SECURE AND PRIVACY-PRESERVING DATA AGGREGATION IN WIRELESS SENSOR NETWORKS

by

Aishah Aseeri

A dissertation submitted to the Faculty of the University of Delaware in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Computer Science

Winter 2022

SECURE AND PRIVACY-PRESERVING DATA AGGREGATION IN WIRELESS SENSOR NETWORKS

by

Aishah Aseeri

Approved: _____
Rudolf Eigenmann, Ph.D.
Interim Chair for Computer and Information Sciences

Approved: _____
Levi T. Thompson, Ph.D.
Dean of the College of Engineering

Approved: _____
Louis F. Rossi, Ph.D.
Vice Provost for Graduate and Professional Education and
Dean of the Graduate College

I certify that I have read this dissertation and that in my opinion it meets the academic and professional standard required by the University as a dissertation for the degree of Doctor of Philosophy.

Signed: _____

Rui Zhang, Ph.D.
Professor in charge of dissertation

I certify that I have read this dissertation and that in my opinion it meets the academic and professional standard required by the University as a dissertation for the degree of Doctor of Philosophy.

Signed: _____

Chien-Chung Shen, Ph.D.
Member of dissertation committee

I certify that I have read this dissertation and that in my opinion it meets the academic and professional standard required by the University as a dissertation for the degree of Doctor of Philosophy.

Signed: _____

Xing Gao, Ph.D.
Member of dissertation committee

I certify that I have read this dissertation and that in my opinion it meets the academic and professional standard required by the University as a dissertation for the degree of Doctor of Philosophy.

Signed: _____

Linke Guo, Ph.D.
Member of dissertation committee

ACKNOWLEDGEMENTS

First and the foremost, I would like to express my sincere gratitude to my PhD advisor Prof. Rui Zhang for his guidance, trust, encouragement, and support throughout my PhD study. His deep knowledge and strong logical thinking have been of great value for me. It was his enlightening supervision that broadened my views of the research area and helped me learn how to read research papers, identify and formulate research problems, come up with ideas, and develop ideas into a full research paper. He is always there whenever I need him and always asks inspiring questions and provide insightful discussions to guide me through my research. It was a wonderful and rewarding journey to learn from him. I am fortunate to have Prof. Zhang as my PhD advisor and hope to be as good an advisor as him when nurturing my own students in the future.

I also take this opportunity to thank the committee members, Prof. Chien-Chung Shen, Prof. Linke Guo, and Prof. Xing Gao for their valuable feedback on my dissertation and helpful advice and suggestions in general.

Many thanks to my dear colleagues at INS laboratory, Yukun Dong, Yidan Hu, Yunzhi Li, Zheyuan Liu, and Tianye Ma. Their friendship, help, and support have made my PhD study a wonderful journey. Special thanks to Yidan Hu for her constant support and encouragement over the years.

Finally, I reserve my most special appreciation to my wonderful family. Firstly, to my parents who have given me countless love and unconditional support during my life whenever I was in success or loss, and also for their patience, encouragement and prayers. I hope everything I have done can make you both happy and healthy. Secondly, to my three lovely sisters as well as to my two wonderful brothers, for their love, encouragement, emotional and moral support. Finally and most of all to my dear

husband who dedicated his unconditional love and support and have made countless sacrifices without a word of complaint to make my dream come true. Also, to my little angel daughter who understood my situation despite her young age and was a good helper for me through out my study journey.Without you both, I could not imagine how could I accomplish this work. I am really indebted and grateful to you and hope I can give you both as much love, care and support as you gave me...This dissertation is dedicated to my whole family...

Last but not least, thanks and acknowledgement is due to the National Science Foundation for the support given to complete this research.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

Wireless sensor networks (WSNs) are widely expected to play an important role in future IoT-power smart cities, which are expected to have all kinds of embedded sensors continuously sensing the city space and generating an unprecedented volume of heterogeneous data. Since blindly collecting all raw sensed data from sensor nodes will incur significant communication and computation overhead and quickly drain sensor nodes' batteries, data aggregation is widely regarded as a key enabling functionality for WSNs but nevertheless faces various security and privacy challenges. Despite the large body of literature on secure and privacy-preserving data aggregation, this dissertation aims to identify new security attacks on data aggregation and develop novel secure and privacy-preserving data schemes to support complex aggregation functions. First, we introduce a novel enumeration attack against existing secure additive aggregation schemes. While secure additive aggregation such as Sum and Average has been studied extensively in the past, none of the existing solutions were designed to detect or defend against compromised sensor nodes forging their own readings, as it is widely assumed that a small number of compromised sensor nodes forging their own reading has very limited impact on the final aggregation result. We take VMAT, a representative secure additive aggregation scheme, as an example to show that this long-held assumption does not hold. Specifically, the enumeration attack allows a small number of compromised sensor nodes to significantly inflate the final aggregation result by selectively forging their own readings. We also introduce an effective defense against the enumeration attack and confirm its effectiveness by simulation studies.

Second, we study the problem of secure quantile summary aggregation. A quantile summary allows a base station to extract the ϕ-quantile for any $0 < \phi < 1$ of all the sensor readings in the network and can provide a more accurate characterization

of the data distribution than simple statistics such as sum and average. While efficient quantile summary aggregation has been studied in the past, there has been no solution for secure quantile summary aggregation. To tackle this open challenge, we first experimentally study the impact of several malicious attacks on quantile summary aggregation and then introduce a novel secure quantile summary aggregation protocol built upon efficient cryptographic primitives.

Finally, we study the problem of privacy-preserving quantile summary aggregation. Privacy-preserving data aggregation is needed when the data generated by sensor nodes, which allows the base station to learn useful aggregates of sensed data while ensuring data privacy for individual sensors. Similar to the lack of a secure quantile summary aggregation solution, how to realize privacy-preserving quantile summary aggregation remains unknown. To fill this void, we design a novel scheme to enable efficient quantile summary aggregation while guaranteeing local differential privacy for individual sensors and use simulation studies to confirm its effectiveness.

Chapter 1

INTRODUCTION

The emerging IoT paradigm is expected to be powered by millions of sensors deployed throughout our physical space that continuously sense the surrounding environment and generate valuable data to optimize and improve our decision making. Wireless sensor networks (WSNs) is widely regarded as a key component of many IoT applications. A typical WSN is a multi-hop wireless network consisting of a base station and many sensor nodes, in which sensor nodes continuously generate sensed data and forward them to the base station. WSNs are ideal for applications that require monitoring and controlling assets from a distance in real-time, and with minimal human intervention. Applications of WSNs include industrial automation, wildlife monitoring, smart agriculture, environmental monitoring, and so on.

In-network aggregation mechanism is one of the methods used to reduce the overall amount of power and bandwidth required to process a query in data gathering. It allows sensor values to be gradually processed by intermediate nodes along the route to the base station. Many effective types of data aggregation functions were explored by researchers to improve the way to compute statistic aggregates in wireless sensor networks using in-network aggregation. These functions are closely related to sensor network applications; such as MAX/MIN, COUNT, SUM, AVERAGE, QUANTILE, MEDIAN.

However, due to the constrains on resources of a sensor node, an attacker may compromise a sensor node to falsify sensor readings, improperly apply an aggregation function and drop legitimate messages from the aggregate result and further get the base station to accept the incorrect result. Also, an attacker may get access to sensitive data of other sensor nodes. Therefore, security and privacy are among the most

challenging obstacles to the wide spread of WSNs deployment especially for critical applications. Secure data aggregation in wireless sensor networks has been studied extensively in the past [2, 3, 4, 5, 6, 7, 8, 9, 10]. Also, privacy-preserving data aggregation in sensor networks has received a lot of attention [11, 12, 13, 14, 15, 16, 17]. Despite the extension of research in the security and privacy of wireless sensor networks, in-network data aggregation in WSNs still faces several critical security and privacy challenges. This dissertation addresses three security and privacy related problems in WSNs as explained below.

1.1 Secure SUM aggregation against Enumeration attack

SUM aggregation is a key function for data aggregation in many applications of WSNs. Secure data aggregation in WSNs including SUM aggregation has been studied extensively in the past. Most of the research efforts [18, 2, 3, 4, 5, 6, 19, 8, 9, 10] have focused on detecting intermediate nodes manipulating partial aggregation results. There is a general consensus [2, 3, 20, 10] that a compromised node forging its own reading is fundamentally difficult to detect but has limited impact on robust aggregation functions such as SUM and COUNT [21]. In this dissertation, we introduce a novel enumeration attack against approximate SUM aggregation to show that this long-held assumption does not hold. Specifically, the enumeration attack allows a small number of compromised sensor nodes to significantly inflate the final aggregation result by selectively forging their own readings. We theoretically analyze the impact of enumeration attack and validate our analysis using simulation studies. Also, we introduce an effective countermeasure against enumeration attack by requiring every sensor node to commit to its reading prior to knowing the random seed for generating random synopsis.

1.2 Secure Quantile aggregation Summaries

While many secure data aggregation schemes have been proposed in the literature, most of them target simple statistics such as Sum, Count, Min/Max, and

Medium. In contrast, a quantile summary allows a base station to learn the ϕ-quantile for any $0 < \phi < 1$ of all the sensor readings in the network and can provide a more accurate characterization of the data distribution. Quantile summary aggregation in wireless sensor networks has been studied by the distributed computing community [22, 23, 24, 1, 25]. Unfortunately, none of the mentioned quantile aggregation schemes have any security provisions. In this dissertation, we fill this void by first evaluating the impact of a range of attacks on quantile summary aggregation using simulation and then introduce a novel secure quantile summary aggregation protocol for wireless sensor networks. Our proposed protocol is based on the quantile summary aggregation protocol proposed by Huang et. al.[1]. Detailed simulation studies confirm the efficacy and efficiency of the proposed protocol.

1.3 Local differential private Quantile aggregation Summaries

Privacy preserving data aggregation is another major challenge facing WSNs especially with the use of in-network aggregation mechanism where data is collected and processed by intermediate nodes. In many scenarios, such data may include sensitive or critical information. Therefore, privacy preserving data aggregation has been extensively studied in the past covering several kinds of aggregation queries in WSNs [11, 12, 13, 14, 15, 16, 17]. However, none of the published work so far have tackled the privacy issue in quantile summary aggregation. In this dissertation, we study the problem of privacy-preserving quantile summary aggregation and design a novel scheme to enable efficient quantile summary aggregation while guaranteeing local differential privacy for individual sensors. We further show the effectiveness of the proposed scheme through simulation studies.

1.4 Organization

The remainder of this dissertation is structured as follows. In chapter 2, we introduce enumeration attack against SUM aggregation, and accordingly propose a defense and confirm its effectiveness through experiments. chapter 3 introduces a novel

secure quantile summary aggregation protocol for wireless sensor networks followed by detailed simulation studies to confirm the efficacy and efficiency of the proposed protocol. In chapter 4, we describe and show the general steps to develop a novel solution that enables efficient quantile summary aggregation while guaranteeing local differential privacy for individual sensors and follow that with simulation studies to evaluate the scheme performance. We finally conclude our work in chapter 5.

Chapter 2

SECURE SUM AGGREGATION AGAINST ENUMERATION ATTACK

2.1 Introduction

Wireless sensor networks play a key role in the emerging IoT paradigm where millions of sensors are expected to be deployed throughout the physical space, which continuously sense the surrounding environment and generate an unprecedented amount of data. A typical wireless sensor network is a multi-hop wireless network formed by many resource-constrained sensor nodes and a base station, where sensed data are forwarded to the base station with Internet connectivity via intermediate sensor nodes. Exemplary applications of wireless sensor networks include manufacture plant monitoring, asset tracking, traffic monitoring, environmental monitoring, public safety, and so on [26].

In-network data aggregation [27, 28] is a key functionality in wireless sensor networks and refers to the process in which the sensed data are processed and aggregated en-route by intermediate sensor nodes. Since sensor nodes are commonly battery powered with limited communication and computation resources, forwarding every sensor reading to the base station would quickly deplete the energy of intermediate nodes. In-network data aggregation allows the base station to learn statistic aggregates of the sensed data while greatly reducing the energy consumption and prolonging the network's lifetime. Consider the SUM aggregation as an example. Sensor nodes first form an aggregation tree rooted at the base station. During the aggregation process, every node sums up the readings from its children and its own and forwards the partial sum to its parent. The base station is able to obtain the sum of all readings at the end of the process. Other common aggregate functions such as MAX/MIN, COUNT, and AVERAGE can be realized in a similar fashion.

As an important network primitive, in-network data aggregation faces several critical security challenges. Since sensor nodes are resource-constrained, they may be physically captured or compromised by attackers and instructed to launch various attacks. For example, a compromised sensor node may modify its partial aggregation result to significantly inflate or deflate the final aggregation result at the base station. Second, even if the base station is able to detect and reject the false aggregation result, a compromised sensor node can launch persistent attack to prevent the base station from receiving correct aggregation result, leading to a special form of Denial-of-service attack. Last but not the least, a compromised sensor node may report arbitrary reading of its own while following the aggregation protocol.

Secure data aggregation in wireless sensor networks has been studied extensively in the past. A common assumption held in the literature is that a single compromised sensor node forging its own reading is fundamentally difficult to detect but has limited impact on the final aggregation result for robust aggregation functions like SUM and COUNT [21]. Most of the research efforts have focused on detecting intermediate node manipulating partial aggregation result. Existing solutions can be broadly classified into two categories. The first category such as [2, 3, 4, 5] can provide accurate aggregation results and detect malicious sensor nodes manipulating partial aggregation results via commitment verification. The second category such as [6, 19, 29, 8, 9, 10] offers statistical estimation of the aggregate result via probabilistic sampling. As mentioned above, a single malicious sensor node can keep attacking the aggregation process to prevent the base station from obtaining the correct aggregate. There are a very few attempts addressing the identification and revocation of compromised nodes with VMAT [9] being a representative. VMAT relies on verifiable MIN aggregation and converts other additive aggregation functions such as SUM and COUNT into MIN aggregation via verifiable sampling.

In this chapter, we introduce a novel enumeration attack against VMAT [9] to highlight the vulnerability of converting additive aggregation functions to MIN aggregation via probabilistic sampling. We observe that a compromised sensor node can

exploit the vulnerability probabilistic sampling by enumerating all possible readings to find the one that leads to significantly inflated aggregation result. In other words, the long-held view that a single compromised node falsifying its local value has limited impact on final aggregation results does not always hold. While VMAT has incorporated a verifiable random number generation mechanism to prevent compromised sensor nodes from generating arbitrary random samples, we show that such mechanism is necessary but inadequate. As a countermeasure, we also introduce an effective defense against the enumeration attack. Our contributions in this chapter can be summarized as follows.

- We introduce a novel enumeration attack against VMAT to highlight the danger of converting additive aggregation into MIN aggregation, whereby a small number of compromised sensors could severely manipulate the final aggregation result.
- We theoretically analyze the impact of enumeration attacks and validate our analysis using simulation studies.
- We also introduce an effective countermeasure against enumeration attacks by requiring every sensor node to commit to its reading prior to knowing the random seed for generating random synopsis. We confirm the efficacy and efficiency of the countermeasure via simulation studies.

The rest of this chapter is structured as follows. Section 2.2 discusses the related work. Section 2.3 presents the network and adversary models. Section 2.4 reviews the VMAT scheme. Section 2.5 presents the enumeration attack and its evaluation. Section 2.6 presents a defense against the enumeration attack and evaluates its performance. Section 2.7 finally concludes this work.

2.2 Related Work

Secure data aggregation in wireless sensor networks and related systems has been studied extensively in the past.

Existing solutions can be generally classified into two categories. The first category such as [2, 3, 4, 5] provides accurate aggregation result at the base station. Most

of these schemes [2, 3, 4] ensure aggregation-result integrity by requiring intermediate nodes to commit to partial aggregation-results through cryptographic means. For example, the scheme introduced in [2] requires each intermediate node to generate a commitment using a Merkle hash tree, which is then forwarded along with the aggregated data to its parent node. Each node adds its reading to the aggregated data will later on send an authentication code to the base station which increases transmission and communication overhead. Accordingly, authors in [4] modifies this scheme to reduce the communication per node by designing a new commitment structure for authentication. SDAP [3] is a secure hop-by-hop data aggregation protocol that can tolerate more than one compromised node through divide-and-conquer in addition to a commitment-and-attest principle to help the base station to verify the correctness of the aggregated data. SIES [5] on the other hand explores homomorphic encryption to detect intermediate nodes modifying partial aggregation result. However, it cannot isolate these nodes which make it vulnerable to denial-of-service attack. The second category such as [6, 19, 8, 9, 10] aims to provide statistical estimation of the aggregate result with probabilistic guarantee. SIA [6] considers a single-aggregator model and statistically detects false aggregation result via random sampling and interactive proof, which is subsequently improved in [19] to realize secure approximate-median aggregation. This scheme addresses integrity but lacks confidentiality. A secure aggregation scheme based on verifiable set sampling was introduced in [8] to compute Count and Sum. This scheme does not only detect malicious nodes but also tolerate them which make it resilient to certain kinds of (Dos) attacks. Synopsis diffusion [29] is a robust aggregation framework against packet loss which also computes Count and Sum. It explores multi-path routing and duplicate-insensitive aggregation, and it is improved in [20] to enable detection of false subaggregate and [10] to tolerate false subaggregate.

While most of the these solutions [18, 2, 3, 4, 5, 6, 19, 8, 9, 10] focus on detecting intermediate node manipulating partial aggregation result, there are a few attempts aiming at identifying compromised nodes during data aggregation in addition to VMAT

[9], which make them resilient to (Dos) attacks. Early proposals [30, 31] rely on expensive public-key cryptography operations and group testing to identify malicious nodes. Xu *et al.* [32] proposed an improvement for SDAP [3] to identify malicious nodes via statistical abnormality detection and random node grouping. Their scheme is ineffective if the attacker adopts its behavior according to the statistical detection rules. In [33], a secure aggregation scheme was introduced to pinpoint intermediate nodes that drop partial aggregation results. The approach, unfortunately, incurs a communication overhead linear to the total number of sensor nodes, which largely nullifies the benefit of in-network aggregation. In [34], Li *et al.* introduced a secure SUM aggregation protocol to misbehaving intermediate aggregators by having every intermediate node's partial aggregation result checked by its children and parent, which is ineffective against two colluding parent and child nodes. In addition, there is a general consensus [2, 3, 20, 10] that a compromised node forging its own reading is fundamentally difficult to detect but has limited impact on robust aggregation functions such as SUM and COUNT [21].

2.3 Network and Adversary Models

In this section, we introduce our system and adversary models.

2.3.1 Network Model

We consider a multi-hop wireless sensor network comprising a base station and n sensor nodes. Each sensor node i has a sensed reading d_i in the range $\{1, \ldots, k\}$. The base station intends to learn $f(d_1, \ldots, d_n)$, where $f(\cdot, \ldots, \cdot)$ is some aggregation function such as MAX/MIN, SUM, AVERAGE, and COUNT. The aggregation is performed over an aggregation tree, which is the directed tree rooted at the base station formed by the unique path from every sensor node to the base station.

2.3.2 Adversary Model

We assume that the base station has adequate computation and energy resources and is safeguarded from possible attacks. In contrast, sensor nodes are constrained in computation and communication resources and may be compromised by the attacker,

e.g., through physical capture. Once compromised, all the information stored at the sensor node such as cryptographic keys is revealed to the attacker. The attacker aims to have the base station accept a significantly inflated aggregation result without being detected. We consider the following two attacks in this chapter.

- A compromised node may falsify its own sensed reading, which may or may not be in the valid reading range.
- A compromised node may modify or drop a partial aggregation result.

We further assume that the attacker can compromise up to c sensor nodes and that all the compromised nodes can collude in an arbitrary fashion under the instruction of the attacker. We focus on the attacks targeting data aggregation in this chapter and refer to the rich literature (e.g., [35, 36, 37, 38, 39, 40]) for other possible attacks on wireless sensor networks.

2.4 Review of VMAT

In this section, we briefly review the VMAT scheme and how to convert additive aggregation functions into MAX aggregation.

VMAT [9] is a representative secure aggregation scheme built upon efficient symmetric-key operations with the capabilities of pinpointing and revoking malicious nodes. These two features added to the support of multi-path aggregation distinguish this scheme from other proposed schemes for reducing communication overhead and preventing different attacks including denial of service attacks. Under VMAT, each node shares one or multiple secret keys, called edge keys, with each of its neighbor, and a distinct secret key with the base station. The key component of VMAT is a secure MIN aggregation scheme. During the aggregation phase, each sensor node creates a message consisting of its node ID, sensor reading, and a MAC encrypted with an edge key shared with its parent. Each intermediate node receives the messages from its children and forwards the message with the smallest reading among its children and itself. At the end of the aggregation phase, the base station obtains the minimal

reading among all sensor nodes and verifies whether this minimal reading has a valid MAC. During the confirmation phase, the base station uses authenticated broadcast to announce the minimum value it received. If the minimum value is higher than the true minimal value, then the sensor node with the true minimal value can detect it and issue a veto message to be flooded back to the base station. The base station can then revoke one of the edge keys used by the reporting sensor node. A sensor will be revoked from the network after excluding certain number of its edge keys. We refer readers to [9] for more details of the secure MIN aggregation protocol.

VMAT explores the distributed randomized algorithm proposed in [41] to convert additive aggregation such as SUM and COUNT into MIN aggregation. Consider SUM aggregation as an example. To compute $S = \sum_{i=1}^{n} d_i$, each node i with reading d_i generates m mutually independent random synopses $s_{i,1}, s_{i,2}, \ldots, s_{i,m}$ from an exponential distribution $\mathsf{Exp}(d_i)$ with mean $1/d_i$. All n sensor nodes then participate in m parallel instances of secure MIN aggregation to allow the base station to obtain $s_1^{\min}, s_2^{\min}, \ldots, s_m^{\min}$, where $s_j^{\min} = \min(s_{1,j}, s_{2,j}, \ldots, s_{n,j})$ for all $1 \leq j \leq m$. The sum of all d_i can then be estimated as

$$\hat{S} = \frac{m}{\sum_{j=1}^{m} s_j^{\min}},$$

which has been shown [41] to be an unbiased estimator of S. In addition, when $m = \Theta(\frac{1}{\epsilon^2} \log \frac{1}{\delta})$, $\hat{S}$ is within $((1-\epsilon)S, (1+\epsilon)S)$ with probability at least $1-\delta$. AVERAGE and COUNT aggregates can be realized in a similar fashion.

