition. The defining feature is *collusion*: the corrupted providers act not as isolated servers but as a single coalition sharing state, keys, tokens, and transcripts. For its chosen C, the adversary observes the full static state and complete per-query transcript of every corrupted provider,

$$\text{View}_C = \Big(\{I_j\}_{j\in C}, \text{CT}, pk, pp, \{(\kappa_{q,j}, \text{msg}_{q,j})\}_{j\in C, q\in[Q]}\Big),$$

namely the replicated index copies, the record ciphertexts CT, the public key pk and parameters pp, and the per-provider DPF keys and messages for each of the Q queries. Queries are adaptive, so $\mathcal{A}$ may choose each one after seeing the responses to all previous ones. The security goal is that this coalition view be simulatable from data-independent leakage revealing neither the keyword, the access pattern, nor the result-so that pooling observations yields no query-dependent advantage.

The surrounding trust assumptions are explicit. The data owner and authorized data users are honest, holding the secret keys as the root of trust; the data producers and in-domain edge relays hold no secret material. All channels are authenticated and confidential, so the adversary gains nothing from wiretapping beyond what the corrupted providers already see and cannot tamper with messages.

Three threats are out of scope. First, malicious (actively deviating) providers that return incorrect results or corrupt evaluations are not handled; defending against them requires verifiable computation, left to future work, which is why verifiable results are not claimed. Second, data-user–provider collusion is excluded, since an authorized user already holds the decryption capability and could trivially reveal results. Third, side channels and traffic-volume leakage beyond the result-set size lie outside the cryptographic model, assumed mitigated by orthogonal system-level defenses. These boundaries delimit the cross-cloud collusion adversary CR-SHE provably defends against.

VI. The CR-SHE Construction

The DO builds a pseudonymous inverted index (keyword w sits at address $\alpha_w = F_K(w)$) and *replicates* it to all providers, with the computable record fields encrypted under HE alongside; the PRF keeps the keyword-to-address map hidden. To search for w, an authorized DU derives its address and generates one DPF key per provider; each provider evaluates its key against its copy and returns an additive share of the matching record set, which the DU combines. Homomorphic aggregates are then evaluated over the HE ciphertexts of the matched records and returned for decryption by the DU. Algorithms 1 and 2 specify the procedures.

Algorithm 1 Outsourcing: Setup, BuildIndex, EncData

procedure Setup($1^\lambda, n, t$)
  $K \leftarrow \{0,1\}^\lambda$; $(pk, sk) \leftarrow \mathsf{KGen}(1^\lambda, L)$
  $pp \leftarrow (n, t, N)$; **return** (K, pk, sk, pp)
end procedure
procedure BuildIndex(K, DB)
  Build inverted index I over $[N]$; address of w is $F_K(w)$
  return I ▷ replicated to every CP_j
end procedure
procedure EncData(pk, DB)
  CT ← HE-encrypt computable fields; symmetric-encrypt payloads
  return CT
end procedure

Algorithm 2 Query: Token, Search, Compute, Dec

procedure Token(K, w)
  $\alpha_w \leftarrow F_K(w)$; $(\kappa_1, \ldots, \kappa_n) \leftarrow \mathsf{DPF.Gen}(1^\lambda, \alpha_w, 1)$
  return $(\kappa_1, \ldots, \kappa_n)$ ▷ $\kappa_j \rightarrow \text{CP}_j$
end procedure
procedure Search($\{\kappa_j\}_{j\in[n]}, I$)
  for all CP_j in parallel **do** $s_j \leftarrow \sum_{x\in[N]} \mathsf{Eval}(j, \kappa_j, x) \cdot I[x]$
  end for
  DU outputs $S \leftarrow \sum_{j\in[n]} s_j$ ▷ matched record set
end procedure
procedure Compute($pk, \{\text{CT}_i\}_{i\in S}, f$)
  return $ct_f \leftarrow \mathsf{Eval}(pk, f, \{\text{CT}_i\}_{i\in S})$ ▷ depth $f \leq L$
end procedure
procedure Dec(sk, ct_f)
  return $\mathsf{Dec}(sk, ct_f)$
end procedure

The index is laid out so the point query at address α_w retrieves (shares of) the posting list of w. The admissible class for Compute covers bounded-degree aggregates count, sum, mean numerator, plaintext-weighted sum and inner product, and depth-$\leq L$ polynomials spanning common IoT analytics while keeping HE parameters small, chosen for the target depth L and the deployment's data types.

