This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2010 proceedings
This paper was presented as part of the main Technical Program at IEEE INFOCOM 2010.

Verifiable Fine-Grained Top-k Queries in Tiered Sensor Networks

Rui Zhang, Jing Shi, Yunzhong Liu, and Yanchao Zhang
Department of Electrical and Computer Engineering
New Jersey Institute of Technology
Email: {rz23, js39, yl92, yczhang}@njit.edu

***Abstract*—Most large-scale sensor networks are expected to follow a two-tier architecture with resource-poor sensor nodes at the lower tier and resource-rich master nodes at the upper tier. Master nodes collect data from sensor nodes and then answer the queries from the network owner on their behalf. In hostile environments, master nodes may be compromised by the adversary and then instructed to return fake and/or incomplete data in response to data queries. Such application-level attacks are more harmful and difficult to detect than blind DoS attacks on network communications, especially when the query results are the basis for making critical decisions such as military actions. This paper presents three schemes whereby the network owner can verify the authenticity and completeness of fine-grained top-k query results in tired sensor networks, which is the first work of its kind. The proposed schemes are built upon symmetric cryptographic primitives and force compromised master nodes to return both authentic and complete top-k query results to avoid being caught. Detailed theoretical and quantitative results confirm the high efficacy and efficiency of the proposed schemes.**

I. Introduction

We consider a two-tier sensor network as shown in Fig. 1, which consists of plenty of resource-poor sensor nodes at the lower tier and relatively fewer resource-rich *master* nodes at the upper tier. Sensor nodes perform sensing task and periodically submit sensed data to nearby master nodes for storage, while master nodes answer ad-hoc data queries from the network owner which are issued via an on-demand wireless link to some master nodes. Such in-network data storage and query processing is a must [1]–[5] in remote and extreme environments, where it is prohibitive or infeasible to maintain a high-speed always-on connection bridging the sensor network to the external network owner. This two-tier architecture is also known to be indispensable for increasing network capacity and scalability, reducing system complexity, and prolonging network lifetime [6], [7].

The reliance on master nodes for data storage and query processing raises serious security concerns in hostile environments. In particular, master nodes in military or homeland security applications might be compromised by the adversary; those commercial sensor networks may likewise be compromised by malicious business competitors to degrade their quality of data service. The adversary can instruct a compromised master node to return fake and/or incomplete data in response to ad-hoc data queries from the network owner. Such application-level attacks are more subtle and harmful than blind Denial-of-Service (DoS) attacks, especially when query results are the basis for making critical military or business decisions. This situation necessitates proactive mechanisms for verifiable queries, by which the network owner can verify the *authenticity* and *completeness* of query responses. Authentication check is needed to detect fake data in query responses, while completeness verification is necessary to make sure that no qualifying data are maliciously omitted by the master node [3]–[5].

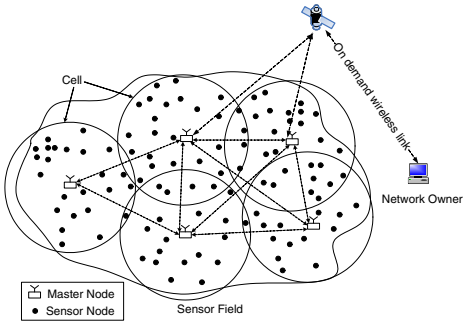


Fig. 1. A remote two-tier sensor network.

Verifiable query processing in tiered sensor networks has received attention only recently. The schemes in [3]–[5] aim at verifiable range queries asking for data within one [3], [4] or multiple [5] attributes falling into specified ranges. It remains an open challenge to realize the verifiability of other important types of data queries commonly seen in sensor networks, e.g., top-k queries [8]. A top-k query asks for data items whose numeric attributes are among the k highest, where k is an application-dependent parameter. An example is "Return the data whose temperature attribute is among the 10 highest between 2pm and 3pm." Top-k queries are useful in practice [8] and deserve attention.