To prevent a compromised node from generating arbitrarily small synopsis, VMAT uses a deterministic pseudorandom number generator to ensure that any synopsis must correspond to a valid reading in range. In particular, the deterministic pseudorandom number generator takes the sensor reading d_i, node ID i, and a nonce r as input and outputs m synopsis $s_{i,1}, \ldots, s_{i,m}$. On receiving $s_1^{\min}, s_2^{\min}, \ldots, s_m^{\min}$, the base station can verify that every minimal synopsis is indeed generated from a valid reading. Unfortunately, we will show in the next section that this mechanism alone is necessary but inadequate.

2.5 Enumeration Attack

In this section, we use SUM aggregation as an example to introduce a novel data enumeration attack.

2.5.1 Attack

In enumeration attack, a compromised sensor node aims to inflate the final aggregate at the base station. In comparison to the naive attack in which a compromised node simply reports the maximum reading in range, enumeration attack is more effective by causing the aggregation result significantly deviating from the true aggregation result.

Enumeration attack exploits the vulnerability that a compromised sensor node can report arbitrary reading of its own. Recall that in VMAT, every node i with reading d_i generates m independent synopsis from an exponential distribution with mean $1/d_i$, and the aggregation result is computed from the m minimal synopsis across all the sensor nodes. Recall that a valid sensed reading is in the range $\{1, \ldots, k\}$. If the sensor node simply reports the maximum reading k, each of its m synopsis is an exponential random variable with mean $1/k$. In enumeration attack, a compromised sensor node attacks one synopsis of its choice. Consider as an example that a compromised sensor node i attacks synopsis $s_{i,1}$. Node i can compute one synopsis for each possible reading $1, \ldots, k$ using the verifiable random number generator $\mathsf{DRNG}(s, d, ID, k)$ to find the reading d^* that leads to the smallest synopsis s_d as

$$d^* = \arg\min \mathsf{DRNG}(s, d, ID, k).$$

It then faithfully participates in the secure MIN aggregation with s_d.

We say the enumeration attack succeeds if s_d happens to be smaller than all the synopsis s_{j_1} generated by non-compromised sensor nodes. It is easy to see that under enumeration attack, the synopsis $s_{i,1}$ is the minimal of k independent exponential random variables with means $1, 1/2, \ldots, 1/k$, respectively, which is smaller than the one

generated from maximum reading k with high probability. In other words, enumeration attack allows a sensor node to generate a much smaller synopsis with high probability.

Multiple compromised sensor nodes can collude to maximize the impact of the enumeration attack. In particular, if the attacker has $c > 1$ sensor nodes, the attacker can instruct each compromised sensor node to attack one distinct synopsis or evenly allocate the compromised sensor nodes across m synopsis if $c > m$. In the worst case, if enumeration attack succeeds for every synopsis, then the final aggregation result computed by the base station is independent from any of the non-compromised sensor nodes' reading.

2.5.2 Theoretical Analysis

We first analyze the probability that a single compromised sensor node can succeed in launching enumeration attack. Without loss of generality, we consider one compromised sensor node i and g non-compromised sensor nodes and assume that node i intends to attack synopsis $s_1^{\min}$. We have the following theorem regarding the success probability of a single node attacking one synopsis.

Theorem 1. *Assume that there are g non-compromised sensor nodes. Further assume that the readings of non-compromised sensor nodes are i.i.d. random variables with probability distribution $\mathsf{Pr}(d_j = x) = p_x$ where $1 \leq x \leq k$. The probability that a single compromised node can successfully launch enumeration attack against a single synopsis is given by*

$$P_{succ} = \int_0^\infty \lambda e^{-\lambda t} \cdot (\sum_{y=1}^{k} p_y e^{-yt})^g dt. \tag{2.1}$$

Proof. Without loss of generality, assume that a compromised sensor node i aims to attack synopsis $s_1^{\min}$. The enumeration attack succeeds if node i can find a reading $d_i \in \{1, \ldots, k\}$ that results in its synopsis $s_{i,1}$, being the minimum among all $s_{1,j}, \ldots, s_{n,j}$. Let s_{em} be the synopsis generated by node i under enumeration attack. We can see that

$$s_{\text{em}} = \min(s[1], s[2] \ldots, s[k]),$$

where $s[1], s[2] \ldots, s[k]$ are mutually independent exponential distributed random variables with means $1, 1/2, \ldots, 1/k$, respectively. It follows that s_{em} is an exponential random variable with p.d.f.

$$f(s_{\text{em}} = t) = \lambda e^{-\lambda t}$$

for $t \geq 0$, where $\lambda = k(k+1)/2$.

Assume that there are g non-compromised sensor nodes. Let $s_{j,1}$ be the synopsis generated by a non-compromised sensor node j. It follows that

$$\begin{aligned}
\mathsf{Pr}(s_{j,1} \leq t) &= \sum_{x=1}^{k} \mathsf{Pr}(s_{j,1} \leq t | d_j = x) \cdot \mathsf{Pr}(d_j = x) \\
&= \sum_{x=1}^{k} (1 - e^{-xt}) p_x \\
&= 1 - \sum_{x=1}^{k} p_x e^{-xt}.
\end{aligned}$$

Let $s_g^{\min}$ be the minimal synopsis among g non-compromised sensor nodes. We have

$$\begin{aligned}
\mathsf{Pr}(s_g^{\min} \leq t) &= 1 - \mathsf{Pr}(s_g^{\min} > t) \\
&= 1 - \prod_{j=1}^{g} \mathsf{Pr}(s_{j,1} > t) \\
&= 1 - (\sum_{y=1}^{k} p_y e^{-xy})^g.
\end{aligned}$$

We finally have

$$\begin{aligned}
P_{\text{succ}} &= \mathsf{Pr}(s_{\text{em}} < s_g^{\min}) \\
&= \int_0^{\infty} \lambda e^{-\lambda t} \cdot \mathsf{Pr}(s_g^{\min} > t) dt \\
&= \int_0^{\infty} \lambda e^{-\lambda t} \cdot (\sum_{y=1}^{k} p_y e^{-yt})^g dt
\end{aligned}$$

□

We also have the following two theorems regarding the expected number of synopsis successfully attacked and the optimal strategy of allocating compromised nodes to synopsis.

Theorem 2. *Assume that there are c compromised sensor nodes. Suppose that the attacker allocates c_j nodes to attack the jth synopsis for all $1 \leq j \leq m$, where $\sum_{j=1}^{m} c_j = c$. The expected number of synopsis successfully attacked is given by*

$$\mathsf{E}(\hat{m}) = m - \sum_{j=1}^{m} (1 - P_{succ})^{c_j}, \tag{2.2}$$

where P_{succ} is given in Eq. (2.1).

Proof. For every $j, 1 \leq j \leq m$, define X_j as the indicator random variable such that $X_j = 1$ if synopsis S_j is successfully attacked and 0 otherwise. Since the attacker allocates c_j nodes to attack synopsis S_j, the probability that S_j is successfully attacked can be computed as

$$\begin{aligned} \mathsf{Pr}(X_j = 1) &= 1 - \mathsf{Pr}(X_j = 0) \\ &= 1 - (1 - P_{\text{succ}})^{c_j}. \end{aligned}$$

Let $\hat{m}$ be the number of number of synopsis successfully attacked. We have $\hat{m} = \sum_{j=1}^{m} X_j$. It follows that

$$\begin{aligned} \mathsf{E}(\hat{m}) &= \sum_{j=1}^{m} \mathsf{E}(X_j) \\ &= \sum_{j=1}^{m} \mathsf{Pr}(X_j = 1) \\ &= \sum_{j=1}^{m} (1 - (1 - P_{\text{succ}})^{c_j}) \\ &= m - \sum_{j=1}^{m} (1 - P_{\text{succ}})^{c_j}, \end{aligned} \tag{2.3}$$

where P_{succ} is given in Eq. (2.1). □

Theorem 3. *Assume that there are c compromised sensor nodes. An optimal attack strategy is to assign the compromised nodes to synopsis in a round robin fashion, i.e., assign the ith compromised node to attack the jth synopsis, where*

$$j = i \mod m.$$

Table 2.1: Default Simulation Settings

Para.	**Val.**	**Description.**
m	50	The number of synopsis
k	100	The readings range
n	100	The number of sensor nodes
c	1	The number of compromised nodes

Proof. Assume that there are c compromised sensor nodes. For every $j, 1 \leq j \leq m$, define X_j as the indicator random variable such that $X_j = 1$ if synopsis S_j is successfully attacked and 0 otherwise. Given total c compromised sensor nodes, the attacker seeks to maximize $\mathsf{E}(\hat{m})$, or equivalently $\sum_{j=1}^{m} \mathsf{Pr}(X_j = 1)$. Consider synopsis S_j as an example. The probability that S_j is successfully attacked is given by

$$\mathsf{Pr}(X_j = 1) = 1 - (1 - P_{\text{succ}})^{c_j}. \tag{2.4}$$

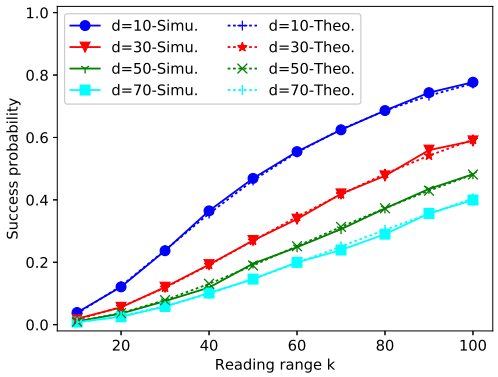
Now suppose that the attacker allocate one extra compromised sensor node to attack S_j. The probability that S_j is successfully attacked is given by

$$\mathsf{Pr}(X_j = 1) = 1 - (1 - P_{\text{succ}})^{c_j+1}. \tag{2.5}$$

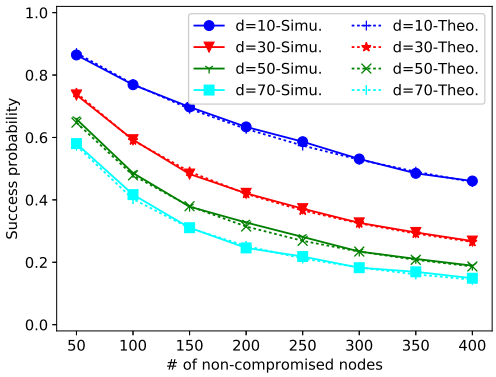
Subtracting Eq. (2.4) from Eq. (2.5), we can obtain the change in the probability caused by the additional one compromised node as

$$\begin{aligned} \triangle P &= 1 - (1 - P_{\text{succ}})^{c_j+1} - (1 - (1 - P_{\text{succ}})^{c_j}) \\ &= P_{\text{succ}}(1 - P_{\text{succ}})^{c_j}. \end{aligned}$$

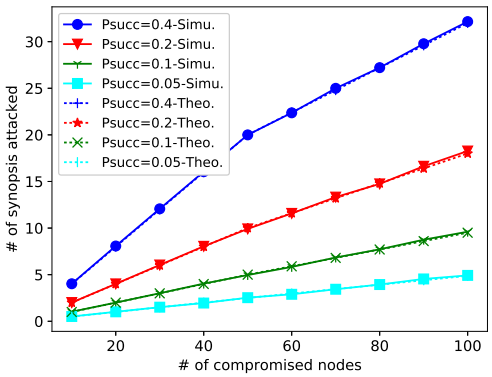
We can see that $\triangle P$ monotonically decreases as c_j increases. This shows that the attacker should always allocate compromised sensor node to attack the synopsis that has been assigned fewest nodes in order to maximize $\mathsf{E}(\hat{m})$. In other words, an optimal attack strategy is to assign the compromised nodes to synopsis in a round robin fashion. □



(a) P_{succ} vs. reading range



(b) P_{succ} vs. # of non-compromised nodes



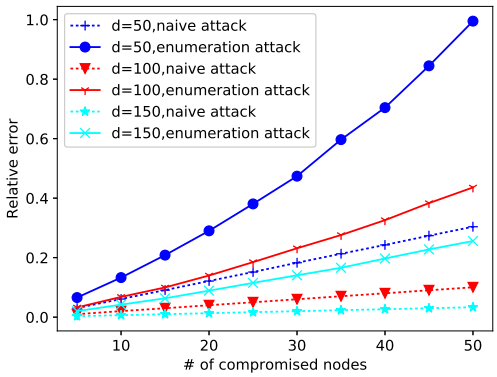
(c) $\hat{m}$ vs. # of compromised nodes

Figure 2.1: Success probability of enumeration attack, where $k = 100$, $n = 100$ and $m = 50$

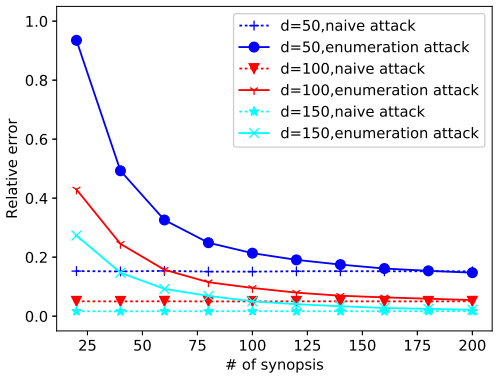
2.5.3 Simulation Results

We conduct simulation studies to validate our theoretical analysis. Specifically, we consider $n = 1000$ sensor nodes and $m = 50$ synopsis as the default setting and evaluate the impact of several parameters. Table 2.1 summarizes our default settings unless mentioned otherwise. We also consider four probability distributions of non-compromised nodes' readings. Every point is the average of 500 runs, each with a distinct random seed.

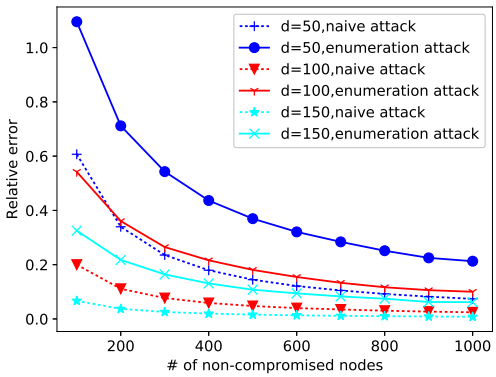
Figs. 2.1a to 2.1c illustrate the impact of valid reading range and the number of non-compromised nodes on P_{succ}, where we assume that the readings from non-compromised sensor nodes following four uniform distributions $\mathsf{U}(5, 15), \mathsf{U}(25, 35), \mathsf{U}(45, 55)$



(a) Aggregation error vs. # of compromised nodes



(b) Aggregation error vs. # of synopsis



(c) Aggregation error vs. # of non-compromised nodes

Figure 2.2: Comparison of enumeration attack and naive attack in estimation error, where $k = 200$, $n = 500$, $c = 25$, and $m = 50$

and $\mathsf{U}(65, 75)$ with mean $10, 30, 50$ and 70, respectively. First of all, we can see that the theoretical results match the simulation results very well, which validate our theoretical analysis. We can see from Fig. 2.1a that the success probability increases as the reading range increases. This is expected, as the larger the reading range, the more readings the compromised sensor node can try to find the minimal possible synopsis, the higher the probability that its synopsis is smaller than all the synopsis generated by the non-compromised sensor nodes, and vice versa. In addition, the larger the expectation of the non-compromised node's reading, the lower the success probability. This

is because it is more likely for non-compromised nodes to generate smaller synopsis with larger readings. We can see from Fig. 2.1b that the success probability decreases as the number of non-compromised nodes increases. This is also anticipated, as the more non-compromised nodes, the smaller the minimal synopsis among all the synopsis generated by the non-compromised nodes. Finally, we can see from Fig. 2.1c that the number of synopsis successfully attacked increases as the number of compromised sensor nodes increases. We can also observe that the pace of increasing slows down after the number of compromised nodes exceeds the number of synopsis.

Figs. 2.2a to 2.2c compares the relative estimation errors under enumeration attack and naive attack where every compromised sensor node simply reports the maximum reading in range. The relative estimation error is defined as $|\hat{S}_{\text{att}} - \hat{S}|/\hat{S}$, where $\hat{S}_{\text{att}}$ and $\hat{S}$ are the sums estimated by the base station under attack and under no attack, respectively. We assume that the average readings of non-compromised sensor nodes are 50, 100, and 150, respectively. We can see from Fig. 2.2a that the relative estimation error increases as the number of compromised nodes increases under both naive and enumeration attacks, which is anticipated. In addition, the relative estimation error under the naive attack is very limited, which is in line with the long-held view and conclusions in [21]. However, the relative estimation error under enumeration attack is always significantly higher than that under the naive attack. For example, enumeration attack can inflate the sum aggregation result by 40% and 100% with 25 and 50 compromised sensor nodes, respectively. Such large aggregation errors highlight the severe impact of the enumeration attack. Moreover, the larger the average reading of non-compromised nodes, the smaller the impact of both naive attack and enumeration attack. We can also see from Fig. 2.2b that the relative estimation error decreases as the number of synopsis increases. This is expected, as if the number of compromised nodes remains the same, the proportion of the synopsis successfully attacked decreases as the number of synopsis increases. When the number of synopsis exceeds 115, the relative estimation error under enumeration attack is about the same

as that under the naive attack. Finally, Fig. 2.2c shows that the aggregation error decreases as the number of non-compromised nodes increases. This is because the more non-compromised nodes results, the lower the success probability, the fewer synopsis successfully attacked, and vice versa.

2.6 Countermeasure

In this section, we introduce an effective countermeasure against the enumeration attack.

2.6.1 Countermeasure

We observe that the enumeration attack is possible because compromised nodes know the nonce used for generating synopsis before choosing its reading. An effective way to defend against enumeration attack is to require every sensor node to commit to its reading before knowing the nonce, so that there is no opportunity for compromised sensor nodes to enumerate all possible readings. Our countermeasure requires each node to commit to its reading and forward the commitment to selected witnesses in its neighborhood, which allows the base station to verify whether the synopsis is generated before the sensor node knowing the random seed. In what follows, we detail the operations.

During network initialization, every node i learns the IDs of all the nodes in its h-hop neighborhood, denoted by $\mathcal{N}^h(i)$, and the base station learns the complete topology of the network. To initiate a data aggregation process, the base station broadcasts a command with a random nonce s_1. On receiving the command, each sensor node i with reading d_i computes a commitment as

$$\mathsf{Commit}_i = \langle ID_i, d_i, MAC(ID_i||s_1||d_i)\rangle,$$

where $MAC(\cdot)$ denotes message authentication code computed using the secret key shared between node i and the base station and $||$ denotes concatenation. It selects λ nodes from $\mathcal{N}^h(i)$ to serve as its witnesses using a deterministic random number

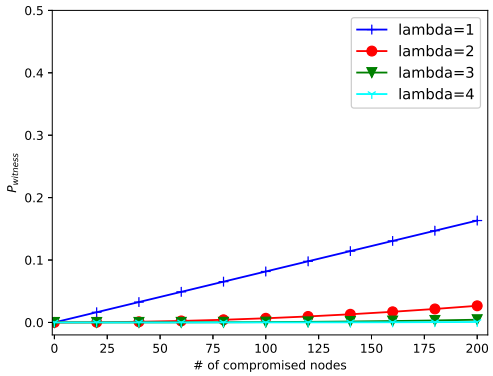
generator seeded by the nonce s_1 and its node ID, where $\lambda \geq 1$ is a system parameter. Node i then forwards Commit_i to each of the λ witnesses.

Every node then follows VMAT to generate m synopsis and participates in m instances of secure MIN aggregation. In particular, the base station broadcasts another nonce s_2. At the end of the aggregation phase, the base station obtains $s_1^{\min}, s_2^{\min}, \ldots, s_m^{\min}$, i.e., m minimal synopsis across all n sensor nodes. For every $s_j^{\min}(1 \leq j \leq m)$, the base station determines the ID of the node that generated this synopsis and verifies that $s_j^{\min}$ is indeed generated from a valid reading as in VMAT. Consider $s_j^{\min}$ as an example. Assume that node i with reading d_i generated $s_j^{\min}$. During the confirmation phase, the base station uses authenticated broadcast to announce $\langle ID_i, d_i, s_j^{\min} \rangle$ to all the nodes. Every witness of node i, say node w, then sends a message $\langle ID_w, \mathsf{Commit}_i, MAC(ID_w || \mathsf{Commit}_i) \rangle$ to the base station. On receiving the message, the base station first verifies whether node w is a valid witness for node i. If so, the base station verifies the MACs in the message and Commit_i. If the verification succeeds, the base station knows that node i's reading d_i was committed before knowing the nonce s_2.

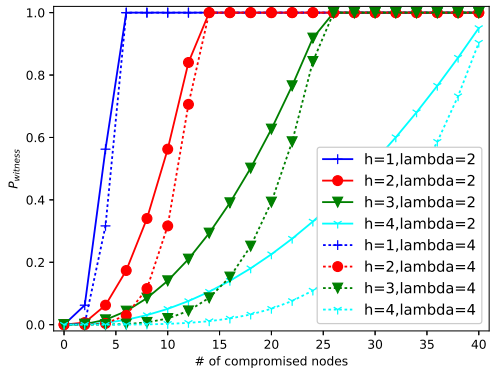
2.6.2 Simulation Results

We also use simulation studies to evaluate the performance of our countermeasure. We consider a 35×35 grid sensor network with $n = 1225$ sensor nodes, where the base station is located at one of the corner. Every sensor node (except the ones near the boundary) has 4 one-hop neighbors, 12 two-hop neighbors, 24 three-hop neighbors, and 40 four-hop neighbors. We measure the communication overhead incurred by our countermeasure as the average number of extra message transmissions per node and per synopsis.

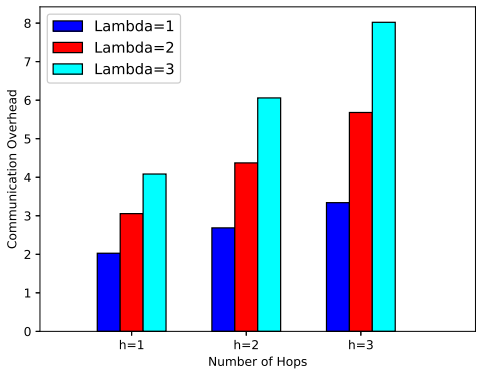
Fig. 2.3a shows the impact of the number of compromised nodes on P_{witness}, the probability of all witnesses being compromised under the assumption that compromised nodes are distributed uniformly at random. As we can see, the larger the λ, the smaller P_{witness}, and vice versa. This is expected, as P_{witness} is approximately $(\frac{c}{n})^{\lambda}$. For



(a) P_{witness} vs. # of compromised nodes



(b) P_{witness} vs. # of compromised nodes



(c) # of hops vs. communication overhead

Figure 2.3: Performance of the countermeasure, where $n = 1225$.

example, when 10% of the nodes are compromised, the probability that all witnesses are compromised is 0.01 if $\lambda = 2$. A compromised sensor node can successfully launch enumeration attack on one selected synopsis if it can find a reading that leads to the minimal synopsis and all λ witnesses are also compromised.