As a concrete instance, a clinician searching `arrhythmia` retrieves the matching record set S via the DPF keys and then requests a homomorphic aggregate, say the mean heart-rate, over $\{\text{CT}_i\}_{i\in S}$; the providers evaluate it on ciphertexts and the user decrypts the result. No single provider, and no coalition of up to $n-1$, learns the queried keyword, the access pattern,

or any plaintext record.

Hiding the access pattern from a coalition forces $\Omega(N)$ work per provider per query, by the lower bounds underlying ORAM and PIR [32]. CR-SHE's search is therefore linear-work but symmetric-only (PRG evaluations inside the DPF), with no public-key operation on the search path, and embarrassingly parallel across providers and cores.

VII. Security Analysis

Leakage profile. We define $\mathcal{L} = (\mathcal{L}_{\text{Setup}}, \mathcal{L}_{\text{Search}}, \mathcal{L}_{\text{Compute}})$ with an observer split. $\mathcal{L}_{\text{Setup}}(\text{DB}) = (n, t, N_d, N_w, I)$ exposes public sizes and the replicated pseudonymous index I. The PRF hides the keyword-to-address map—the coalition never learns which keyword sits at an address—but since I is an inverted index, the posting-list length at each address is visible. This is volume (frequency) leakage: the coalition learns the multiset of keyword frequencies, not the keywords, so an adversary with an auxiliary frequency distribution could attempt to re-identify high- or low-frequency addresses, a known attack against searchable encryption.

Standard countermeasures apply: padding every posting list to a common length removes the leakage at a storage cost, and storing the index under HE removes it at $\mathcal{O}(N)$ homomorphic operations per query. We adopt neither, treating both index-hiding variants as future work, so the guarantees hold relative to a $\mathcal{L}_{\text{Setup}}$ that includes posting-list lengths. For search, $\mathcal{L}^{\text{CP}}_{\text{Search}} = \perp$—the coalition learns nothing query-dependent—while the DU learns only $\mathcal{L}^{\text{DU}}_{\text{Search}} = |S|$, and $\mathcal{L}_{\text{Compute}} = (\text{class}(f), |ct_f|)$. The vanishing coalition leakage $\mathcal{L}^{\text{CP}}_{\text{Search}} = \perp$ is the precise sense in which CR-SHE bounds cross-cloud leakage. A conjunctive query is two retrievals intersected at the DU, adding only result sizes to $\mathcal{L}^{\text{DU}}_{\text{Search}}$, never to the coalition.

Security rests on three guarantees the proof peels off in turn: the PRF makes query addresses look random to the providers; the DPF's $(n{-}1)$-privacy hides which address is queried from any coalition of up to $n-1$ providers; and IND-CPA hides the record contents behind the public key. Composing these, a coalition's entire view is regenerable by a simulator that sees only data-independent leakage, as Definition 1 formalizes and Theorem 2 establishes.

Definition 1 ($\mathcal{L}$-security against t-collusion)**.** *Consider two experiments. In* $\mathbf{Real}_{\mathcal{A},\Pi}(1^\lambda)$*: (1)* $\mathcal{A}$ *outputs* (DB, C)*,* $|C| \le t$*; (2) the challenger runs* $(K, pk, sk, pp) \leftarrow \mathsf{Setup}(1^\lambda, n, t)$*,* $I \leftarrow \mathsf{BuildIndex}(K, DB)$*,* $CT \leftarrow \mathsf{EncData}(pk, DB)$ *and gives* $\mathcal{A}$ *the static view* (I, CT, pk, pp) *(the replicated copies* $\{I_j\}_{j\in C}$ *equal* I*); (3) for* $q = 1, \ldots, Q$*,* $\mathcal{A}$ *adaptively issues a keyword* w_q *or compute request* $(f_q, \cdot)$ *and receives the coalition's per-query view* $(\{\kappa_{q,j}\}_{j\in C}, msg_{q,j\in C})$*; (4)* $\mathcal{A}$ *outputs a bit. In* $\mathbf{Ideal}_{\mathcal{A},\mathcal{S},\mathcal{L}}(1^\lambda)$*, a PPT simulator* $\mathcal{S}$ *produces the views given only* $\mathcal{L}_{\text{Setup}}$ *and the per-query leakage.* Π *is* $\mathcal{L}$*-adaptively-secure against* t*-collusion if for every PPT* $\mathcal{A}$ *corrupting* $\le t$ *providers there is a PPT* $\mathcal{S}$ *with* $|\Pr[\mathbf{Real}_{\mathcal{A},\Pi}(1^\lambda) = 1] - \Pr[\mathbf{Ideal}_{\mathcal{A},\mathcal{S},\mathcal{L}}(1^\lambda) = 1]| \le \text{negl}(\lambda)$.