Providing verifiable authenticity and completeness for top-k queries in tiered sensor networks is a challenging task. First, answering any top-k query requires global information of all the data generated in the *query region*, i.e., the region of interest, while the only entity with access to such information is the master node which might have been compromised. Second, the query region in a top-k query may range from a single node, some nodes, some parts of the network, to the entire network and cannot be predicted during network deployment. It is thus infeasible to predict which sensor nodes

978-1-4244-5837-0/10/$26.00 ©2010 IEEE

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2010 proceedings
This paper was presented as part of the main Technical Program at IEEE INFOCOM 2010.

may have data satisfying a top-k query to pre-configure them for verifiable top-k queries.

This paper investigates verifiable fine-grained top-k queries in tiered sensor networks for the first time in literature. Our main contribution is three schemes whereby the network owner can verify the authenticity and completeness of any top-k query result with overwhelming probability. All our schemes share the same basic idea that sensor nodes embed some chaining relationships among the data items they generated so that master nodes injecting fake data can be easily detected. They, however, differ in how to realize completeness verification. In particular, our first scheme requires master nodes to return some additional data items besides k qualified ones. In the second scheme, sensor nodes exchange some information about their data which is then embedded into their respective data submitted to master nodes. In doing so, the additional information returned by master nodes can be much reduced. Our final scheme is a hybrid of the first two and aims at striking a good balance between the communication overhead inside the sensor network and that incurred by ad-hoc top-k queries issued via an often costly, low-rate, on-demand wireless link (e.g., a satellite link) to some master node(s).

The efficacy and efficiency of our schemes are validated by detailed theoretical and quantitative results. With them in place, compromised master nodes are forced to return both authentic and complete top-k query results to avoid being caught. Built upon symmetric cryptographic primitives, our schemes are very suitable for resource-constrained sensor networks.

II. Network, Query, and Adversary Models

A. Network Model

We assume a similar network model as in [3]–[5]. The network is partitioned into many *cell*s, each consisting of many sensor nodes and one master node. We assume that master and sensor nodes know their respective locations and also affiliated cells. The localization requirement is fundamental in most sensor network applications and can be satisfied by many existing techniques such as [10].

Master and sensor nodes differ significantly in their resources. In particular, master nodes have abundant resources in storage, energy (e.g., a heavy-duty battery or solar panel), and computation, while sensor nodes are much more constrained in every regard. In addition, each master node can communicate with neighboring master nodes via relatively long-range, high-rate radios, thus forming an upper-tier multi-hop network.

As in [3]–[5], we assume that time is divided into *epochs*. At the end of each epoch, each sensor node submits to its affiliated master node all the data (if any) it generated during that epoch. We assume that there is no stable communication link connecting the sensor network to the external network owner, so data must be stored at master nodes. The network owner can issue top-k queries via an on-demand wireless (e.g., satellite) link to some master node(s), which is often costly and relatively low-rate.

B. Fine-Grained Top-k Queries

Data generated by sensor nodes may have multiple attributes, each corresponding to one type of sensors or one aspect of a detected event. Without loss of generality, we assume that each data item can be scored by some scoring functions [9] and ranked based on its score. In this paper, we focus on top-k queries with a single score function. For sake of simplicity, the following *primitive* top-k queries will be considered.

$$(\text{cell } = \mathcal{C}) \wedge (\text{epoch } = t) \wedge (\text{num } = k) \wedge (\text{query region} = \mathcal{I}_t) \ ,$$

where $\mathcal{C}$ and t are the interested cell ID and epoch number, respectively, k is called the *query index* referring to the number of desired data items, and $\mathcal{I}_t$ denotes the set of sensor node IDs which define a query region. Our assumption here is that the network owner knows the mappings between sensor node IDs and their respective geographic locations. We aim to support fine-grained top-k queries, in which $\mathcal{I}_t$ may cover one or more random sensor nodes in cell $\mathcal{C}$. Other more complicated queries that involve multiple cells, epochs, query indexes, and query regions can be easily decomposed into multiple primitive ones.