The attacker may choose to compromise one selected sensor node and then the nodes within its h-hop neighborhood. Fig. 2.3b shows P_{witness} varying with the number of compromised nodes under different h. As we can see, the more compromised nodes, the smaller h, the higher P_{witness}, and vice versa. This is expected, as the λ witnesses are chosen uniform at random from all the nodes within h-hop neighborhood. When the number of compromised nodes exceeds the number of nodes in the h hop neighborhood,

P_{witness} becomes one. In this case, the success probability is reduced to the probability that the compromised node can successfully find a reading that leads to the minimal synopsis among all sensor nodes.

Fig. 2.3c shows the impact of λ, the number of witnesses that store the commitment, on the extra communication overhead incurred by the proposed countermeasure. It is not surprising to see that the larger the λ, the more message transmissions incurred by the proposed countermeasure. In addition, the number of message transmissions also increases as h increases for the same λ. The reason is that the larger h, the larger the average distance between a node and its witnesses. Overall, our countermeasure incurs a small number of extra message transmissions. For example, when $h = 3$ and $\lambda = 3$, the proposed countermeasure incurs approximately 8 extra message transmissions over VMAT.

2.7 Summary

In this chapter, we have introduced a novel enumeration attack against VMAT to highlight the security vulnerability of sensor node reporting arbitrary readings. In comparison with the naive attack, the enumeration attack allows a single compromised sensor node to cause significantly higher estimation error at the base station without being detected. We also introduce an effective countermeasure against the enumeration attack. Theoretical analysis and simulation studies have confirmed the severe impact of the enumeration attack and the effectiveness of the proposed countermeasure.

Chapter 3

SECURE QUANTILE AGGREGATION SUMMARIES

3.1 Introduction

Wireless sensor networks are widely expected to play a key role in emerging Internet of Things (IoT)-based smart cities in which a large number of sensor nodes continuously sense the physical environment and generate data to assist intelligent decision making [42, 43]. Since sensor nodes are typically resource-constrained with limited computation capability, memory, and energy, blindly forwarding all the sensed data to a base station may quickly deplete sensor nodes' limited energy. Data aggregation has been widely considered as a key functionality [44] for reducing data redundancy, improving energy efficiency, and prolonging the lifetime of wireless sensor networks, in which sensed data are aggregated enroute by intermediate sensor nodes, which allow a base station to acquire important statistics about the sensed data.

Secure data aggregation is necessary to safeguard the aggregation process. Resource constrained sensor nodes are subject to a wide range of attacks. Once compromised, a sensor node may carry out a wide range of attacks under the attacker's instruction. For example, it may change the subaggregate that can significantly deviate the final aggregation result at the base station. As a result, secure data aggregation has been investigated extensively over the past to allow the base station to acquire important statistics about the sensed data [2, 7, 6, 45, 21, 3, 4, 5, 8, 10, 46]. Unfortunately, all existing solutions target simple statistics such as Sum, Count, Min/Max, and Median.

Quantile summary aggregation allows a base station to learn a more accurate distribution of the sensed data. Specifically, a quantile summary allows the base station

to extract the ϕ-quantile for any $0 < \phi < 1$ of all the sensor readings in the network and thus can provide a more accurate characterization of the data distribution. Given a set of n distinct data values with a total order, the ϕ-quantile is the value x with rank $r(x) = \lfloor \phi n \rfloor$ in the set, where $r(x)$ is the number of values in the set smaller than x. Since a quantile summary that can provide the exact quantiles must contain the all n values in the worst case, an ϵ-approximate ϕ-quantile is a value with rank between $(\phi - \epsilon)n$ and $(\phi + \epsilon)n$. While several quantile summary aggregation protocols [23, 24, 1, 25] have been proposed in the past, none of them were designed to withstand potential attacks. How to realize secure quantile summary aggregation in wireless sensor networks thus remains an open challenge.

In this chapter, we fill this void by introducing SecQSA, a novel secure quantile summary aggregation protocol for wireless sensor networks. Our proposed protocol is based on the quantile summary aggregation protocol proposed by Huang *et al.* [1], because it can guarantee a constant individual node communication cost independent of network size even for those close to the base station with many decedents. We observe that the key for securing quantile summary aggregation is to ensure the integrity of the merging operation that merges multiple local quantile summaries into one. Based on this observation, we design a secure merging procedure using efficient cryptographic primitives. Our contributions in this chapter can be summarized as follows:

- To the best of our knowledge, we are the first to study secure quantile summary aggregation in wireless sensor networks.
- We identify a range of possible attacks on quantile summary aggregation.
- We introduce a novel secure quantile summary aggregation protocol based on efficient cryptographic primitives to ensure the integrity of the final quantile summary received by the base station.
- We confirm the efficacy and efficiency of the proposed protocol via simulation studies.

The rest of this chapter is structured as follows. Section 3.2 discusses the related work. Section 3.3 introduces the network and adversary models. Section 3.4 evaluates the impact of different attacks on quantile summary aggregation. Section 3.5 introduces the design of SecQSA. Section 3.6 reports the simulation results. Section 3.7 finally concludes this chapter.

3.2 Related Work

Secure data aggregation in wireless sensor networks have been studied extensively in the past. Most of the existing solutions target simple aggregation functions such as Sum, Count, Average, and Min/Max. The resilience of different aggregation functions under a single aggregator model was analyzed in [21]. Przydatek *et al.* [6] introduced a secure aggregation scheme that can support Median, Min/Max, and Average aggregation. In [2], Chan *et al.* presented a secure hierarchical additive aggregation scheme, which was subsequently improved by Frikken *et al.* with reduced communication cost [4]. A commitment-based hop-by-hop aggregation scheme was introduced in [3] which allows the base station to verify abnormal aggregate via hypothesis testing. A secure hierarchical data aggregation scheme based on synopsis diffusion was proposed in [45, 10], which can support additive aggregation functions such as Count and Sum against falsified sub-aggregate attacks. In [5], Papadopoulos *et al.* introduced a secure aggregation scheme for exact Sum aggregation. Chen presented a scheme [9] that realizes secure approximate Sum aggregation via secure Min aggregation, which was later shown to be vulnerable to a special enumeration attack [46].

There are very limited efforts in developing secure aggregation schemes to support Median and Percentile aggregation. The techniques presented in [6, 2] can be used for verifying the correctness of an alleged ϕ-percentile via secure Count aggregation by counting the number of readings that are smaller than the alleged ϕ-percentile. Roy *et al.* [7] extended the secure Count aggregation scheme [2] to realize secure Median aggregation by recursively constructing an increasingly accurate histogram. However, these solutions require the base station to know the percentile of interest, i.e., ϕ, in

advance and incurs a communication cost proportional to the number of percentile queries.

Quantile summary [22] aggregation in wireless sensor networks has been studied. In [23], a quantile digest summary structure was introduced to realize quantile aggregation. Greenwald *et al.* [24] introduced a distributed algorithm to compute an ϵ-approximate quantile summary of sensor data, which was later improved by Huang *et al.* [1] to reduce the maximum per node communication cost. More recently, several efficient gossip algorithms were introduced in [25] to compute exact and approximate quantiles in a fully distributed fashion. Unfortunately, none of the above quantile aggregation schemes have any security provisions. None of these works consider possible attacks, and they cannot be applied to our problem.

3.3 Network and Adversary Models

In this section, we introduce our system and adversary models.

3.3.1 Network Model

We consider a multi-hop wireless sensor network consisting of a base station and s sensor nodes. Every sensor node senses the environment and periodically generates readings at fixed frequency. We assume that every sensor node i has a set of n readings denoted by D_i and every reading is in the range $R = \{1, \ldots, v_{\max}\}$ it should be float numbers. It follows that the total number of readings in the network is sn. As in [1], we assume that all the readings in the sensor network are distinct. While this assumption may seem restrictive, it can be easily accommodated by imposing a total order among the readings by taking node ID and the time at which a reading is generated to break the tie.

The base station aims to obtain a quantile summary of all the readings generated in the network over a certain period. A quantitle summary is a subset of readings along with their (estimated) global ranks which can support *value-to-rank* queries. Specifically, for any value $v \in R$, the value-to-rank query returns an estimated global

rank $\hat{r}(v)$. The ϕ-quantile of all the readings $\bigcup_{i=1}^{s} D_i$ is then the value x with rank $r(x) = \lfloor \phi sn \rfloor$ for any $0 < \phi < 1$.

We assume that the aggregation is performed over an aggregation tree, which is the directed tree rooted at the base station formed by the unique path from every sensor node to the base station. During network initialization, the base station learns the topologies of the network as well as the aggregation tree. We also assume that each sensor node i shares a secret key K_i with the base station. We also assume that any two nodes i and j can establish a shared key $K_{i,j}$ using existing techniques such as [47, 48].

3.3.2 Adversary Model

The attacker aims to mislead the base station into accepting a modified distribution of an aggregated summary without being detected in order to significantly shift any quantile query result from its original position. We assume that the base station is equipped with adequate computation and energy resources and is safeguarded from any malicious attacks. In contrast, sensor nodes are constrained in computation and communication resources which make them susceptible to compromising. Once a sensor node is compromised, all the information stored at the sensor node such as cryptographic keys is revealed to the attacker. The attacker can then instruct compromised sensor nodes to carry out a wide range of attacks.

Since the aggregated summary consists of a subset sampled readings and their ranks, we consider the following two attacks in this chapter.

- A compromised node may forge its own readings, their ranks, or both.
- A compromised node may deviate from protocol operations, which includes dropping other nodes' readings, replacing other nodes' readings with its own, modifying other nodes' readings or their ranks.

In addition, we do not consider denial-of-service attacks, in which a compromised sensor node persistently disrupts the aggregation process.

3.4 Attacks on Quantile Summary Aggregation

In this section, we first briefly review the sampling based quantile summary protocol proposed by Huang *et. al.* [1], which serves as the basis for SecQSA. We then evaluate the impact of a range of attacks on the Huang's protocol via simulation studies.

3.4.1 Review of Huang's Protocol [1]

Huang's protocol [1] is designed based on random sampling. Let $G_1, \ldots, G_k$ be a family of sets of data values, where $G_i \bigcap G_j = \emptyset$ for all $1 \leq i < j \leq k$. If we independently sample each value in G_i with probability q to obtain a subset $S_i \subseteq G_i$ for all $i = 1, \ldots, k$. Denote by $r(v, G_i)$ its local rank within the set G_i for each sampled value $v \in S_i$. Given any value x, we can estimate its local rank $\hat{r}(x, G_i)$ within G_i for all $1 \leq i \leq k$. Let $p(x|S_i)$ be the predecessor of value x in S_i. It has been shown that

$$\hat{r}(x, G_i) = \begin{cases} r(p(x|S_i), G_i) + 1/p, & \text{if } p(x|S_i) \text{ exists;} \\ 0 & \text{otherwise,} \end{cases} \tag{3.1}$$

is an unbiased estimator of $r(x, G_i)$. The global rank of value x within $G = \bigcup_{i=1}^{k} G_i$ can then be estimated as

$$\hat{r}(x) = \sum_{i=1}^{k} \hat{r}(x, G_i) \ .$$

Under Huang's protocol [1], every node i first samples each reading of its own independently to generate a local quantile summary. All the nodes then participate in quantile summary aggregations in which local quantile summaries are forwarded and merged with others into one along the way before reaching the base station. A key advantage of Huang's scheme [1] over prior solutions [24, 23] is that it can guarantee an individual node communication cost of $O(1/\epsilon)$ even for those nodes close to the base station and have many decedents by carefully designed merging conditions. We refer readers to [1] for details of Huang's scheme.

3.4.2 Impact of Attacks

We now evaluate the impact of several attacks on Huang's protocol [1], which will guide the design of SecQSA.

Several possible attacks can be launched by a compromised sensor node. First, a compromised sensor node can arbitrarily forge its own readings and their local ranks, which is fundamentally difficult to detect without any special assumption. Moreover, since a quantile summary consists of a subset of sample values with their local ranks, a compromised sensor node can also modify the readings of its decedent nodes and corresponding ranks. In addition, Huang's protocol [1] requires that every reading is sampled independently during merging operations, but a compromised node may not follow by discarding all the readings from one or more of its decedent nodes. Due to symmetry, we only consider the case in which the attacker intends to inflate the estimated rank of any value and consider the following three attacks.

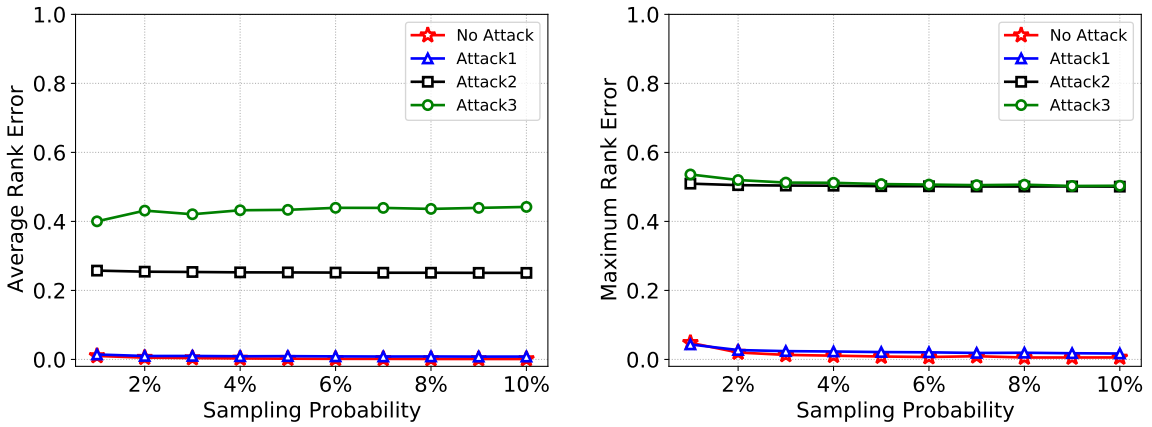
- *Attack 1*: Modify its own sampled values to the minimum and their ranks to the maximum.
- *Attack 2*: Modify children nodes' sampled values to the minimum and their ranks to the maximum.
- *Attack 3*: Modify its own sampled values to the minimum and their ranks to the maximum and drop all the children nodes' value from the quantile summary.

We use the following two metrics to evaluate the impact of the above three attacks on the accuracy of the final quantile summary at the base station. Let $r(v)$ and $\hat{r}(v)$ be the true rank and estimated rank of a value v, respectively, for all $v \in \{1, \ldots, v_{\max}\}$. The normalized average rank error (ARE) and maximum rank error (MRE) are defined as

$$\mathsf{ARE} = \frac{\sum_{v=1}^{v_{\max}} |\hat{r}(v) - r(v)|}{v_{\max}^2} , \tag{3.2}$$

and

$$\mathsf{MRE} = \frac{\max_{v=\{1,\ldots,v_{\max}\}}(|\hat{r}(v) - r(v)|)}{v_{\max}} . \tag{3.3}$$



(a) Average rank error

(b) Maximum rank error

Figure 3.1: Comparison of ARE and MRE under different attacks where $K = 62$, $c = 2$, $l = 6$ and $n = 1000$

We simulate a wireless sensor network consisting of $s = 62$ sensor nodes which form an aggregation tree of height 6 where each sensor node has two children nodes. We assume that each node has $n = 1000$ readings. Every point in the following figures is the average of 100 runs each with a distinct random seed for the sampling process.

Figs. 3.1a and 3.1b compare the ARE and MRE under the three types of attack as well as in the absence of attack with the sampling probability varying from 0.01 to 0.l. As we can see, both ARE and MRE decreases as the sampling probability increases in the absence of attack. This is expected as the higher the sampling probability, the more readings are included in the final quantile summary received by the base station, the more accurate the value-to-rank query results, the lower ARE and MRE, and vice versa. In addition, the ARE and MRE under Attack 1 are very close to those under no attack. The reason is that a single compromised node forging its own readings and local ranks has very limited impact on the accuracy of final quantile summary. In contrast, the ARE and MRE under Attack 2 and Attack 3 are significantly higher than those under Attack 1. In particular, we can see from Fig. 3.1a that the AREs under Attack 2 and Attack 3 are 0.24 and 0.42, respectively. Similarly, the MREs under Attack 2 and Attack 3 are both around 0.5. These results clearly demonstrate the severe impact of Attacks 2 and 3 on the final quantile summary.

3.5 SecQSA: Secure Quantile Summary Aggregation

In this section, we first give an overview of SecQSA and then detail its design.

3.5.1 Overview

We find that the key to secure quantile summary aggregation is to ensure the integrity of the readings and their ranks during merging operations. Specifically, SecQSA is designed to achieve the following goals.

1. *Integrity of sample readings*: every reading in the final quantile summary must be generated by a sensor node and has not been altered during the aggregation process.
2. *Integrity of local ranks*: as readings being aggregated into different quantile summaries through the process, their local ranks within quantile summaries must be correctly computed according to [1].
3. *Compliance of uniform sampling*: when multiple quantile summaries are merged, every reading should be sampled independently according to [1].

We do not intend to defend against a compromised node forging its own readings and their local ranks, which has very limited impact on the aggregation results as shown in Section 3.4.2.

SecQSA is designed to meet the above goals using efficient cryptographic primitives. Under SecQSA, sensor nodes send, receive, and merge local quantile summaries in a secure fashion. Specifically, a quantile summary Q is associated with a ground set G and represented by

$$Q = \langle ID, O, q \rangle,$$

where ID is the node that generates the quantile summary, O is a set of *sample objects*, and q is the sampling probability. Every sample object $o \in O$ corresponds to one reading drawn from the ground set G and has the form

$$o = \langle v, \sigma_{\mathsf{init}}, \sigma_{\mathsf{current}} \rangle.$$

where v is the reading, and σ_{init} and $\sigma_{\mathsf{current}}$ carry the necessary verification information about v in the quantile summary. More specifically, the first component σ_{init} carries the initial rank of v and has the form

$$\sigma_{\mathsf{init}} = \langle ID_i, r(v, D_i), H_{K_i}(v||r(v, D_i))\rangle,$$

where ID_i is the ID of the node that generates reading v, $r(v, D_i)$ is the initial local rank of v in node i's set D_i, K_i is the secret key node i shared with the base station, and $H_*(\cdot)$ denotes a message authentication code keyed with the subscript. The second component $\sigma_{\mathsf{current}}$ has the form

$$\sigma_{\mathsf{current}} = \langle ID_j, r(v, G)\rangle,$$

where ID_j is the ID of the node which merges value v into the current quantile summary Q, and $r(v, G)$ is the local rank of v in the current ground set G.

As a reading v moves through the aggregation process, the first component σ_{init} remains unchanged and will allow the base station to verify the integrity of the reading and compliance of random sampling of any intermediate node. In contrast, the second component $\sigma_{\mathsf{current}}$ will be updated whenever reading v is merged into a new quantile summary.

In what follows, we detail how quantitle summaries are generated by individual sensor nodes and merged through the aggregation process.

3.5.2 Initialization

To initiate a quantile summary aggregation process, the base station broadcasts a command with a random seed d using a proper broadcast authentication protocol such as μ-Telsa [49].

On receiving the command, each sensor node i first generates a local quantile summary Q_i with respect to its set of readings D_i. Let $H(\cdot)$ be a cryptographic hash function that maps any input to an integer in the range $\{0, \ldots, \lambda - 1\}$. Node i samples every reading $v \in D_i$ with probability q_{init}, where q_{init} is a system parameter

that determines the accuracy of the quantile summary and communication overhead. Specifically, every reading v is selected to be included in the local quantile summary Q_i if

$$H(ID_i||r(v, D_i)||d) \leq q_{\text{init}}\lambda \ . \tag{3.4}$$

It is easy to see that each reading is sampled independently with probability q_{init}. We subsequently denote by $S_i \subseteq D_i$ the subset of readings included in Q_i.

For each selected sample reading $v \in S_i$, node i constructs a sample object as $o = \langle v, \sigma_{\mathsf{init}}, \sigma_{\mathsf{current}} \rangle$, where

$$\sigma_{\mathsf{init}} = \sigma_{\mathsf{current}} = \langle ID_i, r(v, D_i), H_{K_i}(v||r(v, D_i)) \rangle \ .$$

3.5.3 Secure Quantile Summary Aggregation

All the nodes then participate in the quantile summary aggregation based on the aggregation tree. Specifically, every leaf node i of the aggregation tree sends its local quantitle summary Q_i to its parent node, say j, as

$$Q_i = \langle ID_i, O_i, q_{\text{init}}, H_{K_{i,j}}(\mathsf{info}) \rangle \ ,$$

where $O_i = \{o | v \in S_i\}$ is the set of sample objects and $\mathsf{info} = ID_i||O_i||q_{\text{init}}$ is the concatenation of all the prior information.

On receiving a local quantile summary Q_i from one of its children nodes, node j first verifies its integrity by checking $H_{K_{i,j}}(\mathsf{info})$ using the shared key $K_{i,j}$. If succeed, node j checks if local quantile summary Q_i exhibits any inconsistency. Specifically, node j checks if the reading in every sample object is in the range R. Without loss of generality, suppose that $O_i = \langle o_1, \ldots, o_x \rangle$, where $o_x = \langle v_i, \sigma_{\mathsf{init}}, \sigma_{\mathsf{current}} \rangle$ and $v_1 < \cdots < v_x$. Node j checks if $r(v_1, D_i) < \cdots < r(v_x, D_i)$. If so, node j considers quantile summary Q_i valid.

Node j then processes Q_i in one of the two possible ways. In the first case, node j directly forwards Q_i to its parent node, say k, by sending

$$j \rightarrow k : \langle ID_i, O_i, q, H_{K_{j,k}}(\mathsf{info}) \rangle \ ,$$

which allows node k to verify its integrity. In the second case, node j merges Q_i with one or more other local quantile summaries to produce a single quantile summary if the conditions specified in [1] are met. In what follows, we use an example to illustrate how multiple local quantile summaries are merged at an intermediate node.