Theorem 1 (Correctness)**.** *For every* DB*, keyword* w*, and admissible* f *of depth* $\le L$*,* Search *returns exactly the records matching* w*, and* $\mathsf{Dec}(sk, \mathsf{Compute}(pk, \{CT_i\}_{i\in S}, f)) = f(\text{records}(S))$*, except with probability* $\le Q^2/2^\lambda + \text{negl}(\lambda)$.

Proof. By DPF correctness, $\sum_j \mathsf{Eval}(j, \kappa_j, x) = f_{\alpha_w,1}(x)$, which is 1 at $x = \alpha_w$ and 0 elsewhere; hence $S = \sum_j s_j = I[\alpha_w]$, the posting list at α_w. Distinct keywords collide only if $F_K(w) = F_K(w')$, with probability $\le Q^2/2^\lambda$ over Q queries; absent collision, $I[\alpha_w]$ is exactly the records containing w. Leveled-HE correctness for depth $\le L$ gives $\mathsf{Dec}(sk, \mathsf{Eval}(pk, f, \{\text{CT}_i\})) = f(\text{records}(S))$ except with $\text{negl}(\lambda)$. Combining the bounds yields the claim. □

Theorem 2 (t-collusion leakage security)**.** *If* F *is a secure PRF,* $(\mathsf{Gen}, \mathsf{Eval})$ *is an* $(n{-}1)$*-private DPF, and the HE and symmetric schemes are IND-CPA, then CR-SHE is* $\mathcal{L}$*-adaptively-secure against* t*-collusion for* $t = n-1$ *with* $\mathcal{L}^{CP}_{\text{Search}} = \perp$*, where* $\mathcal{L}_{\text{Setup}}$ *includes the replicated pseudonymous index* I *(its membership structure under PRF-derived addresses, not the keyword-to-address map). Concretely, for every PPT* $\mathcal{A}$ *there are PPT* $\mathcal{S}$ *and* B_1, B_2, B_3 *with*

$$\big|\Pr[\mathbf{Real}{=}1] - \Pr[\mathbf{Ideal}{=}1]\big| \le \text{Adv}^{\text{PRF}}_{F,B_1} + Q \cdot \text{Adv}^{\text{DPF}}_{B_2} + m \cdot \text{Adv}^{\text{CPA}}_{B_3},$$

where m *is the number of ciphertexts in the coalition view.*

Proof. Take $|C| = n-1$ without loss of generality (a smaller coalition holds a subset of the same keys, revealing no more). We use a sequence of hybrids, $H_0 = \mathbf{Real}$.

H_1 *(PRF → random).* Replace every F_K evaluation with a truly random R; a distinguisher yields a PRF distinguisher B_1, so $|\Pr[H_0{=}1] - \Pr[H_1{=}1]| \le \text{Adv}^{\text{PRF}}_{F,B_1}$. Now query addresses are uniform on $[N]$.

H_2 *(DPF keys → simulated).* Replace the coalition's keys $\{\kappa_{q,j}\}_{j\in C}$ with $\mathcal{S}_{\text{DPF}}(1^\lambda, C, N)$, licensed by $(n{-}1)$-privacy. A hybrid over the Q queries gives $|\Pr[H_1{=}1] - \Pr[H_2{=}1]| \le Q \cdot \text{Adv}^{\text{DPF}}_{B_2}$. The per-query view now depends only on (C, N), independent of every keyword and access pattern.

H_3 *(ciphertexts → encryptions of* 0*).* Replace each of the m ciphertexts visible to the coalition by encryptions of 0 under pk. Since the coalition holds only pk (sk stays with the DO/DU), $|\Pr[H_2{=}1] - \Pr[H_3{=}1]| \le m \cdot \text{Adv}^{\text{CPA}}_{B_3}$.