C. Adversary Model

We assume that some compromised master nodes are instructed to return fake and/or incomplete data in response to ad-hoc top-k queries from the network owner. The adversary may also compromise some sensor nodes to help compromised master nodes escape detection. As a conventional assumption, however, non-compromised sensor nodes are always the majority; there is otherwise no workable solution. Different from [3]–[5], we do not intend to ensure data confidentiality against master nodes. Many sensor network applications do not require data confidentiality but only query-result authenticity and completeness. For example, intrusion events in a sensor network for battlefield reconnaissance are known to the adversary and thus need not be secret. In other words, the adversary knows that he has been detected, but he can instruct compromised master nodes to return fake and/or incomplete query responses so that the network owner cannot precisely determine his itinerary. In such cases, enabling query-result authenticity and completeness verifications becomes a must.

Since each master node is in charge of a unique cell, the adversary will not gain more from the collaboration of multiple compromised master nodes. Without loss of generality, our subsequent discussion thus focuses on a cell $\mathcal{C}$ consisting of a compromised master node $\mathcal{M}$ and N sensor nodes $\{S_i\}_{i=1}^{N}$ whose IDs compose a set $\mathcal{I} = \{1, 2, \cdots, N\}$.

III. Problem Statement

In this section, we formulate the problem and introduce the evaluation metrics we will use throughout.

Assume that during each epoch t, each node $S_i \in \{S_i\}_{i=1}^{N}$ generates μ_i data items, denoted by $\mathcal{D}_i = \{D_{i,j}\}_{j=1}^{\mu_i}$. We will denote by $d_{i,j}$ the score of $D_{i,j}$, i.e., $d_{i,j} = f(D_{i,j})$, where $f(\cdot)$ is a public scoring function [9]. In addition, we will equate $D_{i_1,j_1} \leq D_{i_2,j_2}$ with $d_{i_1,j_1} \leq d_{i_2,j_2}$ for any i_1, i_2, j_1, and j_2.

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2010 proceedings
This paper was presented as part of the main Technical Program at IEEE INFOCOM 2010.

For simplicity, we subsequently assume that all the data items generated in cell $\mathcal{C}$ during epoch t have mutually different scores, which is reasonable if we take node ID and the time of data generating into consideration. This assumption implies that a *unique* correct response exists for any top-k query. The master node $\mathcal{M}$ thus receives $\mu = \sum_{i=1}^{N} \mu_i$ data items at the end of epoch t, which are denoted by $\mathcal{D} = \bigcup_{i=1}^{N} \mathcal{D}_i$.

Given a query $\mathcal{Q}_t = \langle \mathcal{C}, t, k, \mathcal{I}_t \rangle$, we define the corresponding *candidate set* as $\mathcal{D}_t = \bigcup_{i \in \mathcal{I}_t} \mathcal{D}_i$, which contains $\mu_t = \sum_{i \in \mathcal{I}_t} \mu_i$ *candidate items*. It is possible that $\mu_t < k$ in some cases. This, however, has very little impact on our schemes. For simplicity, we hereafter assume $\mu_t \geq k$ in most descriptions and will point out the additional actions that need be taken for $\mu_t < k$ when appropriate.

Assuming that $\mathcal{M}$ returns a query response $\mathcal{R}_t = \{R_i\}_{i=1}^{k}$, the problem of interest is how the network owner can verify whether $\mathcal{R}_t$ satisfies the following two conditions.

- **Authenticity**: All data items in $\mathcal{R}_t$ are indeed generated by nodes in the query region, or equivalently $\mathcal{R}_t \subseteq \mathcal{D}_t$.
- **Completeness**: $\mathcal{R}_t$ indeed contains the top k data items among all the candidates, or equivalently $R_i > D_j, \forall R_i \in \mathcal{R}_t, \forall D_j \in \mathcal{D}_t \setminus \mathcal{R}_t$.

We will use the following performance metrics to evaluate our proposed schemes.