Suppose that node j intends to merge l local quantitle summaries $Q_1, \ldots, Q_l$ into one local quantile summary Q. Each local quantitle summary

$$Q_x = \langle ID_x, O_x, q_x \rangle \ ,$$

is sampled from a ground set G_x with sampling probability q_x independently for all $x = 1, \ldots, l$. The resulting quantile summary $Q_j = \langle ID_j, O_j, q \rangle$ corresponds to the ground set $G = \bigcup_{x=1}^{l} G_x$ where every reading in G is sampled independently with probability q.

The merging operation involves four steps. First, node j resamples every reading in $Q_1, \ldots, Q_l$ to obtain the set of readings to be included in Q. Specifically, for each quantile summary Q_x, $1 \leq x \leq l$, node j samples every sample unit $o \in O_x$ independently with probability q/q_x. In particular, each sample object $o \in O_x$ is selected if

$$H(ID_j || r(v, G_x) || d) \leq \frac{q\lambda}{q_x} \ . \tag{3.5}$$

It follows that each reading v in the ground set G_x is selected to be in Q with probability

$$\begin{aligned} \mathsf{Pr}(v \in Q) &= \mathsf{Pr}(o \in Q | v \in Q_x) \mathsf{Pr}(v \in Q_x) \\ &= \frac{q}{q_x} \cdot q_x \\ &= q \ . \end{aligned}$$

Second, node j computes the local rank of every reading in the quantile summary Q. Let $O'_x \subseteq O_x$ be the subset of sample objects in Q_x selected to be included in Q for all $1 \leq x \leq l$. Consider a sample unit $o \in O'_j$ as an example where $o = \langle v, \sigma_{\mathsf{init}}, \sigma_{\mathsf{current}} \rangle$ and $\sigma_{\mathsf{current}} = \langle ID_z, r(v, G_x), H_{K_z}(v || r(v, G_x)) \rangle$. It follows that v is ranked $r(v, G_x)$

within the ground set G_x. Its local rank within the new ground set $G = \bigcup_{y=1}^{l} G_x$ can then be estimated as

$$r(v, G) = r(v, G_x) + \sum_{y=1, y \neq x}^{l} r(v, G_y) \ ,$$

where

$$r(v, G_y) = \begin{cases} r(p(v|O_y), G_y) + 1/q_y, & \text{if } p(v|O_y) \text{ exists;} \\ 0 & \text{otherwise,} \end{cases}$$

and $p(v|O_y)$ is the predecessor of value v in O_y. It has been shown that $r(v, G)$ is an unbiased estimator of v's local rank within G [1].

Third, node j updates each sample object in Q. Specifically, for each $o = \langle v, \sigma_{\mathsf{init}}, \sigma_{\mathsf{current}} \rangle$ selected, node j updates $\sigma_{\mathsf{current}}$ to

$$\sigma_{\mathsf{current}} = \langle ID_j, r(v, G) \rangle \ .$$

Next, node j, its children nodes, and its parent node k execute a protocol whereby node j's children nodes verify and endorse the new local rank of each sample object in G. Among the l local quantile summaries $Q_1, \ldots, Q_l$, there is at most one local quantile summary generated by node j itself. Without loss of generality, suppose that local quantile summary Q_j is generated by node j itself and that each quantile summary Q_y is received from children node y for all $y = 1, \ldots, l$ and $y \neq j$.

Node j first broadcasts the quantitle summary as

$$j \rightarrow * : \langle Q, H_{K_{j,k}}(Q) \rangle \ ,$$

where $Q = \langle ID_j, O, q \rangle$ and O is the set of sample objects. This message will be received by both node j's parent node k and all the children nodes as they are all in node j's transmission range. On receiving the message, node k verifies its integrity using the shared key $K_{j,k}$.

Node j then seeks its children nodes' endorsement for the new quantile summary Q. Since every children node y knows Q_y it sent earlier and also overheard the quantile summary Q, it knows the subset of sample object $O'_y \subseteq O_y$ being included in Q. Each

node y first verifies whether node j faithfully perform random sampling for O_y according to Eq. (3.5). Moreover, node y also checks whether node j correctly computes the new local rank $r(v, G)$ of each sample object $o \in O$. Specifically, for each sample object $o \in O$ where $o = \langle v, \sigma_{\mathsf{init}}, \sigma_{\mathsf{current}} \rangle$, node j broadcasts the following message

$$j \rightarrow * : v, r(v, G_1), q_1, r(v, G_2), q_2, \ldots, r(v, G_l), q_l \ ,$$

Without loss of generality, consider sample object $o \in O'_y \subseteq O_y$ and Q_y was sent by child node y. Node y first verifies whether

$$r(v, G) = r(v, G_y) + \sum_{z=1, z \neq y}^{l} r(v, G_z) \ .$$

If so, node y sends its endorsement to node j as

$$x \rightarrow j : H_{K_{y,k}}(Q) \ ,$$

where $K_{y,k}$ is the shared key between node y and j's parent node k. Similarly, every other children node z ($z = 1, \ldots, l$, $z \neq y$, and $z \neq j$) which sent Q_z finds $p(v|O_z)$, i.e., the predecessor of v in O_z, and verifies whether

$$r(v, G_z) = r(p(v|O_z), G_z) + 1/q_z \ .$$

If so, node z sends its endorsement to node j as

$$y \rightarrow j : H_{K_{z,k}}(Q) \ ,$$

On receiving the endorsement from each of its children, node j sends an aggregated endorsement of Q to its parent node k

$$j \rightarrow k : \bigoplus_{y=1, y \neq j}^{l} H_{K_{y,k}}(Q) \ .$$

Since the parent node j has previously verified the integrity of Q using $H_{K_{j,k}}(Q)$, it further verifies the aggregated endorsement $\bigoplus_{y=1, y \neq j}^{l} H_{K_{y,k}}(Q)$ using the keys shared with each children node y ($y = 1, \ldots, l$). If all the verifications succeed, node k accepts Q as a valid quantile summary.

3.5.4 Final Verification at the Base Station

At the end of the aggregation process, the base station receives one or multiple quantile summaries from its children nodes. For every quantile summary it receives, the base station verifies the quantitile summary in the following steps.

First, for each sample object $o = \langle v, \sigma_{\mathsf{init}}, \sigma_{\mathsf{current}} \rangle$ where $\sigma_{\mathsf{init}} = \langle ID_i, r(v, D_i), H_{K_i}(v||r(v, D_i)) \rangle$, the base station first verifies v's integrity by recomputing $H_{K_i}(v||r(v, D_i))$ using the shared key K_i.

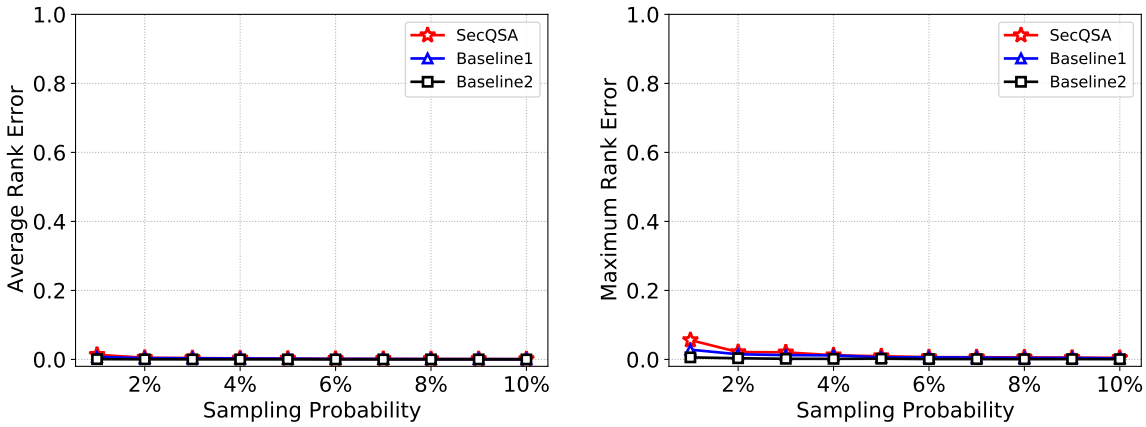
Second, the base station verifies if every node that performed merging operations have faithfully followed random sampling. Since the base station knows the number of readings each node has and the aggregation tree structure and the random sampling performed at each node is based on each reading's initial rank, the ID of the node that performs sampling, and the seed d, the base station can emulate the entire aggregation process to predict (1) the subset of readings sampled in each initial local quantile summary, (2) the number of quantile summaries received at each intermediate node and their corresponding sizes, (3) which nodes should have performed merging operations, and (4) the subset of readings that should have been selected in each merged quantile summary. Specifically, for each node i and every possible local rank $1, \ldots, n$, the base station verifies if (1) for every initial rank that is supposed to survive the entire aggregation process, the corresponding reading is indeed included in the final quantile summary, and (2) if there is any reading in the final quantile summary received should have been dropped by any intermediate node.

3.6 Simulation Results

In this section, we evaluate the performance of SecQSA via simulation.

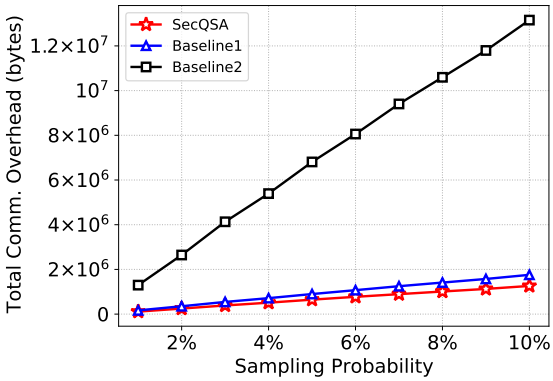
3.6.1 Simulation Setting

We again consider a wireless sensor network consisting of $s = 62$ sensor nodes which form an aggregation tree of height 6 where each sensor node has two children

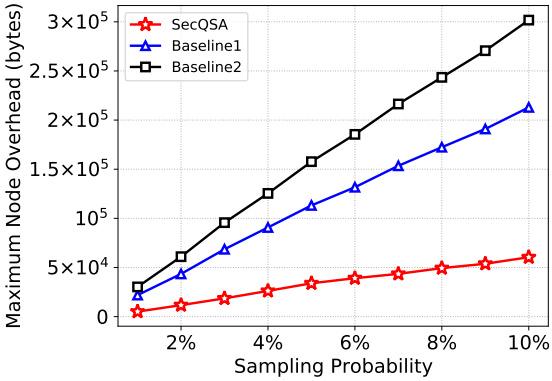


(a) Average rank error

(b) Max. rank error



(c) Total comm. overhead



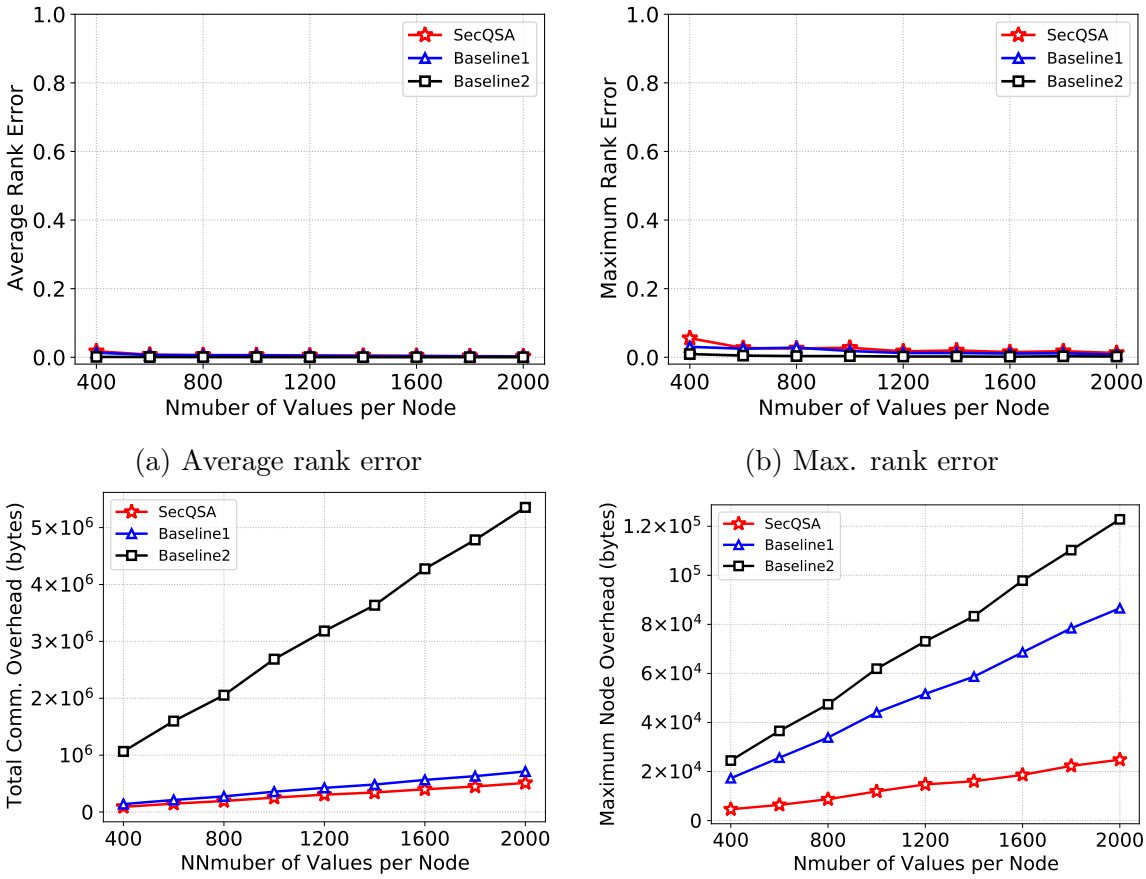
(d) Max. per node overhead

Figure 3.2: Comparison of SecQSA and the baselines with sampling probability varying from 0.01 to 0.1.

nodes. Table 3.1 summarizes our default settings unless mentioned otherwise. Every point in the following graphs is the average of 100 runs, each with a distinct random seed for the sampling process. We adopt the SHA-256 for the message authentication code which results in a 32 bytes code. Also, we assume that each reading is of 16 bits, and each local rank is of 32 bits.

Since there is no prior solution for secure quantile summary aggregation, we compare the proposed protocol with two baseline schemes.

- *Baseline 1*: every node independently samples its readings with probability q and then submits the sampled readings along with their associated ranks and a



(a) Average rank error

(b) Max. rank error

(c) Total Comm. Overhead

(d) Max. per node overhead

Figure 3.3: Comparison of SecQSA and the baselines with the number of values per node varying from 400 to 2000.

MAC to the base station. The base station verifies the integrity of each reading and answers value-to-rank queries according to Eq. (3.1).

- *Baseline 2*: every node independently samples its readings with probability q and then submits the sampled readings along with a MAC to the base station with no ranking information. On receiving all the sample readings, the base station broadcasts all the readings to all the sensor nodes. Finally, all the nodes participate in multiple parallel secure SUM aggregations according to [5] to allow the base station to obtain the global rank of each reading, whereby to answer value-to-rank queries according to Eq. (3.1).

Besides the ARE and MRE, we also use *total communication overhead* and *maximum*

Table 3.1: Default Simulation Settings

Para.	Val.	Description.
p	0.02	The sampling probability
n	1000	The number of values per sensor
l	6	The maximum number of network tree levels
c	2	The maximum number of children per sensor node

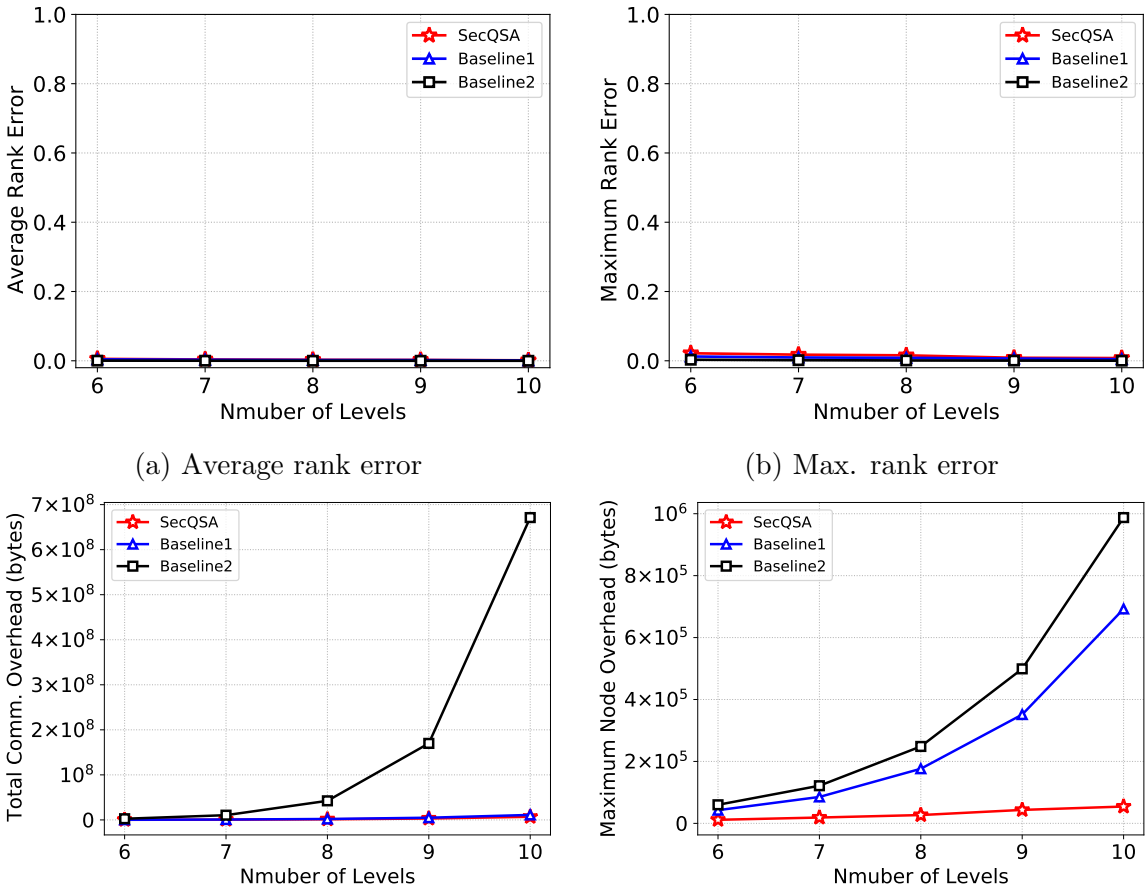
per node communication overhead to evaluate the performance of proposed scheme and the two baseline schemes.

3.6.2 Simulation Results

Figs. 3.2a to 3.2d compare the ARE, MRE, total communication overhead, and maximum per node communication overhead of SecQSA and the two baseline solutions, respectively, with sampling probability varying from 0.01 to 0.1. We can see from Fig. 3.2a and 3.2b that both ARE and MRE decrease as the sampling probability q increases under all three schemes. This is expected as the more readings we sample, the more accurate the rank estimation, and vice versa. Also, we can see from Fig.3.2a that the AREs of all three schemes are almost the same for SecQSA and the two baselines where all of them are too close to the zero. This is because all of the methods are sampling the same number of values and only differ in the way of estimating global ranks which will not produce a big difference in accuracy especially with large number of values. The same is for the MRE shown in Fig. 3.2b, where it also shows a big similarity in the MRE produced by each of the three schemes for the same reason. Accordingly, the accuracy of SecQSA is quite the same as the two baselines. On the other hand, Fig.3.2c shows the total communication overhead under SecQSA and the other two Baselines. In general, it shows that the total communication overhead increases as the sampling probability increases. The reason is that, the more values are sampled, the more information need to be sent and accordingly the more total communication overhead. Moreover, we can see that Baseline 2 has the largest communication overhead among the three because each sensor node needs to send the information of the total number of sampled values in the network compared to Baseline 1 and SecQSA where

each node needs only to send the sampled values of its subtree. On the other hand, SecQSA produces almost the same total communication overhead as Baseline 1 with only a slight decrease due to the merging process in SecQSA which is anticipated to decrease more as we have more merging operations in the aggregation process. Fig. 3.2d plots the maximum per node communication overhead for each of the three schemes. It shows that the maximum per node overhead increases in the three schemes with the increase of the sampling probability, which is anticipated. Also, it shows that Baseline 2 has the largest overhead amongst the two others for the same reason mentioned above and then Baseline 1 comes after it in order while SecQSA beats both Baseline 1 and Baseline 2 with significant margins.