Simulator. In H_3 the view is produced from (a) the index I in $\mathcal{L}_{\text{Setup}}$, (b) keys from $\mathcal{S}_{\text{DPF}}(1^\lambda, C, N)$, and (c) encryptions of 0 sized by the public leakage. Thus $\mathbf{Ideal}_{\mathcal{A},\mathcal{S},\mathcal{L}}$ matches H_3, and summing the three bounds proves the theorem. □

Remark 1 (Content confidentiality)**.** *For two databases of equal public sizes differing only in contents, Theorem 2 makes each coalition view indistinguishable from a single distribution depending only on* $\mathcal{L}_{\text{Setup}}$*, which is identical for both; the two executions therefore differ by at most* $m \cdot \text{Adv}^{\text{CPA}}$*, the content-dependence isolated in the* H_3 *step. Security also degrades gracefully: the same proof applies for any* $|C| \le n-1$*, since* $(n{-}1)$*-privacy holds for every strict subset, while a coalition of all* n *providers is outside the model.*

TABLE IV
ASYMPTOTIC COSTS OF CR-SHE. HERE $\gamma(n, N)$ DENOTES THE PER-PROVIDER DPF KEY SIZE: $\mathcal{O}(\lambda \log N)$ FOR THE TWO-PARTY INSTANTIATION ($n = 2$), AND LARGER FOR GENERAL $(n-1)$-PRIVATE n-PARTY DPFS.

Resource	Per provider	DU (combiner)
Index storage	$\mathcal{O}(\|I\|)$	—
Search comm.	$\mathcal{O}(\gamma(n, N) + \|S\|)$	$\mathcal{O}(n\,\gamma(n, N) + n\|S\|)$
Search comp.	$\mathcal{O}(N)$ (symmetric)	$\mathcal{O}(n\|S\|)$
Compute	$\mathcal{O}(\|S\|)$ (HE ops)	$\mathcal{O}(1)$ decrypt

VIII. COMPLEXITY ANALYSIS

We analyze the asymptotic cost of CR-SHE along four axes: storage, search communication, search computation, and homomorphic computation summarized in Table IV. Each provider stores a replicated index of size $O(|I|)$ plus its ciphertext portion, an $n\times$ index blow-up plus HE ciphertext expansion. For search, the data user sends one DPF key per provider and receives one result share of size $\mathcal{O}(|S|)$ each; the per-key size depends on the construction, $\mathcal{O}(\lambda \log N)$ for the two-party instantiation and larger for general $(n-1)$-private n-party DPFs, where it grows with n. Writing $\gamma(n, N)$ for the per-provider key size, total search communication is $\mathcal{O}(n\,\gamma(n, N) + n|S|)$, which is $\mathcal{O}(n\lambda \log N + n|S|)$ at $n = 2$. Search computation is $O(N)$ symmetric pseudorandom-generator operations per provider in a fully parallelizable full-domain pass, with an $O(n|S|)$ combine at the user. Finally, homomorphic evaluation of an admissible f costs $O(|S|)$ ciphertext operations at depth at most L, plus a single decryption. The hidden constants are characterized empirically next.

IX. IMPLEMENTATION

To validate CR-SHE and characterize its cost, we built an open, reproducible prototype implementing the full scheme end to end, across both private keyword search and homomorphic aggregation. It realizes a two-party distributed point function as a BGI-style GGM tree with a SHA-512 pseudorandom generator, exhaustively tested for correctness; a replicated pseudonymous membership index; and a homomorphic layer for aggregation. While the construction is defined over a general leveled HE scheme, the prototype instantiates its additive fragment with the Paillier cryptosystem at a 2048-bit modulus. This suffices for the workload evaluated here counts, sums, means, and plaintext-weighted inner products where the only ciphertext operation is addition and any multiplication uses a public plaintext scalar; richer analytics needing true multiplicative depth (e.g., variance, or inner products of two encrypted vectors) call for a leveled scheme such as BFV, BGV, or CKKS, left to future work. The code and scripts reproducing every measurement are in Python and released openly for verification.

We evaluate CR-SHE on a single real IoT dataset, the Environmental Sensor Telemetry corpus of 405,184 MQTT messages from three Raspberry-Pi sensor arrays, each carrying a device identifier, a timestamp, and readings for carbon monoxide, humidity, liquid petroleum gas, smoke, temperature, light, and motion [33]. Both halves run on the same records. For search, each record is indexed under the metadata tags a real IoT platform would attach device identifier, one-minute time window, and discretized sensor bands yielding a vocabulary of 15,347 tags whose posting-list sizes are heavy-tailed (median 35, mean 238, maximum 404,702). For aggregation, the scheme operates on the real temperature readings of the matched records. One coherent dataset thus drives both retrieval and analytics on the very same records, as a deployed system would.