- P_{det}**–detection probability**: the probability that a forged and/or incomplete top-k query response is detected.
- C_{T}**–in-cell communication cost**: the total additional communication energy consumption in bits incurred by enabling verifiable top-k queries in cell C per epoch. Here we assume the same energy consumption in transmitting and receiving every bit across each hop.
- C_{V}**–query communication cost**: the total additional information in bits transmitted between $\mathcal{M}$ and the network owner for enabling verifiable top-k queries. The route connecting $\mathcal{M}$ to the network owner may traverse multiple master nodes and the on-demand wireless link. For simplicity, we consider this route a virtual hop associated with an energy cost of transmitting and receiving every bit, which is different from that between neighboring sensor nodes.

IV. Verifiable Top-k Queries

In this section, we propose three schemes for verifiable top-k queries. For clarity, we temporarily ignore compromised sensor nodes and will discuss their impact in Section V.

A. Scheme 1: Verification with Additional Evidence

Scheme 1 enables authenticity verification by creating a chaining relationship by binding ordered adjacent data items with keyed hash function, and completeness verifications by requiring the master node $\mathcal{M}$ to return some data items in addition to the query result. In particular, each node S_i need sort its data items into an ordered list such that $D_{i,j} > D_{i,j+1}$, $\forall j \in [1, \mu_i)$. We then have the following observation for a given top-k query $\mathcal{Q}_t = \langle \mathcal{C}, t, k, \mathcal{I}_t \rangle$.

OBSERVATION *1: If $D_{i,j}$ satisfies $\mathcal{Q}_t$, so does $D_{i,x}, x \in [1, j)$; likewise, if $D_{i,j}$ does not satisfy $\mathcal{Q}_t$, neither does $D_{i,y}, y \in (j, \mu_i]$.*

This observation implies that adjacent data items are very likely to satisfy or dissatisfy a top-k query at the same time. Inspired by this, we assume that each S_i is preloaded with a distinct key K_i with the network owner, which can be realized by many existing techniques. K_i can be epoch-based as in [3]–[5] to achieve forward-secure authenticity. For simplicity, we ignore this issue here and refer readers to [3]–[5] for more details. We also introduce an extremely small public value $\underline{\chi}$ and an extremely large public value $\overline{\chi}$, both out of the known domain of the data score.

*1) **Data Submission**:* Let $h_\star(\cdot)$ denote a good message authentication code (MAC) keyed with the subscript. At the end of epoch t, each S_i submits the following message to $\mathcal{M}$.

- If $\mu_i \geq 1$, the message is

$$\begin{aligned} S_i \rightarrow \mathcal{M} : i, t, &\langle D_{i,1}, h_{K_i}(\overline{\chi} || D_{i,1} || D_{i,2}) \rangle, \\ &\vdots \\ &\langle D_{i,\mu_{i-1}}, h_{K_i}(D_{i,\mu_i-2} || D_{i,\mu_i-1} || D_{i,\mu_i}) \rangle, \\ &\langle D_{i,\mu_i}, h_{K_i}(D_{i,\mu_i-1} || D_{i,\mu_i} || \underline{\chi}) \rangle. \end{aligned}$$

- If $\mu_i = 0$, the message is

$$S_i \rightarrow \mathcal{M} : i, t, h_{K_i}(i || t).$$

We define the *tail* and *outlier* of $\mathcal{D}_i$ with regard to the query $\mathcal{Q}_t$ as the smallest data item satisfying $\mathcal{Q}_t$ (if any) and the largest data item not satisfying $\mathcal{Q}_t$, respectively. For instance, assume that $\mathcal{D}_i$ has γ_i data items satisfying $\mathcal{Q}_t$, which will be $\{D_{i,j}\}_{j=1}^{\gamma_i}$ according to Observation 1. The tail and outlier of $\mathcal{D}_i$ are then D_{i,γ_i} and D_{i,γ_i+1}, respectively. Note that the tail does not exist if there is no data item in $\mathcal{D}_i$ satisfying $\mathcal{Q}_t$, and the outlier does not exist if all the data items in $\mathcal{D}_i$ satisfy $\mathcal{Q}_t$.