Figs. 3.3a to 3.3d compare the ARE, MRE, total communication overhead, and maximum per node communication overhead under SecQSA, Baseline 1 and Baseline 2 with the number of readings per node varying from 400 to 2000. Again we can see from Figs. 3.3a and 3.3b that both ARE and MRE decrease as the number of values per node increases under all three schemes. This is because, the more values per node, the more sampled values, the more accurate the rank estimation. Also, we can see in Fig. 3.3a that the average rank error is almost the same for SecQSA and each of the baselines for the same reason mentioned in previous paragraph. Fig. 3.3b shows the MREs for the three schemes with the same observation, where the MREs are quite the same and close to zero due to the same reason as in Fig. 3.2b. For the communication overhead in Figs. 3.3c and 3.3d, both the total communication overhead and maximum per node overhead produced by the three schemes show an increase as the number of values per node increases. The reason is that, the more number of values per node, the more sampled values, the more information need to be communicated and accordingly the more communication overhead we get either in total or per node. Also, Fig. 3.3c shows that Baseline 2 has the largest overhead amongst the two others as expected in the previous paragraph. On the other hand, SecQSA shows a slight decline in the total communication overhead compared to Baseline 1 due to the merging process. In contrast, Fig. 3.3d shows the maximum per node communication overhead for each of



(a) Average rank error

(b) Max. rank error

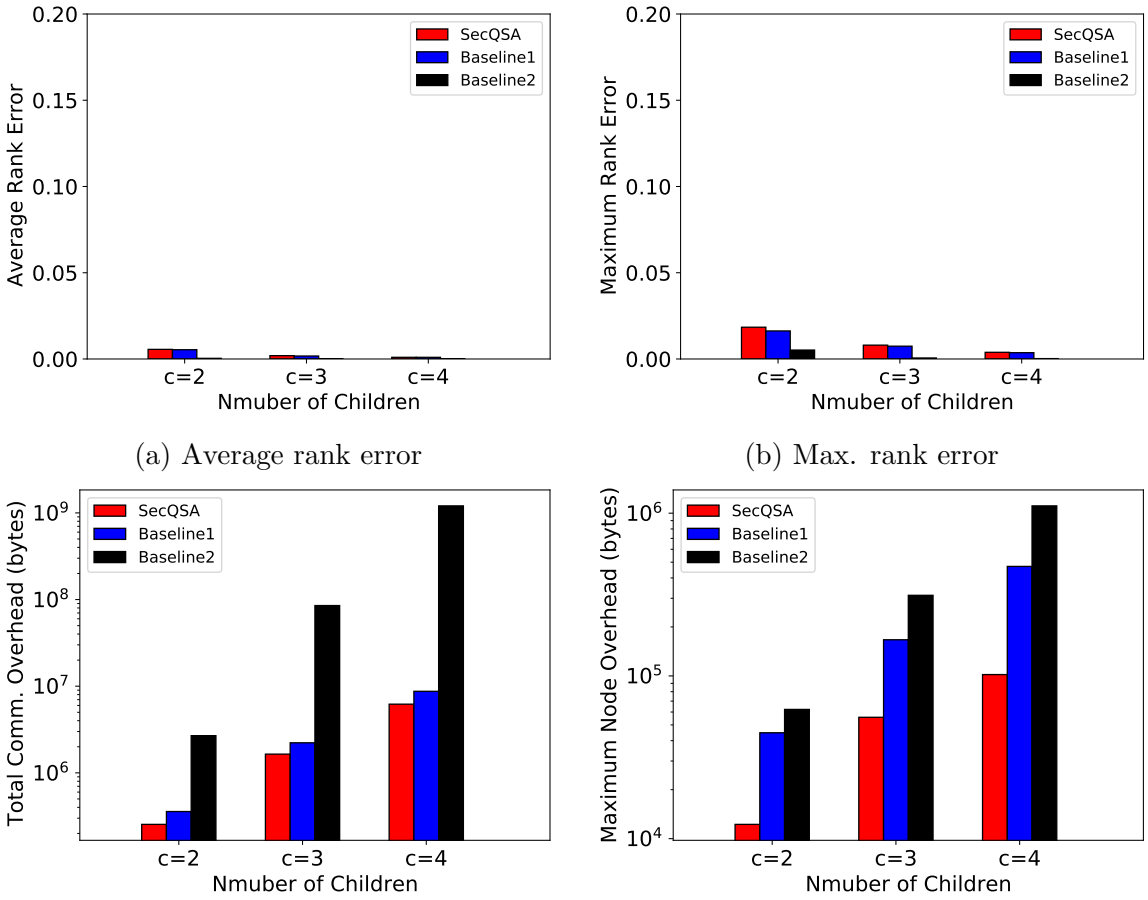
(c) Total Comm. Overhead

(d) Max. per node overhead

Figure 3.4: Comparison of SecQSA and the baselines with the height of the aggregation tree varying from 6 to 10.

the schemes where again as expected before that Baseline 2 comes in the beginning as the largest maximum per node overhead. Second, comes Baseline 1 and then SecQSA as the least per node overhead with a big gap due to the merging operations.

Figs. 3.4a to 3.4d compare the error and communication overhead produced by SecQSA, Baseline 1 and Baseline 2 considering different number of levels in the network tree. As we can see in Figs. 3.4a and 3.4b that both ARE and MRE decrease as the number of levels increases in the network tree for the three schemes. This is expected as the more number of levels, the larger is the total number of values in the network, the more the sampled values which leads to a more accurate result. Also, we can see in Fig. 3.4a that the average rank error for SecQSA and each of the baselines are quite



(a) Average rank error

(b) Max. rank error

(c) Total Comm. overhead

(d) Max. per node overhead

Figure 3.5: Comparison of SecQSA and the baselines with the number of children per node varying from 2 to 4.

the same and close to zero as explained before. Moreover, the MREs produced by the three schemes in Fig. 3.4b are pretty close and similar as expected before. Figs. 3.4c and 3.4d plot the total communication overhead and maximum per node overhead produced by the three schemes where both types of overhead increase as the number of levels increases. The reason is that, the more levels, the more sampled values, the more information to be communicated. Moreover, Figs. 3.4c and 3.4d show that Baseline 2 has the largest total communication overhead and maximum per node communication overhead among the three because it needs to communicate more information. On the other hand, Baseline 1 comes as the second largest communication overhead either in

total with a slight increase over SecQSA or per node. Finally, SecQSA comes at last as the least for either types of communication overhead due to the merging process which shows a clear difference in Fig. 3.4d.

Figs. 3.5a to 3.5d show the ARE, MRE, total communication overhead, and maximum per node communication overhead produced by SecQSA, Baseline 1 and Baseline 2 with different number of children per node in the network tree. As we can see, Figs. 3.5a and 3.5b show that both ARE and MRE decrease as the number of children increases in the network tree for the three schemes. This is anticipated as the more children, the larger the total number of values in the network, the more sampled values, the more accurate the rank estimation, and vice versa. Also, we can see in Fig. 3.5a that the ARE for SecQSA and each of the baselines are almost matching and close to zero as we discussed before. Fig. 3.5b plots the maximum rank errors produced by the three schemes which appears to be also quite matching as expected. Figs. 3.5c and 3.5d show that the total communication overhead and maximum per node overhead under the three schemes increase as the number of children in the network increases. As expected, the more children, the more sampled values, the more information to be communicated. Also, Figs. 3.5c and 3.5d show that amongst Baseline 1 and SecQSA, Baseline 2 comes at first in regard to the size of total communication overhead and maximum per node overhead because it communicates more information. Then, Baseline 1 comes in the second order and then comes SecQSA because SecQSA communicates the least amount of overhead for either types of communication due to the merging process.

In summary, overall results show that SecQSA performance is better than the two baselines as it has the same accuracy as Baseline 1 and Baseline 2 while at the same time incurs much lower total and maximum per node communication overhead.

3.7 Summary

In this chapter, we have initiated the study of secure quantile summary aggregation in wireless sensor networks. After examining the impact of different attacks on

quantile summary aggregation via simulation, we introduced the design and evaluation of SecQSA, the first secure quantile summary aggregation protocol for wireless sensor networks. Built upon efficient cryptographic primitives, SecQSA can ensure the integrity of sampled readings and their ranks in the final quantile summary. Detailed simulation results have confirmed significant advantages of SecQSA over alternative solutions.

Chapter 4

LOCAL DIFFERENTIAL PRIVATE QUANTILE SUMMARY AGGREGATION

4.1 Introduction

Wireless sensor networks are prospected to overrun a major role in the emerging IoT paradigm where a large number of sensor nodes are expected to continuously sense the environment and produce large amounts of sensed data. Without a doubt, these collected data can provide us with a wealth of information about our physical world and immense benefits and enable a wide range of exciting applications. In-network data aggregation [44] has been widely recognized as a key technique for reducing energy consumption and prolonging network lifetime by allowing sensed data to be aggregated by intermediate nodes along the route to the base station to eliminate possible redundancy.

Data privacy is a key issue in many IoT applications. For example, data generated by sensor nodes in an IoT-based smart-home system may contain a variety of sensitive information about users such as appliance usage and home occupancy. Directly submitting such information to a base station would not only reveal sensitive information but also subject users to profiling. As another example, sensor nodes deployed in remote areas for wildlife monitoring may generate data that could reveal the locations of endangered species. These situations call for privacy-preserving data aggregation that can allow the base station to learn valuable statistic of the data generated in the wireless sensor network while ensuring the data privacy of individual sensor nodes.

Privacy-preserving data aggregation has received significant attentions over the past decade due to its importance. Similar to secure data aggregation, existing privacy-preserving data aggregation schemes [11, 12, 13, 14, 15, 16, 17] all target simple statistic functions such as SUM, COUNT, and MAX/MIN. In contrast, a quantile summary allows a base station to learn a more accurate distribution of the sensed data than simple statistics functions. More specifically, a quantile summary allows the base station to retrieve the ϕ-quantile for any $0 \leq \phi \leq 1$, which can provide a much better characterization of the distribution of data generated by a wireless sensor network. Unfortunately, how to realize privacy-preserving quantile summary aggregation remains an open challenge.

In this chapter, we introduce the design and evaluation of PrivQSA, a novel privacy-preserving quantile summary aggregation scheme for wireless sensor networks. Specifically, we design PrivQSA to satisfy ϵ-Local Differential Privacy (LDP), which is a model widely considered as the gold standard for data privacy and has been explored for various data analytic tasks such as frequency estimation [50, 51, 52, 53], heavy hitter discovery [54, 55, 56, 57], mean estimation [58, 59, 60, 61], and probability distribution estimation [62, 63, 64, 65]. Similar to SecQSA introduced in Chapter 3, PrivQSA is also based on the quantile summary aggregation protocol proposed by Huang *et al.* [1] due to its guarantee of low per node communication overhead. Under PrivQSA, every sensor node randomly perturbs its set of readings to ensure ϵ-LDP. All the nodes then participate in the quantile summary aggregation according to [1] which allows the base station to obtain a quantile summary of the perturbed readings. The base station then estimates a quantile summary of the original sensed data based on the perturbation mechanism followed by individual sensor nodes. Our contributions in this chapter can be summarized as follows.

- We are the first to study privacy-preserving quantile summary aggregation in wireless sensor networks.
- We introduce PrivQSA, a novel privacy-preserving quantile summary aggregation scheme that can allow the base station to learn a quantile summary of the sensed

data while ensuring local differential privacy for individual sensor nodes.

- We confirm the efficacy and efficiency of PrivQSA via both theoretical analysis and detailed simulation studies, which demonstrate significant advantages over other baseline solutions.

The rest of this chapter is structured as follows. Section 4.2 discusses the related work. Section 4.3 introduces the network model and some preliminaries. Section 4.4 introduces the design of PrivQSA. Section 4.5 provides a theoretical analysis of PrivQSA. Section 4.6 reports the simulation results. Section 4.7 finally concludes this chapter.

4.2 Related Work

We have discussed existing solutions for quantile summary aggregation in Chapter 3, and none of them were designed to provide data privacy for individual sensor nodes. In this section, we review some additional related works in privacy preserving data aggregation in WSNs and local differential privacy.

4.2.1 Privacy-Preserving Data Aggregation in WSNs

Privacy-preserving data aggregation in sensor networks has received a lot of attention over the past two decades [11, 13, 15, 66, 67, 68, 69, 70, 71]. Generally speaking, existing solutions for privacy preserving data aggregation can be classified into two categories.

The first category uses encryption techniques such as homomorphic encryption [66, 67, 68, 13, 15, 72], secure multiparty computation [69], and modulo addition-based encryption [73]. Castelluccia *et al.* [66] proposed to use homomorphic encryption ciphers to allow efficient aggregation of encrypted data without requiring decryption at intermediate nodes. This scheme was later extended by the same authors to improve computational and communication efficiency by requiring only a small number of single-precision additions [67]. Shi *et al.* [13] also adopted the similar technique in [66] as a tool while applying data slicing to preserve user privacy. Inspired by [66], Westhoff *et al.* [68] showed how to construct relevant aggregation functions based on

an additively homomorphic structure of a private homomorphism. Li *et al.* [15] introduced an efficient secure aggregation protocol for mobile sensing systems in which each mobile user encrypts its data using an additively homomorphic cipher to keep the data private from the aggregator and other users. Zhang *et al.* [72] proposed a ring-based privacy preserving aggregation scheme that uses homomorphic encryption techniques to encrypt the data while allowing enroute aggregation. Lindell *et al.* [69] reviewed a number of techniques that use secure multi-party computation (SMC) that can ensure no party can learn anything beyond the output at the end of the computation. A modulo addition-based encryption scheme was introduced in [73] to realize differential privacy-preserving aggregation for smart metering systems. All these solutions require secure communication channels, pre-established shared secret/keys, and a trusted authority and usually incur high computation and communication overhead, which is undesirable for large-scale wireless sensor networks.

The second category uses random perturbation [70, 11, 71], in which each sensor node randomly perturbs its data according to a suitable probability distribution before participating in data aggregation, and the base station can still infer valuable statistics from the perturbed data. The seminar work [70] introduced the concept of Differential Privacy which ensure the computation results over two adjacent databases are indistinguishable. He *et al.* [11] proposed (CPDA), a cluster-based private data aggregation (CPDA) which adds random seeds into the original data. Yang *et al.*[71] proposed a machine learning based privacy protection mechanism using differential privacy in a fog computing architecture. To the best of our knowledge, there is no prior work tackling privacy-preserving quantile summary aggregation.

4.2.2 Local Differential Privacy

LDP has been studied extensively for various data analysis problems, including frequency estimation [74, 50, 75, 76, 77, 53], heavy hitter identification [78, 56, 79, 57], regular itemset mining [80, 81], marginal release [51], spatiotemporal data aggregation [82], and range queries [83]. In Ref. [74], a locally differentially private frequency

estimation scheme was introduced based on data encoding using Hadamard matrix. Another frequency estimation oracle was proposed by Bassily and Smith [50], which uses Random Matrix Projection. Google and Microsoft have deployed LDP in their applications such as RAPPOR [76] for website browsing history aggregation and privately collecting telemetry data [75]. Qin *et al.* [77] proposed MEFA, a scheme under LDP that calculates the input frequency through maximum likelihood estimation. Wang *et al.* [53] introduced an abstract framework for frequency estimation oracles which makes it possible to compare different protocols and analyze their privacy guarantee. Heavy hitters is closely related to frequency estimation where the goal is to find the most frequent items in a set and compute their frequencies. Wang *et al.* [78] presented a trie-based solution for new word discovery under LDP, which can efficiently find new words with high frequency by spanning the nodes with large supports. LDPMiner [56] and TreeHist [79] are two algorithms for heavy hitters identification under LDP, both of which adopt a two-stage approach. In the first stage, a portion of the privacy budget is used to learn a candidate set, and the remaining privacy budget is used in the second stage to refine the estimates of the candidates. LDPMiner focuses on set-values data while TreeHist considers a single value element for each user. In [57], an LDP scheme was presented to identify heavy hitters in a large domain where users are divided into groups and each group reports a prefix of its value. Wang *et al.* [80] proposed a protocol to identify the frequent itemsets which provides a better accuracy than LDPMiner through privacy amplification under sampling. A recent work [81] introduced an iterative approach to estimate the frequent itemsets under LDP with high accuracy using a two-level randomization technique by exploiting the correlation of the presence of items in a user's itemset. In [51], the authors provided a set of algorithms for materializing marginal statistics under LDP. For private spatial data aggregation problem, Chen *et al.* [82] used frequency estimation as a primitive to learn user distribution guaranteeing personalized LDP. Kulkarni *et al.* [83] designed a method to realize range queries with LDP and ensures good accuracy using Haar wavelet transform. There are a number of studies that target LDP over set-value data including [76, 84, 60, 56, 85]. Most of

these solutions [76, 84, 60, 56] divide the privacy budget into multiple portions used in different steps, which result in the reduction in data utility. In contrast, a scheme was introduced in [85] which sanitizes set-valued data as a whole by randomly outputting a subset of items without the need to split the privacy budget. All these solutions assume that data contributors directly submit their data to a data collector after perturbation without involving any en-route aggregation. As a result, none of existing solutions can be applied to privacy-preserving quantile summary aggregation.

4.3 Problem Formulation

In this section, we first introduce the network model and a background on quantile summary. We then provide the definition of Local Differential Privacy.

4.3.1 Network Model

We consider a wireless sensor network model consisting of a base station and n sensor nodes. Let $R = \{1, \ldots, d\}$ be the range of possible readings. We assume that every sensor node i has a set of m readings $V_i = \{v_{i,1}, \ldots, v_{i,m}\}$, where every reading $v_{i,j} \in R$ for all $1 \leq i \leq n$ and $1 \leq j \leq m$. The set of all the sensed data generated in the sensor network is then $V = \bigcup_{i=1}^{n} V_i$. The base station aims to obtain a quantile summary of V.

4.3.2 Quantile Summary

As we discussed in Chapter 3, a quantitle summary is a subset of readings along with their (estimated) global ranks which can support *value-to-rank* query over any $v \in R$ as well as ϕ-quantile queries for any $0 < \phi < 1$. Specifically, given a set of N distinct data values with a total order, the ϕ-quantile is the value v with rank $r(v) = \lfloor \phi N \rfloor$ in the set, where $r(v)$ is the number of values in the set smaller than v. Since a quantile summary that can provide the exact quantiles must contain the all N values in the worst case, an ϵ'-approximate ϕ-quantile is a value with rank between $(\phi - \epsilon')N$ and $(\phi + \epsilon')N$.

4.3.3 Local Differential Privacy (LDP)

Local Differential Privacy is a strong privacy notion widely considered as the gold standard for data privacy, which ensures that an adversary cannot differentiate two inputs based on the output he observe beyond certain predefined threshold. We give the definition of ϵ-Local Differential Privacy below.

Definition 1. *(ϵ-Local Differential Privacy). A randomized mechanism $\mathcal{M}$ satisfies ϵ-local differential privacy if and only if*

$$\frac{Pr[\mathcal{M}(x) = y]}{Pr[\mathcal{M}(x') = y]} \leq e^{\epsilon}$$

for any two inputs $x, x' \in X$ and any output $y \subseteq Range(\mathcal{M})$, where X is the domain of the input, $Range(\mathcal{M})$ is the domain of the output, and ϵ is commonly referred to as the privacy budget.

We can see that the smaller the privacy budget ϵ, the more indistinguishable of the two probability distributions $\mathcal{M}(x)$ and $\mathcal{M}(x')$ induced by the mechanism $\mathcal{M}$, and the more difficult for anyone to infer the input from the output y. The model of LDP differs from the centralized DP [70] where the data collector (i.e., base station) is considered trusted.

4.3.4 Design Goals

We seek to design a privacy-preserving quantile summary aggregation scheme with the following goals in mind.

- *Local Differential Privacy.* The scheme should satisfy ϵ-LDP for individual sensor nodes.
- *High accuracy.* The quantile summary obtained by the base station should be able to answer value-to-rank queries with high accuracy.
- *Communication efficiency.* The scheme should incur low communication overhead.

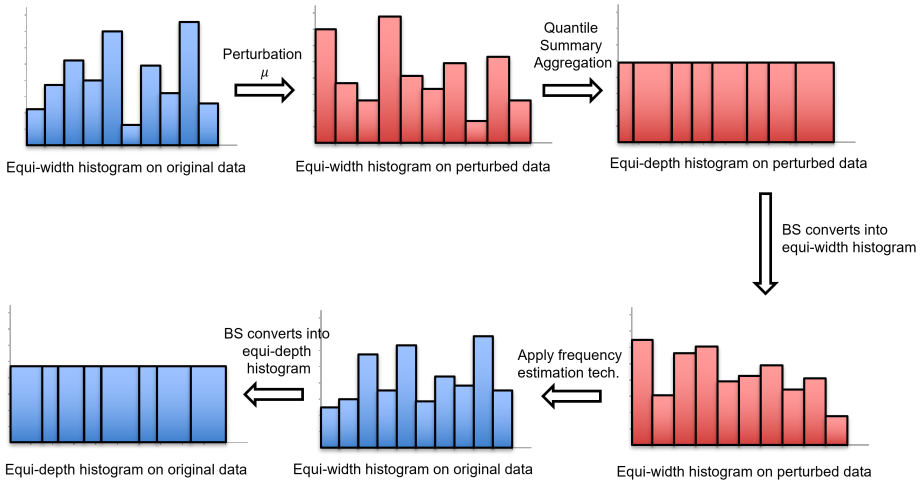


Figure 4.1: A high level idea of the key steps of PrivQSA.

4.4 PrivQSA: Quantile Summary Aggregation with LDP

In this section, we first give an overview of PrivQSA and then detail its design.

4.4.1 Overview

We design PrivQSA by exploring the inherent connection between a quantile summary and a histogram. Specifically, a quantile summary can be viewed as an equi-depth histogram in which every bucket has the same number of values, and all the buckets in a standard histogram have the same width but different number of values. In addition, a quantile summary can be converted into a standard histogram under moderate assumption, and vice versa. Based on this idea, we first let every sensor node randomly perturb its set of readings to generate a set of perturbed readings to ensure ϵ-LDP. All the sensor nodes then participate in a quantile summary aggregation to allow the base station to receive a quantile summary, i.e., an equi-depth histogram, of the perturbed readings, The base station can then convert the quantile summary of the perturbed readings into a an equi-width histogram whereby to estimate an equi-width histogram of the original readings based on the randomized perturbation mechanism used at individual sensor nodes. Finally, the base station can convert the estimated equi-width histogram of the original readings into a quantile summary of the original

readings whereby to answer any value-to-rank and percentile queries. The high level idea of PrivQSA kety steps is illustrated in Fig. 4.1. In what follows, we detail the design of PrivQSA.

4.4.2 Detailed Design

PrivQSA consists of the following six steps.

4.4.2.1 Perturbation at Individual Sensor Nodes

Each sensor node i randomly perturbs its set of readings $V_i = \{v_{i,1}, \ldots, v_{i,m}\}$ into a set of n perturbed readings S_i' via the Exponential Mechanism to ensure ϵ-LDP.

The exponential mechanism is a classical technique to provide differential privacy via outcome randomization. The key idea is to associate every pair of input x and candidate outcome o with a real-value quality score $q(x, o)$, where a higher quality score indicates higher utility of the outcome. Given the output space O, a score function $q(\cdot, \cdot)$, and the privacy budget ϵ, the exponential mechanism randomly selects an outcome $o \in O$ with probability proportional to $\exp(\epsilon q(x, o))$.

In our context, the input is a set $V_i \subseteq R$ of m readings, and the outcome $\tilde{V}_i$ of the exponential mechanism is also a subset of R with m elements. For every possible input set V_i and output set $\tilde{V}_i$, we define the quality score function as

$$q(V_i, \tilde{V}_i) = \frac{|V_i \bigcap \tilde{V}_i|}{m} , \tag{4.1}$$

which is the ratio of their common elements. For example, if $V_i = \tilde{V}_i$, then the quanlity score is one. As another example, if $V_i \bigcap \tilde{V}_i = \emptyset$, then the quality score is zero. Under the quality score function $q(\cdot, \cdot)$, each node i then randomly chooses an m-element set $V_i \subset R$ with probability proportional to $\exp(\frac{\epsilon |V_i \bigcap \tilde{V}_i|}{m})$.