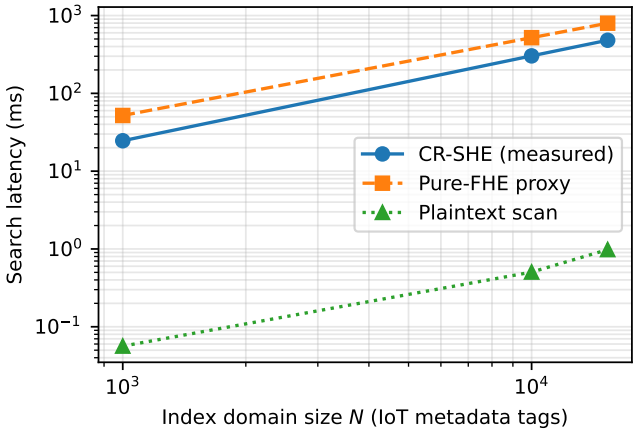

Fig. 2. Per-query search latency vs. domain size N (real IoT metadata-tag domain; one provider, $n = 2$; single core).

We compare against two reference points. A plaintext linear scan provides no privacy and lower-bounds achievable latency. A pure-FHE proxy charges one homomorphic operation per item a deliberately generous lower bound, since a real homomorphic equality scan costs many operations per item, so the proxy understates rather than overstates the cost of a fully homomorphic alternative. All runs are single-core, so the figures reflect per-core cost before any parallelism the construction permits.

X. RESULTS ANALYSIS AND DISCUSSION

We first evaluate search latency as the index domain N grows over the real IoT metadata-tag domain, measured at a single provider (providers run in parallel) alongside the per-provider DPF key size. Latency grows linearly in N, as the complexity analysis predicts, from 24.6 ms at $N = 1{,}000$ to 481.8 ms at $N = 15{,}347$, staying below the pure-FHE proxy throughout (481.8 vs. 797.6 ms at the full domain), while the DPF key grows only logarithmically, 194 to 262 bytes, confirming the succinct $O(\lambda \log N)$ communication. These appear in Fig. 2 and Table V, with the key-size growth in Fig. 3.

We next measure homomorphic-aggregate latency over the real temperature telemetry of the matched records, for the three operations the admissible class supports. Summation scales linearly with the matched-set size $|S|$, from 0.46 ms at $|S| = 10$ to 48.2 ms at $|S| = 1{,}000$; the inner product is heavier owing to the plaintext-scalar multiplications, and the mean

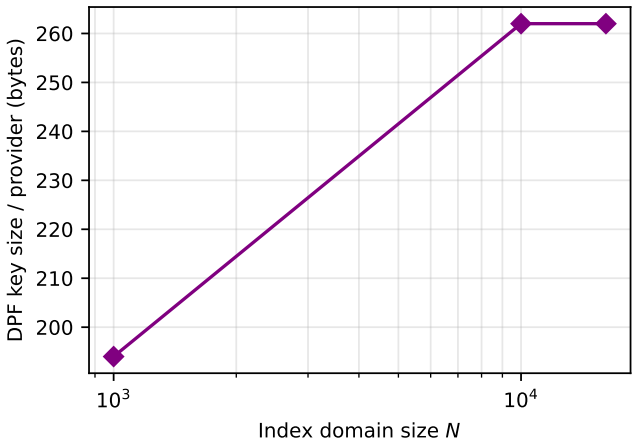

Fig. 3. Per-provider DPF key size vs. domain size N: logarithmic growth ($O(\lambda \log N)$).

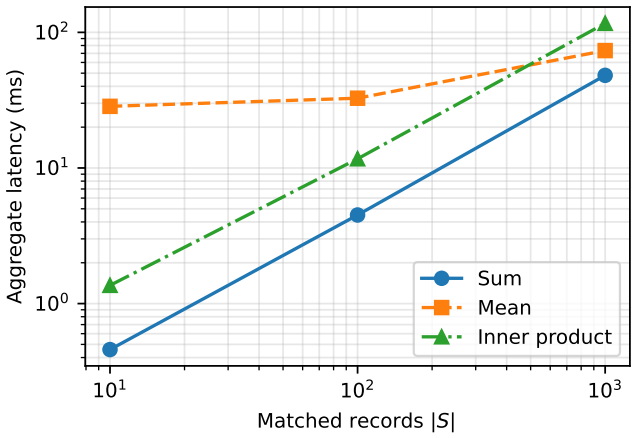

Fig. 4. Homomorphic-aggregate latency vs. matched-set size $|S|$ over real temperature telemetry of matched records (Paillier, 2048-bit).