*2) **Query Processing**:* After receiving a top-k query $\mathcal{Q}_t$, $\mathcal{M}$ first locates the largest k data items in the candidate set $\mathcal{D}_t$ and then returns them along with some additional information for completeness verification. In particular, recall that if S_i has γ_i data items satisfying $\mathcal{Q}_t$, they will be $\{D_{i,j}\}_{j=1}^{\gamma_i}$. $\mathcal{M}$ need return the following information for each $S_i, \forall i \in \mathcal{I}_t$.

- If $\gamma_i = \mu_i = 0$, the information is

$$\mathcal{M} \rightarrow owner : i, h_{K_i}(i || t).$$

- If $\gamma_i = 0$ and $\mu_i = 1$, the information is

$$\mathcal{M} \rightarrow owner : i, \langle D_{i,1}, h_{K_i}(\overline{\chi} || D_{i,1} || \underline{\chi}) \rangle.$$

- If $\gamma_i = 0$ and $\mu_i \geq 2$, the information is

$$\mathcal{M} \rightarrow owner : i, \langle D_{i,1}, D_{i,2}, h_{K_i}(\overline{\chi} || D_{i,1} || D_{i,2}) \rangle.$$

- If $0 < \gamma_i < \mu_i$, the information is

$$\begin{aligned} \mathcal{M} \rightarrow owner : i, &\langle D_{i,1}, h_{K_i}(\overline{\chi} || D_{i,1} || D_{i,2}) \rangle, \\ &\langle D_{i,2}, h_{K_i}(D_{i,1} || D_{i,2} || D_{i,3}) \rangle, \\ &\vdots \\ &\langle D_{i,\gamma_i}, h_{K_i}(D_{i,\gamma_i-1} || D_{i,\gamma_i} || D_{i,\gamma_i+1}) \rangle, \\ &D_{i,\gamma_i+1}. \end{aligned}$$

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2010 proceedings
This paper was presented as part of the main Technical Program at IEEE INFOCOM 2010.

Although the outlier D_{i,γ_i+1} does not satisfy the query, it still need be returned for recomputing the MAC $h_{K_i}(D_{i,\gamma_i-1}||D_{i,\gamma_i}||D_{i,\gamma_i+1})$.

- If $0 < \gamma_i = \mu_i$, the information is

$$\mathcal{M} \rightarrow owner : i, \langle D_{i,1}, h_{K_i}(\overline{\chi}||D_{i,1}||D_{i,2})\rangle, \langle D_{i,2}, h_{K_i}(D_{i,1}||D_{i,2}||D_{i,3})\rangle, \vdots \langle D_{i,\gamma_i}, h_{K_i}(D_{i,\gamma_i-1}||D_{i,\gamma_i}||\underline{\chi})\rangle.$$

The first three cases mean that $\mathcal{M}$ still need return some verifiable proof information for a queried node S_i (i.e., $i \in \mathcal{I}_t$) even when S_i has no data satisfying the query. Also note that only $\{D_{i,j}\}_{j=1}^{\gamma_i}$ in either of the last two cases belong to the aforementioned query result $\mathcal{R}_t = \{R_i\}_{i=1}^{k}$, where $k = \sum_{i\in\mathcal{I}_t} \gamma_i$. All the other returned information is for authenticity and completeness verifications.

*3) **Query-result Verification**:* Upon receiving the query response from $\mathcal{M}$, the network owner first locates the key for each queried node $S_i, \forall i \in \mathcal{I}_t$, with which it can verify the MACs returned along with every piece of information. If all the MAC verifications succeed, the network owner considers the query result authentic, as each key is known only to himself and the corresponding sensor node. Since the information related to node S_i follows one of the five formats, the network owner can then easily determine γ_i for each $i \in \mathcal{I}_t$ and also $\mathcal{R}_t$. The subsequent task is to check the completeness of $\mathcal{R}_t$, for which we have the following observation about tails and outliers.