It is necessary to detail the procedures for the above random perturbation because the number of m-element subsets is $\binom{d}{m}$ and naive sampling the output space would incur a computation complexity of $\mathcal{O}(d^m)$.

We first compute the probability that each m-element set $V_i \subset R$ is selected by the exponential mechanism. We notice that the number of m-element subsets of R that shared k common elements with V_i is

$$c_k = \binom{m}{k} \cdot \binom{d-m}{m-k} , \tag{4.2}$$

for all $0 \leq k \leq m$. Assume that each set $\tilde{V}_i$ is selected by the exponential mechanism with probability $\exp(\frac{\epsilon|V_i \bigcap \tilde{V}_i|}{m}) \cdot \rho$, where ρ is some constant. We then have

$$\sum_{k=0}^{m} c_k \cdot \exp(\frac{\epsilon k}{m}) \cdot \rho = 1 . \tag{4.3}$$

Solving the above equation, we have

$$\rho = \frac{1}{\sum_{k=0}^{m} c_k \cdot \exp(\frac{\epsilon k}{m})} , \tag{4.4}$$

where c_k is given in Eq. (4.2). Therefore, each m-element set $\tilde{V}_i \subset R$ is selected with probability $\exp(\frac{\epsilon|V_i \bigcap \tilde{V}_i|}{n}) \cdot \rho$, where ρ is given by Eq. (4.4). We give the formal definition of the perturbation mechanism below.

Definition 2. *(Set Perturbation (d, m, ϵ)). Given an original m-element set V_i, select the output m-element set $\tilde{V}_i \subset R$ with probability*

$$Pr[\mathcal{M}(V_i) = \tilde{V}_i] = \frac{\exp(\frac{\epsilon|V_i \bigcap \tilde{V}_i|}{m})}{\sum_{k=0}^{m} c_k \cdot \exp(\frac{\epsilon k}{m})} , \tag{4.5}$$

where $c_k = \binom{m}{k} \cdot \binom{d-m}{m-k}$ for all $k = 0, 1, \ldots, m$.

We further introduce an efficient algorithm for each node i to generate a perturbed set $\tilde{V}_i$ from its original set of readings V_i. Instead of sampling over all m-element subsets of R, we first determine the number of common elements between V_i and the output set $\tilde{V}_i$. Since there are c_k m-elements subsets that share k common elements with $\tilde{V}_i$, and each of these c_k sets is chosen with probability $\frac{\exp(\frac{\epsilon k}{m})}{\sum_{i=0}^{m} c_i \cdot \exp(\frac{\epsilon i}{m})}$, it follows that the probability of selecting a subset that shares k common elements with V_i is given by

$$p_k = \frac{c_k \exp(\frac{\epsilon k}{m})}{\sum_{i=0}^{m} c_i \cdot \exp(\frac{\epsilon i}{m})} . \tag{4.6}$$

Once we choose the number of common elements shared between V_i and the output set $\tilde{V}_i$, say k, we then randomly select k elements from V_i and $m-k$ elements from $R \setminus V_i$ to form the output set $\tilde{V}_i$. We summarize the procedure in Algorithm 1, in which the procedure $\mathsf{random}(0,1)$ returns a real number between 0 and 1 uniform at random, and the procedure $\mathsf{RandomSample}(X,k)$ returns k elements of set X uniform at random.

Algorithm 1: Set Perturbation

Input: Original set $V_i = \{v_{i,1}, \ldots, v_{i,n}\}$, domain $R = \{1, \ldots, d\}$, and privacy budget ϵ

Output: Perturbed set T_i

$\tilde{V}_i \leftarrow \emptyset$;
forall *k=0,1,...,m* **do**
  $p_k \leftarrow \frac{c_k \exp(\frac{\epsilon k}{m})}{\sum_{i=0}^{m} c_i \cdot \exp(\frac{\epsilon i}{m})}$;
$r \leftarrow \mathsf{random}(0,1)$;
$q \leftarrow 0$;
$p \leftarrow 0$;
while $p < r$ **do**
  $p \leftarrow p + p_q$;
  $q \leftarrow q + 1$;
$\tilde{V}_i \leftarrow \mathsf{RandomSample}(V_i, q)$;
$\tilde{V}_i \leftarrow \tilde{V}_i \bigcup \mathsf{RandomSample}(R \setminus V_i, m - q)$;
return $\tilde{V}_i$;

4.4.2.2 Data Augmentation

Since existing quantile summary aggregation schemes including Huang *et al.*'s protocol [1] requires every data value is distinct, every sensor node augments its perturbed readings by its node ID. Let $\tilde{V}_i = \{\tilde{v}_{i,1}, \ldots, \tilde{v}_{i,m}\}$ be node i's set of perturbed readings. Each node i augments each perturbed reading $\tilde{v}_{i,j}$ as

$$\hat{v}_{i,j} = \tilde{v}_{i,j} || i \ , \tag{4.7}$$

for all $1 \leq j \leq m$, where node ID i is encoded by $\gamma = \lceil \log_2 n \rceil$ bits. Doing so can ensure that every reading generated in the network is unique.

4.4.2.3 Quantile Summary Aggregation

All the sensor nodes then participate in quantile summary aggregation according to Huang *et al.*'s protocol [1]. Denote by $\hat{V}_i$ the set of perturbed and augmented readings of node i. Every node i randomly samples each perturbed reading $\hat{v}_{i,j} \in \hat{V}_i$ independently with probability h to obtain a subset of perturbed readings $S_i \subseteq \hat{V}_i$ and sends a local quantile summary to its parent node as

$$Q_i = \{(\hat{v}_{i,j}, j) | \hat{v}_{i,j} \in S_i\} \ , \tag{4.8}$$

where h is a system parameter, and j is the perturbed reading $\hat{v}_{i,j}$'s local rank within $\hat{V}_i$. As discussed in Chapter 3, intermediate nodes may merge multiple local quantile summaries into one to reduce the maximum per node communication cost. To simplify our discussion, we ignore any merging operation here as it does not affect any subsequent steps.

On receiving local quantile summaries $Q_1, \ldots, Q_n$ from all the sensor nodes, the base station performs value-to-rank query on every possible perturbed value to learn the distribution of the perturbed readings. Specifically, for every possible perturbed value $\hat{v} = v||i$ where $v \in R$ and $i \in \{1, \ldots, n\}$, the base station estimates its global rank among $\bigcup_{i=1}^{n} \hat{V}_i$ as

$$\hat{r}(\hat{v}) = \sum_{i=1}^{n} \hat{r}(\hat{v}, \hat{V}_i) \ , \tag{4.9}$$

where

$$\hat{r}(\hat{v}, \hat{V}_i) = \begin{cases} r(p(\hat{v}|Q_i), \hat{V}_i) + 1/h, & \text{if } p(\hat{v}|Q_i) \text{ exists;} \\ 0 & \text{otherwise,} \end{cases} \tag{4.10}$$

As discussed in Chapter 3, $\hat{r}(\hat{v})$ is an unbiased estimator of $r(\hat{v}, \bigcup_{i=1}^{n} \hat{V}_i)$.

Next, the base station computes the global rank of each possible value $v \in R$ by removing the augmented node ID from the perturbed readings. In particular, for each pair of perturbed value and estimated rank $(\hat{v}, \hat{r}(\hat{v}))$, the base station updates its value as

$$\tilde{v} = \hat{v} \mod 2^{\gamma} \tag{4.11}$$

and records $(\tilde{v}, \hat{r}(\hat{v}))$.

After removing the augmented IDs from all perturbed readings, the base station obtains one or more estimated global ranks for each possible value $v \in R$. Without loss of generality, let $r^-(v)$ and $r^+(v)$ be the lowest and highest estimated global ranks, respectively, of value v for all $v \in R$. If value v has only a unique estimated global rank, then $r^-(v) = r^+(v)$.

4.4.2.4 Histogram Construction

The base station then constructs a histogram of the perturbed readings from the received quantile summaries by estimating the frequency of each value $v \in R$.

We formulate the histogram construction as an optimization problem. In particular, let f_v be the frequency of value v for all $v \in R$. It follows that value 1 is ranked from the 1st to the f_1th, and value v is ranked from $(\sum_{i=1}^{v-1} f_i + 1)$th to $(\sum_{i=1}^{v} f_i)$th for all $1 \leq v \leq d$.

We formulate the estimation of $f_1, \ldots, f_d$ as the following optimization problem

$$\begin{aligned} \min_{(f_1,\ldots,f_d)\in\mathbb{N}^d} \quad & \mathcal{F}(f) = \sum_{v\in R}(\sum_{i=1}^{v-1} f_i + 1 - r^-(v))^2 + (\sum_{i=1}^{v} f_i - r^+(v))^2, \\ \text{such that} \quad & \sum_{v=1}^{d} f_v = nm, \\ & \sum_{i=1}^{v-1} f_i + 1 \leq r^-(v), \forall v \in R, \\ & \sum_{i=1}^{v} f_i \geq r^+(v), \forall v \in R, \end{aligned} \tag{4.12}$$

where we seek to minimize the total square errors between the two boundaries and the corresponding lowest and and highest estimated global ranks. In the above optimization problem, the first constraint indicates that the sum of all the values' frequencies should be nm, the second and third constraints guarantee that the lowest and highest estimated global ranks of a value v, i.e., $r^-(v)$ and $r^+(v)$, should fall in the range $[\sum_{i=1}^{v-1} f_i + 1, \sum_{i=1}^{v} f_i]$.

The partial derivatives of $\mathcal{F}(f)$ with respect to f_j can be computed as

$$\begin{aligned}\frac{\partial \mathcal{F}}{\partial f_j} = 2(\sum_{i=1}^{j} f_i - r^+(j)) + \sum_{v=j+1}^{d-1} (2(\sum_{i=1}^{v-1} f_i + 1 - r^-(v)) \\ + 2(\sum_{i=1}^{v} f_i - r^+(v))) + 2(\sum_{i=1}^{d-1} f_i + 1 - r^-(d))) \ ,\end{aligned} \tag{4.13}$$

for all $j \in R$.

Let $\frac{\partial \mathcal{F}}{\partial f_j} = 0$ for all $j \in R$. We can solve the system of linear equations by obtain

$$\begin{cases} f_1 = \frac{r^+(1)+r^-(2)-1}{2}, \\ f_i = \frac{r^+(i)+r^-(i+1)-r^+(i-1)-r^-(i)}{2}, \quad \forall 2 \leq i \leq d-1, \\ f_d = nm - \frac{r^+(d-1)+r^-(d)-1}{2}. \end{cases}$$

We summarize the construction of the histogram in Algorithm 2.

Algorithm 2: Histogram Construction

Input: $\{(v, r^-(v), r^+(v)) | v \in R\}$ (Estimated value ranks), d (Size of original domain)

Output: $\{(v, f) | v \in R\}$ (Histogram of perturbed values)

```
for v ∈ R do
    if v = 1 then
        f_v = (r⁺(v)+r⁻(v+1)−1)/2;
    if 1 < v < d then
        f_v = (r⁺(v)+r⁻(v+1)−r⁺(v−1)−r⁻(v))/2;
    if v = d then
        f_v = nm − (r⁺(v−1)+r⁻(v)−1)/2;
return {(v, f_v) | v ∈ R}
```

4.4.2.5 Estimating Histogram of Original Readings

Given the estimated histogram of perturbed readings $f_1, \ldots, f_d$ obtained above, the base station proceed to estimate the histogram of original readings.

The key idea is to ignore the quantile summary aggregation process and any loss of accuracy caused by the random sampling. Instead, we view the estimated histogram

of perturbed readings $f_1, \ldots, f_d$ as if they are obtained by having every sensor nod i submit all of its perturbed readings $\tilde{V}_i$ and counting the number of each value.

We can then estimate the histogram of original readings based on the set perturbation mechanism at individual sensor nodes. Let g_v be the frequency of value v among the original readings $\bigcup_{i=1}^{n} V_i$ for all $v \in R$.

Consider an item $v \in R$ and sensor node i's original reading set V_i. Given the perturbation mechanism in Definition 2, if $v \in V_i$, the probability that v shows up in the perturbed set $\tilde{V}_i$ can be computed as

$$\begin{aligned} Pr[v \in \tilde{V}_i | v \in V_i] &= Pr[v \in V_i \bigcap \tilde{V}_i | v \in V_i] \\ &= \sum_{k=1}^{m} Pr[v \in V_i \bigcap \tilde{V}_i | v \in V_i, |V_i \bigcap \tilde{V}_i| = k] \cdot Pr[|V_i \bigcap \tilde{V}_i| = k] \ . \end{aligned} \tag{4.14}$$

We further have

$$Pr[v \in V_i \bigcap \tilde{V}_i | v \in V_i, |V_i \bigcap \tilde{V}_i| = k] = \frac{k}{m} \ , \tag{4.15}$$

Substituting Eqs. (4.15) and (4.6) into Eq. (4.14), we have

$$Pr[v \in \tilde{V}_i | v \in V_i] = \sum_{k=1}^{m} \frac{k}{m} \cdot \frac{c_k \exp(\frac{\epsilon k}{m})}{\sum_{j=0}^{m} c_j \cdot \exp(\frac{\epsilon j}{m})} \ . \tag{4.16}$$

Now let us analyze the probability that v shows up in the perturbed set $\tilde{V}_i$ given that $v \notin V_i$. We have

$$\begin{aligned} Pr[v \in \tilde{V}_i | v \notin V_i] &= Pr[v \in \tilde{V}_i \setminus V_i | v \notin V_i] \\ &= \sum_{k=0}^{m-1} Pr[v \in \tilde{V}_i \setminus V_i | v \notin V_i, |V_i \bigcap \tilde{V}_i| = k] \cdot Pr[|V_i \bigcap \tilde{V}_i| = k] \ . \end{aligned} \tag{4.17}$$

Since

$$Pr[v \in \tilde{V}_i \setminus V_i | v \notin V_i, |V_i \bigcap \tilde{V}_i| = k] = \frac{m-k}{d-m} \ , \tag{4.18}$$

Substituting Eqs. (4.18) and (4.6) into Eq. (4.17), we have

$$Pr[v \in \tilde{V}_i | v \notin V_i] = \sum_{k=0}^{m-1} \frac{m-k}{d-m} \cdot \frac{c_k \exp(\frac{\epsilon k}{m})}{\sum_{j=0}^{m} c_j \cdot \exp(\frac{\epsilon j}{m})} \ . \tag{4.19}$$

Assume that value v appears in g_v sensor nodes' original reading sets. It follows that $n - g_v$ sets do not contain value v. The expected number of perturbed sets that include value v can be estimated as

$$E[f_v] = g_v \cdot Pr[v \in \tilde{V}_i | v \in V_i] + (n - g_v) \cdot Pr[v \in \tilde{V}_i | v \notin V_i] \tag{4.20}$$

Solving the above equation, we have

$$g_v = \frac{E[f_v] - n \cdot Pr[v \in \tilde{V}_i | v \notin V_i]}{Pr[v \in \tilde{V}_i | v \in V_i] - Pr[v \in \tilde{V}_i | v \notin V_i]}, \tag{4.21}$$

where $Pr[v \in \tilde{V}_i | v \in V_i]$ and $Pr[v \in \tilde{V}_i | v \notin V_i]$ are given in Eq. (4.16) and (4.19), respectively. We therefore estimate the number of copies of v in $\bigcup_{i=1}^{n} V_i$ as

$$\hat{g}_v = \frac{f_v - n \cdot Pr[v \in \tilde{V}_i | v \notin V_i]}{Pr[v \in \tilde{V}_i | v \in V_i] - Pr[v \in \tilde{V}_i | v \notin V_i]}, \tag{4.22}$$

where $Pr[v \in \tilde{V}_i | v \in V_i]$ and $Pr[v \in \tilde{V}_i | v \notin V_i]$ are given in Eqs. (4.16) and (4.19), respectively.

Algorithm 3: Estimating Original Histogram

Input: $\{(v, f_v) | v \in R\}$ (Histogram of perturbed values), R (original domain), n (the number of nodes)

Output: $\{(v, g_v) | v \in R\}$ (Histogram of original values)

1 **for** $v \in R$ **do**

2 $\quad x_v \leftarrow \sum_{k=0}^{m-1} \frac{m-k}{d-m} \cdot \frac{c_k \exp(\frac{\epsilon k}{m})}{\sum_{j=0}^{m} c_j \cdot \exp(\frac{\epsilon j}{m})}$;

3 $\quad y_v \leftarrow \sum_{k=1}^{m} \frac{k}{m} \cdot \frac{c_k \exp(\frac{\epsilon k}{m})}{\sum_{j=0}^{m} c_j \cdot \exp(\frac{\epsilon j}{m})}$;

4 $\quad g_v \leftarrow \frac{f_v - x_v}{y_v - x_v}$;

5 **return** $\{(v, g_v) | v \in R\}$

4.4.2.6 Final Quantile Summary Construction

Given the estimated histogram of the original readings obtained in the previous step, the base station then constructs a final quantile summary of the original readings, which is equivalent to estimating the rank for every value $v \in R$ and answering ϕ-quantile query for all $0 < \phi < 1$.

To answer a value-to-rank query over value $v \in R$, the base station can simply return the median rank of value v as

$$r(v) = \lfloor \frac{r^-(v) + r^+(v)}{2} \rfloor \ , \tag{4.23}$$

where $r^-(v) = \sum_{i=0}^{v-1} p_i + 1$ and $r^+(v) = \sum_{i=0}^{v} p_i$.

Moreover, to answer a ϕ-quantile query where $0 < \phi < 1$, the base station returns value v such that

$$r^-(v) \leq nm\phi \leq r^+(v) \ . \tag{4.24}$$

4.5 Theoretical Analysis

Theorem 4. *The set perturbation mechanism (d, m, ϵ) of the PrivQSA mechanism satisfies ϵ-LDP.*

Proof. Let V_1 and V_2 be two arbitrary sets of readings such that $|V_1| = |V_2| = m$. Let $\mathcal{M}$ denote the randomized mechanism given by Definition 2, and $\tilde{V}$ be any possible output of $\mathcal{M}$. Since $0 \leq |\tilde{V} \cap V_1| \leq m$ and $0 \leq |\tilde{V} \cap V_2| \leq m$ for any V_1 and V_2, we have

$$\begin{aligned} \frac{\mathsf{Pr}[\mathcal{M}(V_1) = \tilde{V}]}{\mathsf{Pr}[\mathcal{M}(V_2) = \tilde{V}]} &= \frac{\exp(\epsilon \cdot \frac{|\tilde{V} \cap V_1|}{m})}{\exp(\epsilon \cdot \frac{|\tilde{V} \cap V_2|}{m})} \\ &\leq \frac{\exp(\epsilon \cdot \frac{m}{m})}{\exp(\epsilon \cdot \frac{0}{m})} \\ &\leq \exp(\epsilon) \ . \end{aligned} \tag{4.25}$$

The theorem is thus proved. □

4.6 Simulation Results

In this section, we evaluate the performance of PrivQSA via simulation studies.

Table 4.1: Default Simulation Settings

Para.	Val.	Description.
ϵ	50	The privacy budget
h	0.5	The sampling probability
n	1022	The number of sensor nodes
m	10	The size of user value set
d	100	The maximum number in the range of user values

4.6.1 Simulation Settings

We simulate a wireless sensor network consisting of $n = 1022$ sensor nodes. We assume that each node has $m = 10$ readings and every reading is in the range $R = 1, \ldots, 100$. We consider every reading is of one byte, and every rank is of 3 bytes. Table 4.1 summarizes our default settings unless mentioned otherwise. Every point in the following graphs is the average of 10 runs, each with a distinct random seed.

Since there is no prior solution for private quantile summary aggregation, we compare PrivQSA with the following two baseline schemes.

- *Baseline 1*: Following the first two steps of PrivQSA, every node randomly perturbs its m-element reading set into another m-element reading set using the perturbation scheme in Definition 2 and Algorithm 1. Each node then independently samples its perturbed readings with probability h and submits only the sampled readings without corresponding ranks to the base station. The base station constructs a histogram of the perturbed values frequencies and then estimates the missing items by multiplying the frequency of each value with $1/h$. Finally, the base station estimates the original distribution and the final quantile summary using the same method as in Steps 6 and 7 whereby to answer value-to-rank queries. Baseline 1 satisfies ϵ-LDP without involving quantile summary aggregation.
- *Baseline 2*: Every node independently samples its readings according to Huang *et al.* [1] scheme with probability h and then submits the sampled readings along with their associated ranks to the base station. The base station then estimates

the global rank of each reading to answer any value-to-rank query. Baseline 2 is a quantile summary aggregation protocol without any privacy guarantee.