TABLE V
MEASURED SEARCH LATENCY AND DPF KEY SIZE VS. N (REAL IOT METADATA-TAG DOMAIN; ONE PROVIDER, $n = 2$). PURE-FHE PROXY $= N\times$ ONE PAILLIER OPERATION (A GENEROUS LOWER BOUND FOR FHE).

N	CR-SHE (ms)	Plaintext (ms)	Pure-FHE proxy (ms)	DPF key (B)
1,000	24.6	0.06	52.0	194
10,000	303.2	0.50	519.7	262
15,347	481.8	0.98	797.6	262

adds one decryption. Since the real posting lists skew toward small result sets, these costs are practical for the large majority of queries. The latencies appear in Table VI and Fig. 4, and the result-set-size distribution in Fig. 5.

These prototype-level numbers must be read with care. First, the DPF uses a pure-Python SHA-512 generator rather than hardware AES-NI, which is two to three orders of magnitude faster per evaluation, so the absolute latencies reflect implementation overhead rather than the construction's intrinsic cost. Second, the pure-FHE proxy charges only one homomorphic addition per item and so understates a real homomorphic equality scan. Third, the prototype measures the two-party ($n = 2$) DPF; the construction, proofs, and asymptotics hold for general n, but the reported key sizes and latencies are specific to $n = 2$, with larger n-party keys characterized asymptotically rather than measured. The searchable tags are also derived from record attributes time windows and sensor bands rather than operator labels, and with three physical devices the tag cardinality comes from time and sensor conditions rather than many devices. The evaluation thus substantiates correctness, linear-in-N symmetric search, succinct $O(\lambda \log N)$ keys, and feasible homomorphic aggregates on real data, while a fully optimized comparison against an AES-NI DPF and a faithful homomorphic-equality baseline remains to settle the absolute efficiency margin.

CR-SHE makes deliberate scope choices. Its search is linear-work per provider, the unavoidable price of hiding the

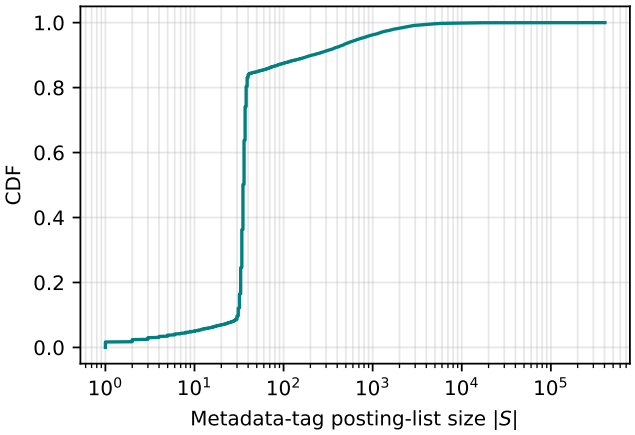

Fig. 5. CDF of metadata-tag posting-list sizes in the IoT telemetry dataset: the real $|S|$ distribution driving the homomorphic-compute workload.

access pattern from a coalition; a partitioning or preprocessing variant could recover sublinear online cost. The $n\times$ index-storage blow-up and the growth of n-party DPF key size with n define a security cost tradeoff, where tolerating more collusion costs more. The adversary is semi-honest; extending to malicious providers would add verification, connecting to verifiable results. The construction assumes a single data owner, with multi-owner multi-key sharing and post-quantum HE instantiations natural extensions.

Taken together, these results map onto the paper's two central claims: the formal collusion model and proven leakage bound, established by Theorems 1 and 2 independently of any measurement, and the practicality of the hybrid design for IoT workloads where pure FHE is not, supported on real data by the search and compute results, with the absolute efficiency margin pending the optimized instantiation noted above.

XI. CONCLUSION

This paper formalized a t of n colluding multi-cloud adversary for IoT data sharing and presented CR-SHE, a

TABLE VI
MEASURED HOMOMORPHIC-AGGREGATE LATENCY VS. MATCHED-SET SIZE $|S|$ OVER REAL TEMPERATURE TELEMETRY OF MATCHED RECORDS (PAILLIER, 2048-BIT).