OBSERVATION *2: Given a top-k query $\mathcal{Q}_t$, any outlier must be smaller than any tail, i.e., $D_{i,\gamma_i+1} < D_{j,\gamma_j}$, $i, j \in \mathcal{I}_t$.*

This observation implies that if $\mathcal{M}$ returns all the qualified k data items and all the outlier items, the network owner can perform completeness verification by checking whether the above condition holds. In particular, he can check whether the smallest tail is larger than the largest outlier. If so, the query result is considered complete and otherwise incomplete.

*4) **Performance Analysis**:*

THEOREM *1: Scheme 1 can detect any forged and/or incomplete top-k query result with probability $P_{\text{det}} = 1$ as long as none of the queried sensor nodes is compromised.*

Proof: (*Sketch*) Consider a queried S_i which has γ_i data items $\{D_{i,j}\}_{j=1}^{\gamma_i}$ satisfying the query. Since the adjacent data items are chained together with keyed MACs for which $\mathcal{M}$ does not have the corresponding key, $\mathcal{M}$ cannot insert forged data items into or omit some from $\{D_{i,j}\}_{j=1}^{\gamma_i}$ without being detected during the authenticity check.

The only option left for $\mathcal{M}$ is to replace some qualified data items from some node(s), say S_i, with other unqualified data items from some other node(s), say S_j. This means decreasing γ_i to γ_i' and increasing γ_j to γ_j'. Since the data items of both S_i and S_j are ordered, the outlier of S_i will increase from D_{i,γ_i+1} to $D_{i,\gamma_i'+1}$, while the tail of S_j will decrease from D_{j,γ_j} to $D_{j,\gamma_j'}$. Obviously, we have $D_{j,\gamma_j'} < D_{i,\gamma_i+1} < D_{i,\gamma_i'+1}$, which contradicts Observation 2 and will fail the completeness check. Any incomplete query result can thus be detected. ■

We now derive the in-cell communication cost C_{T} incurred by Scheme 1. The cost for transmitting data items and qualified node IDs are not considered because they have to be submitted even without our scheme. Assume that each node ID is of l_{id} bits, $h_\star(\cdot)$ is of l_m bits, and the average number of hops between a sensor node and $\mathcal{M}$ is $\bar{L}$. We then have

$$C_{\text{T}} = \sum_{i=1}^{N} \mu_i l_m \bar{L} = \mu l_m \bar{L} \ . \tag{1}$$

We then derive the query communication cost C_{V} incurred by returning MACs and outliers to the network owner. Let δ be the size of the query region, i.e., $\delta = |\mathcal{I}_t| = \delta_q + \delta_u$, where δ_q and δ_u denote the number of qualified and unqualified sensor nodes, respectively. We also denote by l_D the length of each data item. According to the five cases in query processing, the top k data items will totally incur k MAC transmissions; each qualified node will at most additionally incur the transmission of one outlier (Case 4); and each unqualified node will at most incur the transmission of one node ID, two outliers, and one MAC (Case 3). Then C_{V} can be upper-bounded by

$$C_{\text{V}} = k l_m + \delta_q \cdot l_D + \delta_u \cdot (l_{id} + 2l_D + l_m) \ . \tag{2}$$

We can estimate δ_u as follows. Assume that each top k data item is generated by each node in $\mathcal{I}_t$ with equal probability. Then each sensor node is unqualified with probability $(1 - 1/\delta)^k$, and there are thus $\delta_u \approx \delta(1-1/\delta)^k$ unqualified sensor nodes on average.

B. Scheme 2: Verification by Crosscheck

Scheme 1 works well when δ is small. However, when δ is large and k is small, C_{V} will be significant. For example, if the network owner queries for the top-2 data items generated in epoch t in cell $\mathcal{C}$, then $\delta = N$ which leads to an overwhelming C_{V}. The situation becomes even worse if the network owner issues multiple different queries regarding cell $\mathcal{C}$ and epoch t. Since the query response traverses the on-demand wireless link which is possibly costly and low-rate, it is necessary to explore other alternatives to reduce C_{V} while still enabling authenticity and completeness verifications.