To evaluate the performance of PrivQSA, we use two metrics to measure the accuracy of the final quantile summary at the base station. Let $r(v)$ and $\hat{r}(v)$ be the true rank and estimated rank of a value v, respectively, for all $v \in \{1, \ldots, d\}$. Also let $r_{\max} = nm$ be the maximum rank in the network which is the total number of readings in the network. The normalized average rank error (ARE) is defined as

$$\mathsf{ARE} = \frac{\sum_{v=1}^{d} |\hat{r}(v) - r(v)|}{r_{\max} d} , \tag{4.26}$$

and the maximum rank error (MRE) is defined as

$$\mathsf{MRE} = \frac{\max_{v=\{1,\ldots,d\}}(|\hat{r}(v) - r(v)|)}{r_{\max}} . \tag{4.27}$$

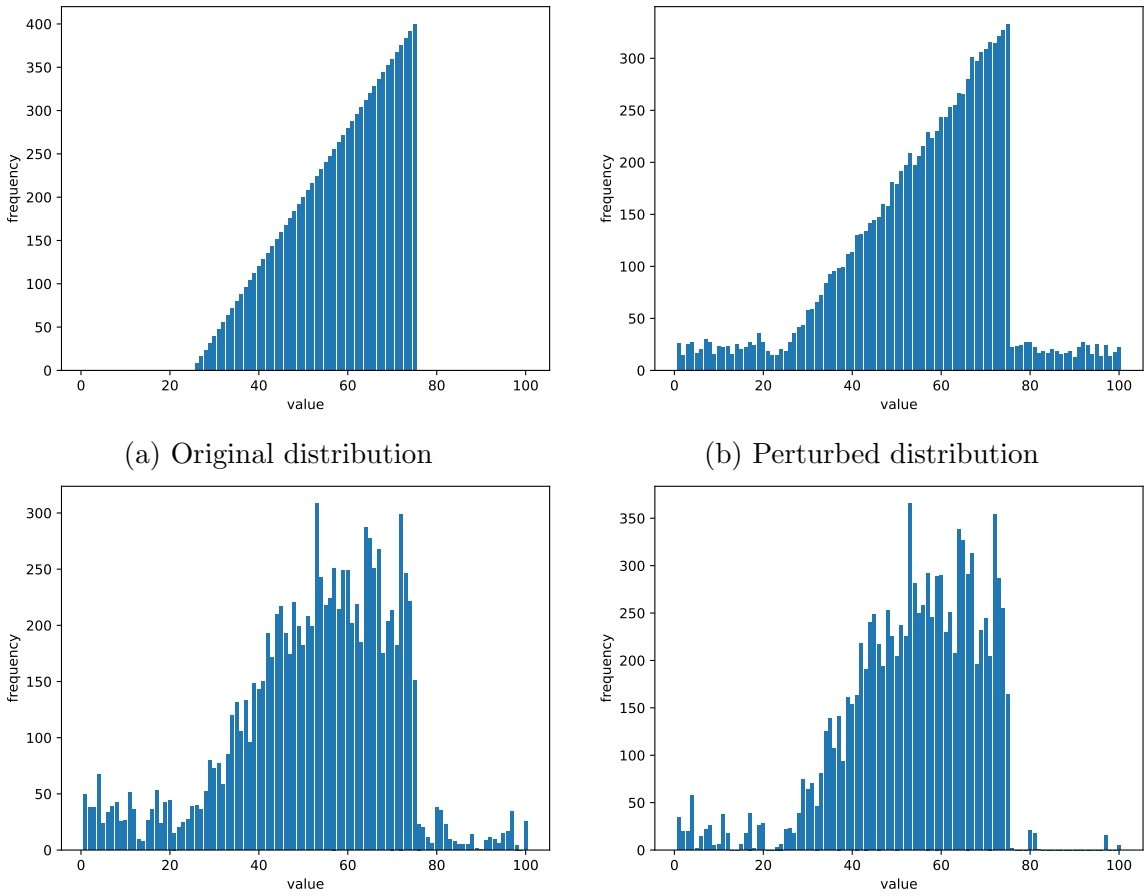
In addition, we also use total communication cost and maximum per node communication cost to compare the performance of PrivQSA and the two baseline solutions.

4.6.2 Simulation Results

We now report our simulation results.

4.6.2.1 Examples of Data Processing under PrivQSA

Figs. 4.2a to 4.2d give examples of the output of PrivQSA in different steps to provide a high level idea of how PrivQSA works. Specifically, Fig. 4.2a shows the distribution of the original readings generated by the sensor network. Fig. 4.2b shows the distribution of all perturbed readings before sampling. We can see that the difference between Fig. 4.2a and Fig. 4.2b is that the distribution of the perturbed readings is more flat than that of the original readings, which is due to the application of the set perturbation mechanism based on the exponential mechanism. Fig. 4.2c shows the distribution of the perturbed data learned from quantile summary aggregation. The difference between Fig. 4.2b and Fig. 4.2c is because of the error produced by the sampling process involved in the quantile summary aggregation. Finally, Fig. 4.2d



(a) Original distribution

(b) Perturbed distribution

(c) Perturbed distribution learned from quantile summary aggregation

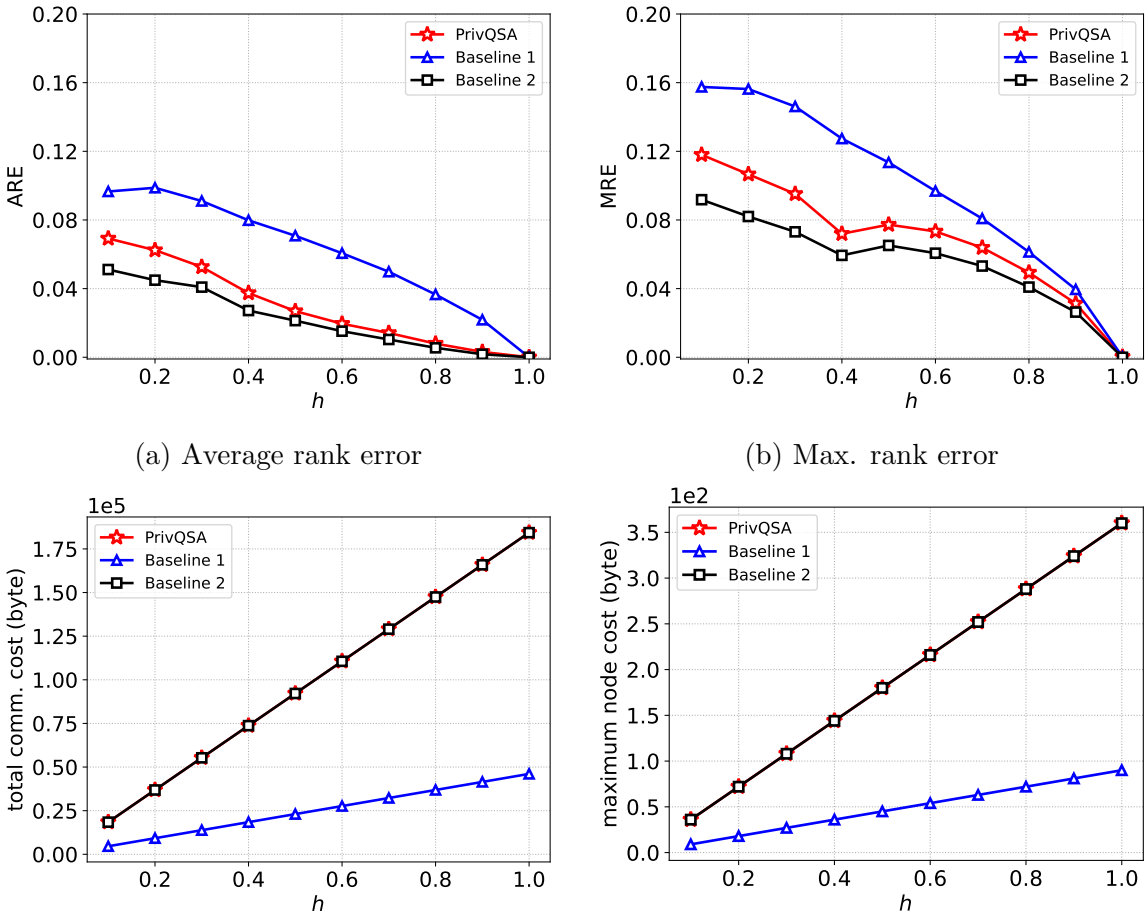
(d) Estimated original distribution

Figure 4.2: Examples of data processing by PrivQSA in different steps.

shows the distribution of original readings estimated using Algorithm 3. The difference between Fig. 4.2a and Fig. 4.2d is due to the perturbation and quantile summary aggregation involved in PrivQSA.

4.6.2.2 Impact of Sampling Probability

Figs. 4.3a to 4.3d compare the ARE, MRE, total communication cost, and maximum per node communication cost of PrivQSA and the two baseline solutions, respectively, with sampling probability varying from 0.1 to 1.0. We can see from Fig. 4.3a and 4.3b that both the ARE and MRE decrease as the sampling probability h increases under all three schemes. This is expected as the more readings we sample,



(a) Average rank error

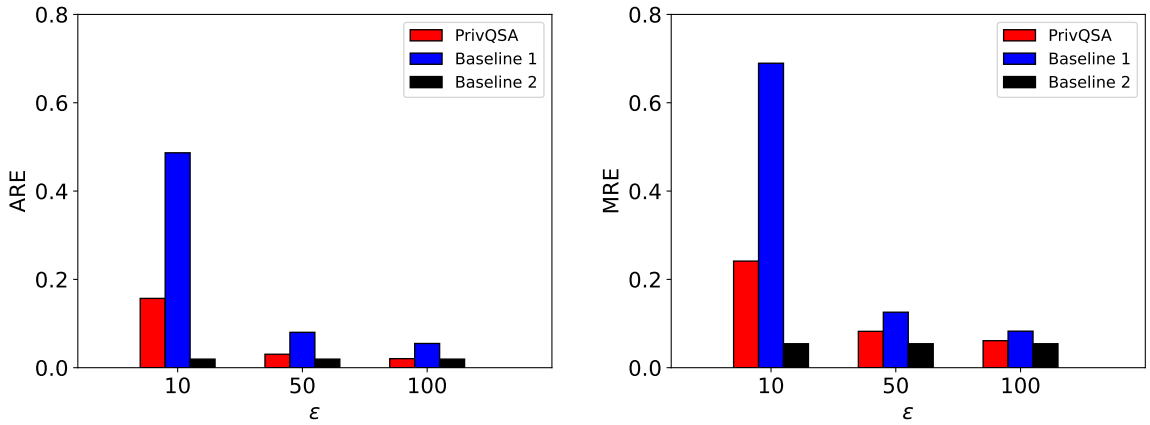
(b) Max. rank error

(c) Total communication cost

(d) Maximum per node cost

Figure 4.3: Comparison of PrivQSA and the baselines with sampling probability h varying from 0.1 to 1.0.

the more accurate the value-to-rank query results, and vice versa. Moreover, we can see from the same two figures that the ARE and MRE of Baseline 2 is the lowest in comparison with PrivQSA and Baseline 1 because it does not involve any random perturbation. PrivQSA comes in the second place with a small difference compared to Baseline 2 which is the cost of providing local differential privacy. Finally, Baseline 1 comes with the largest rank errors among the three due to the random perturbation and it does not involve rank information in estimating the original distribution. On the other hand, Fig.4.3c shows the total communication cost under PrivQSA and the other two Baselines. Generally speaking, we can see that the total communication cost



(a) Average rank error

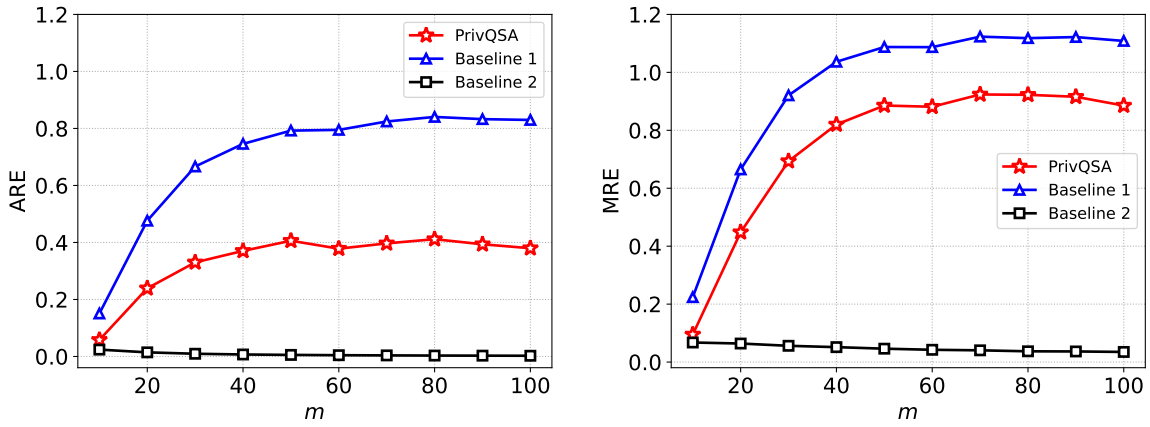
(b) Maximum rank error

Figure 4.4: Comparison of PrivQSA and the baselines with privacy budget ϵ varying from 10 to 100.

increases as the sampling probability increases. The reason is that, the more values are sampled, the more information need to be sent and accordingly the higher the total communication cost. In addition, we can see that PrivQSA and Baseline 2 have the same communication cost, which is larger than Baseline 1's communication cost. This is because under both PrivQSA and Baseline 2 every sensor node needs to send the rank information beside the sampled values whereas under Baseline 1 only sampled readings need to be sent. Fig. 4.3d plots the maximum per node communication costs of the three schemes, which shows that the maximum per node cost increases under all three schemes with the increase of the sampling probability, which is anticipated. Finally, it shows that PrivQSA and Baseline 2 incur similar communication cost which is also larger than Baseline 1's communication cost for the same reason mentioned above.

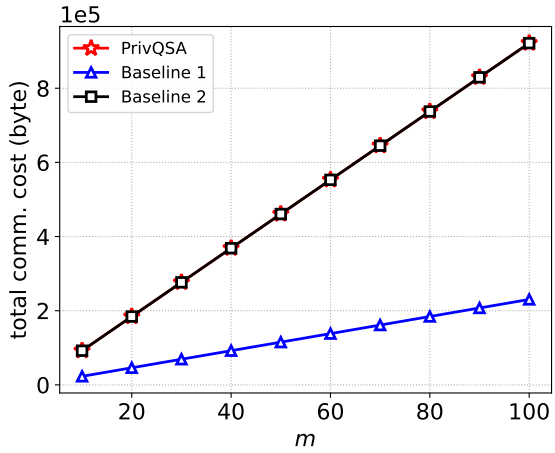
4.6.2.3 Impact of Privacy Budget

Figs. 4.4a and 4.4b compare the ARE and MRE under PrivQSA and Baseline 2 with the privacy budget ϵ varying from 10 to 100, where those under Baseline 2 are plotted for reference only as Baseline 2 does not involve any random perturbation and is not affected by the change in the privacy budget ϵ. We can see from both figures that

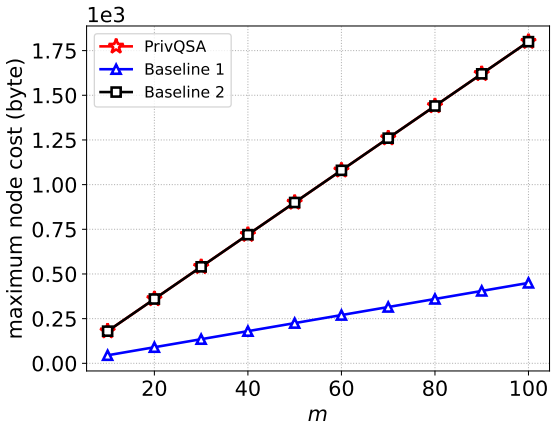


(a) Average rank error

(b) Maximum rank error



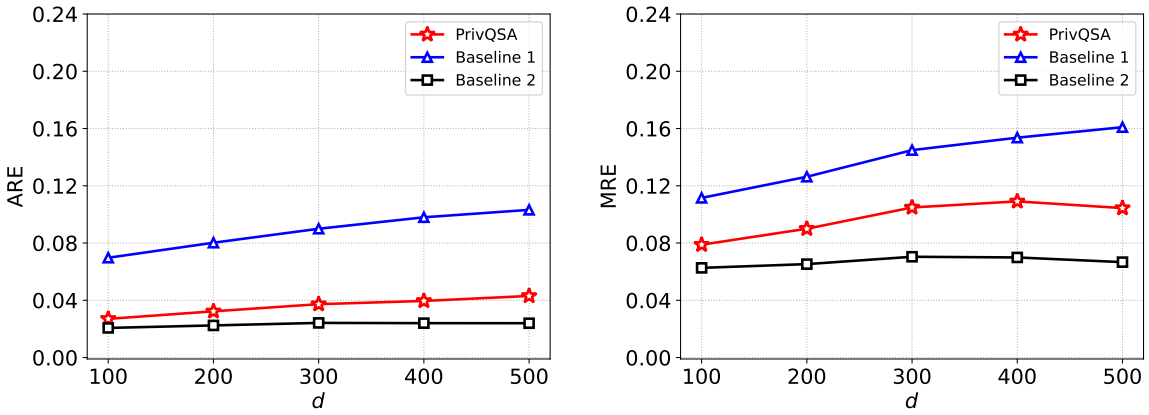
(c) Total communication cost



(d) Maximum per node cost

Figure 4.5: Comparison of PrivQSA and the two baseline solutions with set value size varying from 10 to 100.

both ARE and MRE decrease as the privacy budget ϵ increases both under PrivQSA and Baseline 1. This is because the larger the ϵ, the larger the size of the intersection of the original reading set and the perturbed reading set, the smaller the added noise, the more accurate the estimated original distribution, the more accurate the value-to-rank query results, and vice versa. In addition, we can see that Baseline 1 always has the largest ARE and MRE compared to PrivQSA and Baseline 2. PrivQSA comes in after with a lower ARE and MRE compared to Baseline 1 due to the rank information included in the quantile summary aggregation but is larger than that of Baseline 2 because of the random perturbation involved in PrivQSA.



(a) Average rank error

(b) Maximum rank error

Figure 4.6: Comparison of PrivQSA and the baselines with value domain varying from 100 to 500.

4.6.2.4 Impact of m

Figs. 4.5a to 4.5d compare the ARE, MRE, total communication cost, and maximum per node communication cost under PrivQSA, Baseline 1, and Baseline 2 with m, i.e., the number of readings per node varying from 10 to 100. As we can see from Figs. 4.5a and 4.5b, both ARE and MRE decrease as the number of reading per node increases under Baseline 2. As we mentioned in Chapter 3, the reason is that the more readings each node has, the more sampled readings, the more accurate the value-to-rank query results, and vice versa. In contrast, we can see from the same two figures that both ARE and MRE increase as the number of values per node increases under PrivQSA and Baseline 1. This is because perturbing a larger set of readings with the same privacy budget leads to adding larger noises to each reading in the set and lowering the accuracy of frequency estimation as well as larger rank errors. One more thing to notice from Figs. 4.5a and 4.5b is that again Baseline 2 has the lowest ARE and MRE compared to PrivQSA and Baseline 1 while PrivQSA comes in the second place and Baseline 1 is in the last place for the same reasons mentioned earlier. For the communication cost in Figs. 4.5c and 4.5d, both the total communication cost and maximum per node cost produced by the three schemes increase as the number of readings each node has increases. The reason is that, the more readings that each node

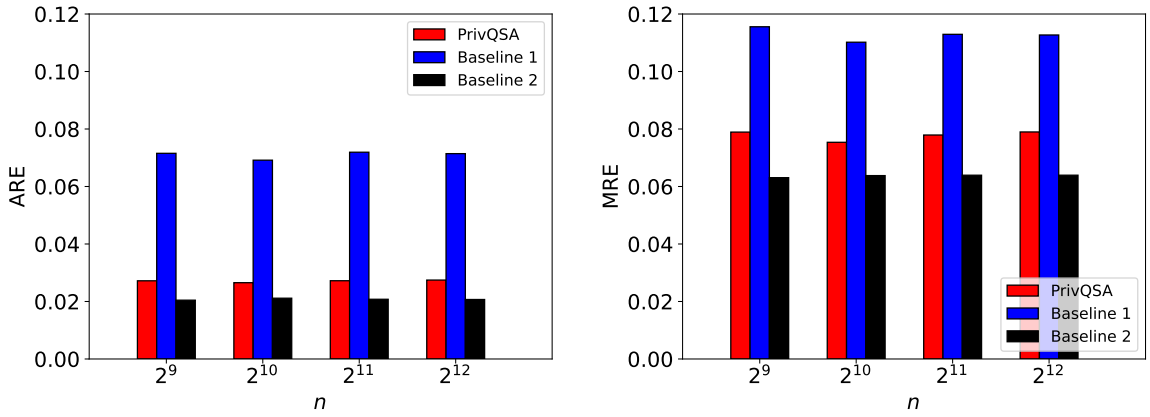
has, the more sampled readings, the more information need to be communicated and accordingly the more communication overhead they incur. Moreover, Figs. 4.5c and 4.5d show that Baseline 1 incurs the lowest communication cost among the three as explained in the previous paragraphs. On the other hand, PrivQSA and Baseline 2 incur the same communication cost either in total or per node because they communicate the same amount of information.

4.6.2.5 Impact of d

Figs. 4.6a and 4.6b compare the ARE and MRE produced by PrivQSA, Baseline 1, and Baseline 2 considering different sizes for the value domain in the network. As we can see in Figs. 4.6a and 4.6b, Baseline 2 shows a slight increase in both ARE and MRE as the size of the domain range increases. This is expected as the larger the domain range, the more values in the domain range that need to have their ranks estimated, the higher the ARE and MRE under a fixed sampling probability. For the same reason, we can see that PrivQSA and Baseline 1 achieve higher ARE and MRE which also increase faster in comparison with Basline 2 as the size of domain range increases. This is because the perturbation mechanism depends on the value range in that the larger the value domain, the fewer common elements between the original reading set and the perturbed reading set after perturbation, the larger the noise added, and the larger the rank estimation errors, and vice versa.

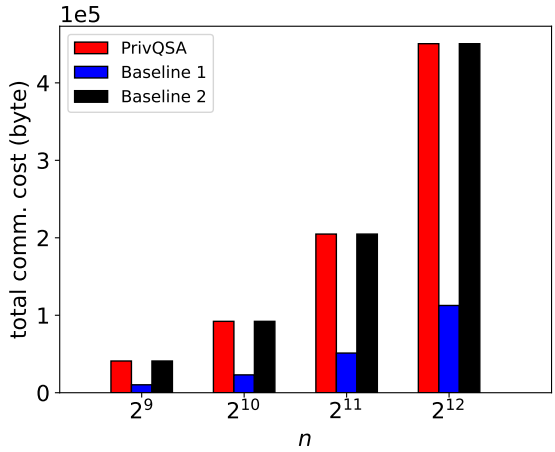
4.6.2.6 Impact of n

Figs. 4.7a to 4.7d show the ARE, MRE, total communication cost, and maximum per node cost produced by PrivQSA, Baseline 1, and Baseline 2, with n, i.e., the number of nodes varying from 512 to 2,048. As we can see, Figs. 4.7a and 4.7b show that the ARE and MRE are relatively insensitive to the increase in the number of nodes in the network. This is anticipated as the total number of readings produced by the sensor network is proportional to the number of nodes, and both ARE and MRE are normalized rank error that are inversely proportional to the total number of readings.

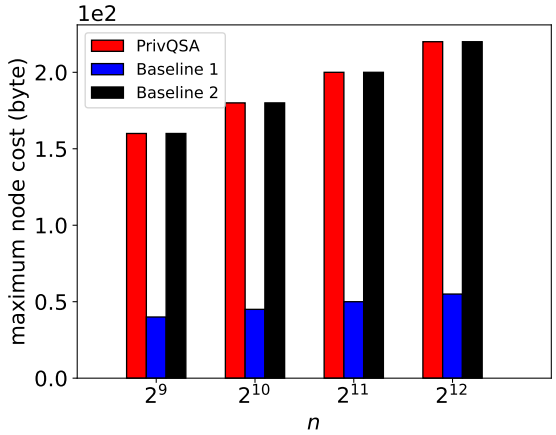


(a) Average rank error

(b) Maximum rank error



(c) Total communication cost



(d) Maximum per node cost

Figure 4.7: Comparison of PrivQSA and the baselines with number of nodes varying from 2^9 to 2^{12}.