$\|S\|$	Sum (ms)	Mean (ms)	Inner product (ms)
10	0.46	28.48	1.36
100	4.50	32.69	11.68
1,000	48.19	73.39	116.54

hybrid searchable homomorphic scheme. By replicating a pseudonymous index across providers and querying it via a distributed point function, CR-SHE ensures the joint view of any coalition of up to $t = n - 1$ providers leaks no data-dependent information, while homomorphic analytics are evaluated over retrieved records without server-side decryption. We proved correctness, leakage security, and content confidentiality under standard assumptions, while an evaluation on a real IoT telemetry dataset confirmed linear in N symmetric search, succinct $O(\lambda \log N)$ keys, and practical homomorphic aggregation. Building on these foundations, future work will pursue sublinear preprocessing, malicious-secure and multi-owner extensions, verifiable and index-hiding variants, and post-quantum instantiations.

REFERENCES

[1] L. Yang, J. Chang, Y. Zhang, and Y. Liu, "The hybrid-encryption-based data sharing scheme with keyword-based auditing function in cloud storage setting," *Cluster Computing*, vol. 28, 2025.

[2] T. O. Adesina, "Enhancing data security and privacy in multi-cloud environments using advanced encryption and access control mechanisms," Ph.D. dissertation, National University, 2026.

[3] H. N. B. Manjyanaik, R. Mohanty, and J. M. Kannan, "Preserving confidential data using improved rivest-shamir adleman to secure multi-cloud." *International Journal of Intelligent Engineering & Systems*, vol. 17, no. 4, 2024.

[4] M. Islam and R. Palit, "A keyword based searching and sharing scheme on the encrypted cloud data," in *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*. IEEE, 2023, pp. 1–6.

[5] D. P. K. Reddy, K. Anjaneyulu, K. Yuvaraj, and L. Joseph, "Encrypted cloud storage approach for a multicloud environment using fog computing," in *2022 Second International Conference on Computer Science, Engineering and Applications (ICCSEA)*. IEEE, 2022, pp. 1–6.

[6] M. Q. Alsudani, H. F. Fakhruldeen, H. A.-J. Al-Asady, and F. I. Jabbar, "Storage and encryption file authentication for cloud-based data retrieval," *Bulletin of Electrical Engineering and Informatics*, vol. 11, no. 2, pp. 1110–1116, 2022.

[7] S. R. C. C. T. Tadi, "Evaluating blockchain and homomorphic encryption for secure data processing in multi-cloud hybrid database systems: A systematic literature review."

[8] N. Bharot, N. Mehta, J. G. Breslin, and P. Verma, "Cloudlock: secure data sharing using a hybrid cryptosystem in multi-cloud data storage," *Cluster Computing*, vol. 28, no. 7, p. 464, 2025.

[9] S. Muthuvel, S. Priya, and K. Kumar, "Design of a multi-layered privacy-preserving architecture for secure medical data exchange in cloud environments," *Scientific Reports*, vol. 16, 2026.

[10] N. Joshi, K. Sambrekar, A. Patankar, A. Rajawat, and M. Muqeem, "Advanced security and privacy in cloud computing: Enhancing data protection with multikeyword ranked search in encrypted environments," *Scalable Computing: Practice and Experience*, vol. 26, pp. 467–489, 2025.

[11] A. Kamble, M. M. Jiet, and C. Puri, "Homomorphic encryption and its applications in multi-cloud security," in *2024 International Conference on Inventive Computation Technologies (ICICT)*. IEEE, 2024, pp. 1493–1499.

[12] S. Halder and T. Newe, "Enabling secure time-series data sharing via homomorphic encryption in cloud-assisted iiot," *Future Generation Computer Systems*, vol. 133, 2022.

[13] L. Wang, Z. Yang, and X. Song, "Shamc: A secure and highly available database system in multi-cloud environment," *Future Generation Computer Systems*, vol. 105, pp. 873–883, 2020.

[14] A. Prakash, C. C. Vignesh, M. Suresh, J. Pavan, S. S. Ahamad, and M. Rama, "Homomorphic encryption and prim's process for multi-keyword exploration in encrypted cloud data for trades," in *2023 1st International Conference on Innovations in High Speed Communication and Signal Processing (IHCSP)*. IEEE, 2023, pp. 432–436.

[15] W. Xu, J. Zhang, Y. Yuan, and Z. Li, "Privacy-preserving multi-cloud based dynamic symmetric searchable encryption," in *2021 2nd International Conference on Computer Communication and Network Security (CCNS)*. IEEE, 2021, pp. 176–181.