Scheme 2 works by letting sensor nodes in a cell exchange some information which is then embedded in their submitted data. By doing so, the data of qualified sensor nodes can crosscheck each other such that the master node no longer needs to return any data for unqualified sensor nodes. C_{V} can thus be significantly reduced. As an example, if the network owner finds $D_{i,x}$ and $D_{i,x+1}$ in the response and also knows that node $S_j, j \in \mathcal{I}_t$, has a data item with a score in $(d_{i,x}, d_{i,x+1})$, he can ascertain that it should appear in the response as well; otherwise, the query result is considered incomplete.

*1) **Data Submission**:* In Scheme 2, we introduce a short *gossip phase* just before data submission at the end of each epoch, during which each node broadcasts its highest score (if

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2010 proceedings
This paper was presented as part of the main Technical Program at IEEE INFOCOM 2010.

any) within its cell. Consider node S_i as an example. During the gossip phase of epoch t, S_i broadcasts its highest score $d_{i,1}$ within cell $\mathcal{C}$ with probability p which is a system parameter.

$$S_i \rightarrow * : i, d_{i,1} \ .$$

Here we assume a suitable broadcast authentication protocol like multilevel μTESLA [11] for secure and reliable transmissions of such gossip messages.

S_i then sorts its own data scores $\{d_{i,j}\}_{j=1}^{\mu_i}$ and the received ones in the descending order. Recall our assumption that all the data items generated during each epoch in cell $\mathcal{C}$ have different scores. Let $\mathcal{I}_{i,1}$, $\mathcal{I}_{i,x+1}$ $(1 \leq x \leq \mu_i - 1)$, and $\mathcal{I}_{i,\mu_i+1}$ denote the sets of received node IDs with score higher than $d_{i,1}$, between $d_{i,x}$ and $d_{i,x+1}$, and smaller than d_{i,μ_i}, respectively. We call each such $\mathcal{I}_{i,j}$ a gossip ID set henceforth. Let $\{\cdot\}_\star$ denote an OCB-like authenticated encryption primitive [12] using the key on the subscript. Finally, S_i submits the following message to $\mathcal{M}$.

- If $\mu_i \geq 1$, the message is

$$\begin{aligned} S_i \rightarrow \mathcal{M} : i, t, & \langle E_{i,1}, h_{K_i}(E_{i,0}||E_{i,1}||E_{i,2})\rangle, \\ & \langle E_{i,2}, h_{K_i}(E_{i,1}||E_{i,2}||E_{i,3})\rangle, \\ & \quad \vdots \\ & \langle E_{i,\mu_i}, h_{K_i}(E_{i,\mu_i-1}||E_{i,\mu_i}||E_{i,\mu_i+1})\rangle, \\ & E_{i,\mu_i+1}, \end{aligned}$$

  where $E_{i,0} = \overline{\chi}$, $E_{i,\mu_i+1} = \mathcal{I}_{i,\mu_i+1}||\underline{\chi}$, and $E_{i,j} = \{\mathcal{I}_{i,j}\}_{K_i}||D_{i,j}, \forall j \in [1, \mu_i]$. Hereafter we will call each $E_{i,j}$ an *enhanced* data item.
- If $\mu_i = 0$, the message is

$$S_i \rightarrow \mathcal{M} : i, t, h_{K_i}(i||t).$$

Why are $\mathcal{I}_{i,j}$s encrypted? The purpose is to prevent $\mathcal{M}$ from finding out which data items can crosscheck each others. Without the encryptions, $\mathcal{M}$ might be able to selectively drop some data items without being detected.