As a result, while the absolute rank errors increase as the number of nodes and the total number of readings in the network increase, both ARE and MRE are relatively insensitive to the change in the number of nodes. Moreover, Figs. 4.7a and 4.7b again show that Baseline 1 always has the largest ARE and MRE compared to PrivQSA and Baseline 2, followed by PrivQSA and Baseline 2 for the same reasons discussed above. Figs. 4.7c and 4.7d show that the total communication cost and maximum per node cost under all three schemes increase as the number of nodes in the network increases. This is expected, becauase the more nodes, the more sampled readings, the more information to be communicated. Also, Figs. 4.7c and 4.7d show that PrivQSA and

Baseline 1 incur similar communication cost which are higher than that of Baseline 2 because both of them communicate more information than Baseline 1.

In summary, the above simulation results clearly demonstrate the advantages of PrivQSA over the two baseline solutions. On the one hand, PrivQSA incurs the same communication cost as Baseline 2 but provides ϵ-LDP with a slight decrease in the estimation accuracy. On the other hand, PrivQSA achieves a much better estimation accuracy than Baseline 1 but incurs only a slightly higher communication cost than Baseline 1 due to the rank information contained in the quantile summaries.

4.7 Summary

In this chapter, we have initiated the study of privacy-preserving quantile summary aggregation in wireless sensor networks. We introduced the design of PrivQSA, the first locally differentially private quantile summary aggregation protocol for wireless sensor networks, which can guarantee ϵ-LDP for individual sensor node's readings. We have confirmed the significant advantages of PrivQSA over alternative solutions via detailed simulation studies.

Chapter 5

CONCLUSION AND FUTURE WORK

In this dissertation, we have tackled three key security and privacy challenges in data aggregation in wireless sensor network. First, we have identified a novel enumeration attack against existing secure additive data aggregation schemes based on randomized sampling. Taking VMAT, a representative secure additive aggregation scheme, as an example, we show that even a single compromised sensor node can significantly modify the final aggregation result by forging its own reading. As a countermeasure, we have also introduced an effective defense against the enumeration attack and confirmed its effectiveness by simulation studies.

Second, we have presented SecQSA, the first secure quantile summary aggregation scheme for wireless sensor networks. Built upon the state-of-art quantile summary aggregation protocol and efficient cryptographic primitives, SecQSA can effectively defend against a range of malicious attacks launched by compromised sensor node. We have also confirmed its efficacy and efficiency via detailed simulation studies.

Finally, we have introduced PrivQSA, the first privacy-preserving quantile summary aggregation scheme for wireless sensor networks. PrivQSA allows a base station to obtain the quantile summary of the data generated in a wireless sensor network while providing ϵ-Local Differential Privacy for individual sensor nodes' readings. We have confirmed PrivQSA's advantages over alternative solutions via a combination of theoretical analysis and simulation studies.

As our future work, we plan to further investigate a number of issues. First, we plan to study the impact of enumeration attacks on other secure aggregation protocols and develop general defense mechanisms against the enumeration attack. Second, we

plan to extend SecQSA to detect and defend against multiple compromised sensor nodes by generalizing the local quantile summary merging operation to involve nodes further down the subtree. Third, we plan to explore other set perturbation mechanisms in PrivQSA and analyze the impact of random sampling on the level of LDP provision. Last but not the least, we plan to integrate SecQSA and PrivQSA to realize secure and privacy preserving quantile summary aggregation.

REFERENCES

[1] Z. Huang, L. Wang, K. Yi, and Y. Liu, "Sampling based algorithms for quantile computation in sensor networks," in *Proceedings of the 2011 ACM SIGMOD International Conference on Management of data*, 2011, pp. 745–756.

[2] H. Chan, A. Perrig, and D. Song, "Secure hierarchical in-network aggregation in sensor networks," in *ACM CCS'06*, 2006, pp. 278–287.

[3] Y. Yang, X. Wang, S. Zhu, and G. Cao, "Sdap: A secure hop-by-hop data aggregation protocol for sensor networks," *ACM TISSEC*, vol. 11, no. 4, pp. 1–43, 2008.

[4] K. B. Frikken and J. A. Dougherty IV, "An efficient integrity-preserving scheme for hierarchical sensor aggregation," in *ACM WiSec'08*, 2008, pp. 68–76.

[5] S. Papadopoulos, A. Kiayias, and D. Papadias, "Secure and efficient in-network processing of exact sum queries," in *IEEE ICDE'11*, 2011, pp. 517–528.

[6] B. Przydatek, D. Song, and A. Perrig, "Sia: Secure information aggregation in sensor networks," in *ACM SenSys'03*, 2003, pp. 255–265.

[7] S. Roy, M. Conti, S. Setia, and S. Jajodia, "Secure median computation in wireless sensor networks," *Ad Hoc Networks*, vol. 7, no. 8, pp. 1448–1462, 2009.

[8] H. Yu, "Secure and highly-available aggregation queries in large-scale sensor networks via set sampling," *Distributed Computing*, vol. 23, no. 5-6, pp. 373–394, 2011.

[9] B. Chen and H. Yu, "Secure aggregation with malicious node revocation in sensor networks," in *2011 31st International Conference on Distributed Computing Systems*. IEEE, 2011, pp. 581–592.

[10] S. Roy, M. Conti, S. Setia, and S. Jajodia, "Secure data aggregation in wireless sensor networks: Filtering out the attacker's impact," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 4, pp. 681–694, 2014.

[11] W. He, X. Liu, H. Nguyen, K. Nahrstedt, and T. Abdelzaher, "Pda: Privacy-preserving data aggregation in wireless sensor networks," in *IEEE INFOCOM 2007-26th IEEE International Conference on Computer Communications*. IEEE, 2007, pp. 2045–2053.

[12] S. Ozdemir and Y. Xiao, "Secure data aggregation in wireless sensor networks: A comprehensive overview," *Computer Networks*, vol. 53, no. 12, pp. 2022–2037, 2009.

[13] J. Shi, R. Zhang, Y. Liu, and Y. Zhang, "Prisense: privacy-preserving data aggregation in people-centric urban sensing systems," in *2010 Proceedings IEEE INFOCOM*. IEEE, 2010, pp. 1–9.

[14] M. M. Groat, W. Hey, and S. Forrest, "Kipda: k-indistinguishable privacy-preserving data aggregation in wireless sensor networks," in *2011 Proceedings IEEE INFOCOM*. IEEE, 2011, pp. 2024–2032.

[15] Q. Li and G. Cao, "Efficient and privacy-preserving data aggregation in mobile sensing," in *2012 20th IEEE International Conference on Network Protocols (ICNP)*. IEEE, 2012, pp. 1–10.

[16] J. A. Naranjo, L. G. Casado, and M. Jelasity, "Asynchronous privacy-preserving iterative computation on peer-to-peer networks," *Computing*, vol. 94, no. 8-10, pp. 763–782, 2012.

[17] M. Xue, P. Papadimitriou, C. Raïssi, P. Kalnis, and H. K. Pung, "Distributed privacy preserving data collection," in *International Conference on Database Systems for Advanced Applications*. Springer, 2011, pp. 93–107.

[18] L. Hu and D. Evans, "Secure aggregation for wireless networks," in *2003 Symposium on Applications and the Internet Workshops, 2003. Proceedings.* IEEE, 2003, pp. 384–391.

[19] S. Roy, M. Conti, S. Setia, and S. Jajodia, "Securely computing an approximate median in wireless sensor networks," in *Proceedings of the 4th international conference on Security and privacy in communication netowrks*, 2008, pp. 1–10.

[20] ——, "Secure data aggregation in wireless sensor networks," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 3, pp. 1040–1052, 2012.

[21] D. Wagner, "Resilient aggregation in sensor networks," in *Proceedings of the 2nd ACM workshop on Security of ad hoc and sensor networks*, 2004, pp. 78–87.

[22] M. Greenwald and S. Khanna, "Space-efficient online computation of quantile summaries," in *ACM SIGMOD'01*, Santa Barbara, CA, 2001, p. 5866.

[23] N. Shrivastava, C. Buragohain, D. Agrawal, and S. Suri, "Medians and beyond: new aggregation techniques for sensor networks," in *Proceedings of the 2nd international conference on Embedded networked sensor systems*, 2004, pp. 239–249.

[24] M. B. Greenwald and S. Khanna, "Power-conserving computation of order-statistics over sensor networks," in *Proceedings of the twenty-third ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, 2004, pp. 275–285.

[25] B. Haeupler, J. Mohapatra, and H.-H. Su, "Optimal gossip algorithms for exact and approximate quantile computations," in *Proceedings of the 2018 ACM Symposium on Principles of Distributed Computing*, 2018, pp. 179–188.

[26] D. P. Agrawal, "Applications of sensor networks," in *Embedded Sensor Systems*. Springer, 2017, pp. 35–63.

[27] S. Madden, M. J. Franklin, J. M. Hellerstein, and W. Hong, "Tag: A tiny aggregation service for ad-hoc sensor networks," *ACM SIGOPS Operating Systems Review*, vol. 36, no. SI, pp. 131–146, 2002.

[28] J. Zhao, R. Govindan, and D. Estrin, "Computing aggregates for monitoring wireless sensor networks," in *Proceedings of the First IEEE International Workshop on Sensor Network Protocols and Applications, 2003.* IEEE, 2003, pp. 139–148.

[29] S. Nath, P. B. Gibbons, S. Seshan, and Z. Anderson, "Synopsis diffusion for robust aggregation in sensor networks," *ACM Transactions on Sensor Networks (TOSN)*, vol. 4, no. 2, pp. 1–40, 2008.

[30] P. Haghani, P. Papadimitratos, M. Poturalski, K. Aberer, and J.-P. Hubaux, "Efficient and robust secure aggregation for sensor networks," in *2007 3rd IEEE Workshop on Secure Network Protocols*. IEEE, 2007, pp. 1–6.

[31] G. Taban and V. D. Gligor, "Efficient handling of adversary attacks in aggregation applications," in *European Symposium on Research in Computer Security*. Springer, 2008, pp. 66–81.

[32] X. Xu, Q. Wang, J. Cao, P.-J. Wan, K. Ren, and Y. Chen, "Locating malicious nodes for data aggregation in wireless networks," in *2012 Proceedings IEEE INFOCOM*. IEEE, 2012, pp. 3056–3060.

[33] S. Choi, G. Ghiniţă, and E. Bertino, "Secure sensor network sum aggregation with detection of malicious nodes," in *37th Annual IEEE Conference on Local Computer Networks*. IEEE, 2012, pp. 19–27.

[34] H. Li, K. Li, W. Qu, and I. Stojmenovic, "Secure and energy-efficient data aggregation with malicious aggregator identification in wireless sensor networks," *Future Generation Computer Systems*, vol. 37, pp. 108–116, 2014.

[35] M. Li, I. Koutsopoulos, and R. Poovendran, "Optimal jamming attacks and network defense policies in wireless sensor networks," in *IEEE INFOCOM 2007-26th*

IEEE International Conference on Computer Communications. IEEE, 2007, pp. 1307–1315.

[36] R. Zhang, Y. Zhang, and K. Ren, "Dp^{2}ac: Distributed privacy-preserving access control in sensor networks," in *IEEE INFOCOM 2009*. IEEE, 2009, pp. 1251–1259.

[37] R. Zhang, J. Shi, and Y. Zhang, "Secure multidimensional range queries in sensor networks," in *Proceedings of the tenth ACM international symposium on Mobile ad hoc networking and computing*, 2009, pp. 197–206.

[38] R. Zhang and Y. Zhang, "Lr-seluge: Loss-resilient and secure code dissemination in wireless sensor networks," in *2011 31st International Conference on Distributed Computing Systems*. IEEE, 2011, pp. 497–506.

[39] R. Zhang, J. Shi, Y. Zhang, and J. Sun, "Secure cooperative data storage and query processing in unattended tiered sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 30, no. 2, pp. 433–441, 2012.

[40] R. Zhang, J. Shi, Y. Zhang, and X. Huang, "Secure top-k query processing in unattended tiered sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 63, no. 9, pp. 4681–4693, 2014.

[41] D. Mosk-Aoyama and D. Shah, "Computing separable functions via gossip," in *Proceedings of the twenty-fifth annual ACM symposium on Principles of distributed computing*, 2006, pp. 113–122.

[42] N. Khalil, M. R. Abid, D. Benhaddou, and M. Gerndt, "Wireless sensors networks for internet of things," in *IEEE ISSNIP'14*, 2014, pp. 1–6.

[43] Y.-W. Kuo, C.-L. Li, J.-H. Jhang, and S. Lin, "Design of a wireless sensor network-based iot platform for wide area and heterogeneous applications," *IEEE Sensors Journal*, vol. 18, no. 12, pp. 5187–5197, 2018.

[44] R. Rajagopalan and P. K. Varshney, "Data-aggregation techniques in sensor networks: A survey," *IEEE Communications Surveys & Tutorials*, vol. 8, no. 4, pp. 48–63, 2006.

[45] S. Roy, S. Setia, and S. Jajodia, "Attack-resilient hierarchical data aggregation in sensor networks," in *Proceedings of the fourth ACM workshop on Security of ad hoc and sensor networks*, 2006, pp. 71–82.

[46] A. Aseeri and R. Zhang, "Secure data aggregation in wireless sensor networks: Enumeration attack and countermeasure," in *IEEE ICC'19*, 2019, pp. 1–7.

[47] D. Liu and P. Ning, "Establishing pairwise keys in distributed sensor networks," in *ACM CCS*, Washington, DC, October 2003, pp. 52–61.

[48] W. Zhang, M. Tran, S. Zhu, and G. Cao, "A compromise-resilient scheme for pairwise key establishment in dynamic sensor networks," in *ACM MobiHoc*, Montreal, Canada, September 2007, pp. 90–99.

[49] A. Perrig, R. Szewczyk, V. Wen, D. Culler, and J. D. Tygar, "SPINS: Security protocols for sensor networks," in *MobiCom*, Rome, Italy, July 2001, pp. 189–199.

[50] R. Bassily and A. Smith, "Local, private, efficient protocols for succinct histograms," in *Proceedings of the forty-seventh annual ACM symposium on Theory of computing*, 2015, pp. 127–135.

[51] G. Cormode, T. Kulkarni, and D. Srivastava, "Marginal release under local differential privacy," in *Proceedings of the 2018 International Conference on Management of Data*, 2018, pp. 131–146.

[52] P. Kairouz, K. Bonawitz, and D. Ramage, "Discrete distribution estimation under local privacy," in *International Conference on Machine Learning*. PMLR, 2016, pp. 2436–2444.

[53] T. Wang, J. Blocki, N. Li, and S. Jha, "Locally differentially private protocols for frequency estimation," in *26th {USENIX} Security Symposium ({USENIX} Security 17)*, 2017, pp. 729–745.

[54] R. Bassily, K. Nissim, U. Stemmer, and A. Thakurta, "Practical locally private heavy hitters," *arXiv preprint arXiv:1707.04982*, 2017.

[55] M. Bun, J. Nelson, and U. Stemmer, "Heavy hitters and the structure of local privacy," *ACM Transactions on Algorithms (TALG)*, vol. 15, no. 4, pp. 1–40, 2019.

[56] Z. Qin, Y. Yang, T. Yu, I. Khalil, X. Xiao, and K. Ren, "Heavy hitter estimation over set-valued data with local differential privacy," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, 2016, pp. 192–203.

[57] T. Wang, N. Li, and S. Jha, "Locally differentially private heavy hitter identification," *IEEE Transactions on Dependable and Secure Computing*, 2019.

[58] N. Wang, X. Xiao, Y. Yang, J. Zhao, S. C. Hui, H. Shin, J. Shin, and G. Yu, "Collecting and analyzing multidimensional data with local differential privacy," in *2019 IEEE 35th International Conference on Data Engineering (ICDE)*. IEEE, 2019, pp. 638–649.

[59] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," in *Theory of cryptography conference*. Springer, 2006, pp. 265–284.

[60] T. T. Nguyên, X. Xiao, Y. Yang, S. C. Hui, H. Shin, and J. Shin, "Collecting and analyzing data from smart device users with local differential privacy," *arXiv preprint arXiv:1606.05053*, 2016.

[61] J. C. Duchi, M. I. Jordan, and M. J. Wainwright, "Minimax optimal procedures for locally private estimation," *Journal of the American Statistical Association*, vol. 113, no. 521, pp. 182–201, 2018.

[62] ——, "Local privacy and minimax bounds: Sharp rates for probability estimation," *arXiv preprint arXiv:1305.6000*, 2013.

[63] T. Murakami, H. Hino, and J. Sakuma, "Toward distribution estimation under local differential privacy with small samples," *Proceedings on Privacy Enhancing Technologies*, vol. 2018, no. 3, pp. 84–104, 2018.

[64] A. Pastore and M. Gastpar, "Locally differentially-private distribution estimation," in *2016 IEEE International Symposium on Information Theory (ISIT)*. Ieee, 2016, pp. 2694–2698.

[65] M. Ye and A. Barg, "Optimal schemes for discrete distribution estimation under locally differential privacy," *IEEE Transactions on Information Theory*, vol. 64, no. 8, pp. 5662–5676, 2018.

[66] C. Castelluccia, E. Mykletun, and G. Tsudik, "Efficient aggregation of encrypted data in wireless sensor networks," in *The second annual international conference on mobile and ubiquitous systems: networking and services*. IEEE, 2005, pp. 109–117.

[67] C. Castelluccia, A. C. Chan, E. Mykletun, and G. Tsudik, "Efficient and provably secure aggregation of encrypted data in wireless sensor networks," *ACM Transactions on Sensor Networks (TOSN)*, vol. 5, no. 3, pp. 1–36, 2009.

[68] D. Westhoff, J. Girao, and M. Acharya, "Concealed data aggregation for reverse multicast traffic in sensor networks: Encryption, key distribution, and routing adaptation," *IEEE Transactions on mobile computing*, vol. 5, no. 10, pp. 1417–1431, 2006.

[69] Y. Lindell and B. Pinkas, "Secure multiparty computation for privacy-preserving data mining," 2008.

[70] C. Dwork, "Differential privacy," ser. ICALP'06. Berlin, Heidelberg: Springer-Verlag, 2006, p. 1–12.

[71] M. Yang, T. Zhu, B. Liu, Y. Xiang, and W. Zhou, "Machine learning differential privacy with multifunctional aggregation in a fog computing architecture," *IEEE Access*, vol. 6, pp. 17 119–17 129, 2018.

[72] K. Zhang, Q. Han, Z. Cai, and G. Yin, "Rippas: a ring-based privacy-preserving aggregation scheme in wireless sensor networks," *Sensors*, vol. 17, no. 2, p. 300, 2017.

[73] G. Ács and C. Castelluccia, "I have a dream!(differentially private smart metering)," in *International Workshop on Information Hiding*. Springer, 2011, pp. 118–132.

[74] J. Acharya, Z. Sun, and H. Zhang, "Hadamard response: Estimating distributions privately, efficiently, and with little communication," in *The 22nd International Conference on Artificial Intelligence and Statistics*. PMLR, 2019, pp. 1120–1129.

[75] B. Ding, J. Kulkarni, and S. Yekhanin, "Collecting telemetry data privately," *arXiv preprint arXiv:1712.01524*, 2017.

[76] Ú. Erlingsson, V. Pihur, and A. Korolova, "Rappor: Randomized aggregatable privacy-preserving ordinal response," in *Proceedings of the 2014 ACM SIGSAC conference on computer and communications security*, 2014, pp. 1054–1067.

[77] D. Qin and Z. Zhang, "A frequency estimation algorithm under local differential privacy," in *2021 15th International Conference on Ubiquitous Information Management and Communication (IMCOM)*. IEEE, 2021, pp. 1–5.

[78] N. Wang, X. Xiao, Y. Yang, T. D. Hoang, H. Shin, J. Shin, and G. Yu, "Privtrie: Effective frequent term discovery under local differential privacy," in *2018 IEEE 34th International Conference on Data Engineering (ICDE)*. IEEE, 2018, pp. 821–832.

[79] R. Bassily, K. Nissim, U. Stemmer, and A. Thakurta, "Practical locally private heavy hitters," *arXiv preprint arXiv:1707.04982*, 2017.

[80] T. Wang, N. Li, and S. Jha, "Locally differentially private frequent itemset mining," in *2018 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2018, pp. 127–143.

[81] S. Afrose, T. Hashem, and M. E. Ali, "Frequent itemsets mining with a guaranteed local differential privacy in small datasets," in *33rd International Conference on Scientific and Statistical Database Management*, 2021, pp. 232–236.

[82] R. Chen, H. Li, A. K. Qin, S. P. Kasiviswanathan, and H. Jin, "Private spatial data aggregation in the local setting," in *2016 IEEE 32nd International Conference on Data Engineering (ICDE)*. IEEE, 2016, pp. 289–300.

[83] T. Kulkarni, "Answering range queries under local differential privacy," in *Proceedings of the 2019 International Conference on Management of Data*, 2019, pp. 1832–1834.

[84] G. Fanti, V. Pihur, and Ú. Erlingsson, "Building a rappor with the unknown: Privacy-preserving learning of associations and data dictionaries," *arXiv preprint arXiv:1503.01214*, 2015.

[85] S. Wang, L. Huang, Y. Nie, P. Wang, H. Xu, and W. Yang, "Privset: Set-valued data analyses with locale differential privacy," in *IEEE INFOCOM 2018-IEEE Conference on Computer Communications*. IEEE, 2018, pp. 1088–1096.

Appendix A

PERMISSIONS

Internal or personal use of IEEE/ACM copyrighted materials involved in this dissertation is permitted.

Chapter 2 is based on paper:

©IEEE. Reprint, with permission, from Aishah Aseeri, and Rui Zhang "Secure data aggregation in wireless sensor networks: Enumeration attack and countermeasure." In ICC 2019-2019 IEEE International Conference on Communications (ICC), pp. 1-7. IEEE, 2019.

Chapter 3 is based on paper:

©IEEE. Reprint, with permission, from Aishah Aseeri, and Rui Zhang "SecQSA: Secure Sampling-Based Quantile Summary Aggregation in Wireless Sensor Networks." In MSN 2021-2021 IEEE International Conference on Mobility, Sensing and Networking (MSN), pp. . IEEE, 2021.