[16] R. Du, H. Jiang, and M. Li, "Lightweight searchable encryption with small clients on edge cloud," in *2022 17th Asia Joint Conference on Information Security (AsiaJCIS)*, 2022, pp. 41–48.

[17] W. Xu, J. Zhang, Y. Yuan, and X. Wang, "Symmetric searchable encryption with supporting search pattern and access pattern protection in multi-cloud," *Concurrency and Computation: Practice and Experience*, vol. 35, no. 9, p. e7651, 2023.

[18] X. Zhang, C. Huang, Y. Su, and J. Qin, "Secure, dynamic, and efficient keyword search with flexible merging for cloud storage," *IEEE Transactions on Services Computing*, vol. 17, no. 5, pp. 2822–2835, 2024.

[19] N. Yang, C. Tang, Q. Zhou, and D. He, "Dynamic consensus committee-based for secure data sharing with authorized multi-receiver searchable encryption," *IEEE Transactions on Information Forensics and Security*, vol. 18, pp. 5186–5199, 2023.

[20] J. Li, H. Yan, and Y. Zhang, "Efficient identity-based provable multi-copy data possession in multi-cloud storage," *IEEE Transactions on Cloud Computing*, vol. 10, no. 1, pp. 356–365, 2022.

[21] W. Li, W. Susilo, C. Xia, L. Huang, F. Guo, and T. Wang, "Secure data integrity check based on verified public key encryption with equality test for multi-cloud storage," *IEEE Transactions on Dependable and Secure Computing*, vol. PP, pp. 1–15, 2024.

[22] C. Xu, N. Wang, L. Zhu, K. Sharif, and C. Zhang, "Achieving searchable and privacy-preserving data sharing for cloud-assisted e-healthcare system," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8345–8356, 2019.

[23] D. Wu, C. Nie, Z. Yang, P. Zhang, and R. Wang, "Distributed secure data sharing method for internet of vehicles based on homomorphic hash tree," *IEEE Transactions on Vehicular Technology*, vol. 73, no. 11, pp. 17 572–17 585, 2024.

[24] A. K. Verma, P. Patel, V. S. Bhatia, A. Rastogi, A. Prasad, and P. Ranjan, "Secure federated multi-party computation with homomorphic encryption and blockchain for preserving privacy in cloud storage of medical records," in *2025 2nd International Conference on Intelligent Algorithms for Computational Intelligence Systems (IACIS)*. IEEE, 2025, pp. 1–6.

[25] O. M. Ijiga, N. Okika, S. A. Balogun, O. J. Agbo, and L. A. Enyejo, "Recent advances in privacy-preserving query processing techniques for encrypted relational databases in cloud infrastructure," *Journal of Cloud Computing*, vol. 10, no. 2, pp. 102–119, 2025.

[26] D. Horkos and L. Perret, "SoK: On cryptography for multi-cloud storage," Cryptology ePrint Archive, Paper 2026/207, 2026. [Online]. Available: https://eprint.iacr.org/2026/207

[27] O. Goldreich, S. Goldwasser, and S. Micali, "How to construct random functions," in *providing sound foundations for cryptography: On the Work of Shafi Goldwasser and Silvio Micali*, 2019, pp. 241–264.

[28] N. Gilboa and Y. Ishai, "Distributed point functions and their applications," in *Annual International Conference on the Theory and Applications of Cryptographic Techniques*. Springer, 2014, pp. 640–658.

[29] E. Boyle, N. Gilboa, and Y. Ishai, "Function secret sharing: Improvements and extensions," in *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, 2016, pp. 1292–1303.

[30] J. Fan and F. Vercauteren, "Somewhat practical fully homomorphic encryption," *Cryptology ePrint Archive*, 2012.

[31] J. H. Cheon, A. Kim, M. Kim, and Y. Song, "Homomorphic encryption for arithmetic of approximate numbers," Cryptology ePrint Archive, Paper 2016/421, 2016. [Online]. Available: https://eprint.iacr.org/2016/421

[32] O. Goldreich and R. Ostrovsky, "Software protection and simulation on oblivious rams," *Journal of the ACM (JACM)*, vol. 43, no. 3, pp. 431–473, 1996.

[33] G. A. Stafford, "Environmental sensor telemetry data," Kaggle Datasets, July 2020, version 1.0. https://www.kaggle.com/datasets/garystafford/environmental-sensor-data-132k.