*2) **Query Processing**:* After receiving a top-k query $\mathcal{Q}_t = \langle \mathcal{C}, t, k, \mathcal{I}_t\rangle$, $\mathcal{M}$ locates the k data items whose scores are among the k highest in $\mathcal{D}_t$ and also determines γ_i for each node S_i, $i \in \mathcal{I}_t$. For each qualified node S_i (i.e., $\gamma_i > 0$), $\mathcal{M}$ includes the following information in the query response.

$$\begin{aligned} \mathcal{M} \rightarrow owner : i, & \langle E_{i,1}, h_{K_i}(E_{i,0}||E_{i,1}||E_{i,2})\rangle, \\ & \langle E_{i,2}, h_{K_i}(E_{i,1}||E_{i,2}||E_{i,3})\rangle, \\ & \quad \vdots \\ & \langle E_{i,\gamma_i}, h_{K_i}(E_{i,\gamma_i-1}||E_{i,\gamma_i}||E_{i,\gamma_i+1})\rangle, \\ & E_{i,\gamma_i+1}, \end{aligned}$$

where E_{i,γ_i+1} is unqualified and only returned to enable authenticity check. Scheme 2 differs clearly from Scheme 1 in that no information is returned for unqualified sensor nodes.

Different from Scheme 1, special actions need be taken in Scheme 2 if $\mathcal{M}$ returns less than k data items to the network owner, e.g., because $\mu_t < k$. This may also be caused by $\mathcal{M}$ intentionally returning data items only from some nodes with less than k data items in total while claiming that other nodes in the query region generated no data during the queried epoch. As an example, for a top-10 query regarding two nodes S_1 and S_2, $\mathcal{M}$ only returns all the eight data items generated by S_1, while claiming S_2 has no data items. If S_2 did not broadcast its highest score, then no information about S_2's data will be embedded in S_1's data items. The network owner thus cannot differentiate these two cases. To defeat this attack, we require $\mathcal{M}$ to contain the following information in any response with less than k data items from a subset $\hat{\mathcal{I}}_t \subset \mathcal{I}_t$ of queried nodes.

$$\mathcal{M} \rightarrow owner : h(\quad || \quad h_{K_i}(i||t)) \ ,$$
$$\qquad\qquad\quad i \in \mathcal{I}_t \setminus \hat{\mathcal{I}}_t$$

where h is a good hash function. Such information serves as an aggregated proof that no node in $\mathcal{I}_t \setminus \hat{\mathcal{I}}_t$ generated any data during epoch t, and it can be easily verified by the network owner.

*3) **Query-result Verification**:* Once receiving the query response, the network owner verifies its authenticity and derives the query result $\mathcal{R}_t$ using the same method as in Scheme 1. If the authentication check succeeds, he proceeds to verify the completeness of $\mathcal{R}_t$ through the following steps.

The network owner first obtains the gossip ID sets $\mathcal{I}_{i,j}$s contained in the query response after doing decryptions with the corresponding keys. Then for every data item in $\mathcal{R}_t$, say $D_{i,j}$ with a non-empty set $\mathcal{I}_{i,j}$, the network owner checks whether there is at least one data item from node S_x for all $x \in \mathcal{I}_{i,j} \cap \mathcal{I}_t$. If not, the query result is considered incomplete. The underlying rationale is very simple. If $x \in \mathcal{I}_{i,j} \cap \mathcal{I}_t$, node S_x must have at least one data item whose score is higher than $d_{i,j}$ according to the definition of $\mathcal{I}_{i,j}$. If $\mathcal{R}_t$ passes the first check, the network owner continues to verify that it complies with Observation 2. If so, the query result is considered complete and incomplete otherwise.

*4) **Performance Analysis**:*

THEOREM *2: Assuming that none of the queried sensor nodes is compromised, Scheme 2 can detect any forged and/or incomplete top-k query result with probability $P_{\text{det}} \geq 1 - (1-p)^\lambda$, where $\lambda \geq 1$ is the number of qualified sensor nodes whose data items are all replaced with other unqualified ones.*

Proof: (*Sketch*) Similar to the proof of Theorem 1, it can be easily shown that Scheme 2 enables the network owner to deterministically detect any query result containing forged data items.

There are two options left for $\mathcal{M}$ to return incomplete data. First, $\mathcal{M}$ can replace some qualified data items from some node(s) with other unqualified data items from some other node(s). This can be easily detected, as explained in the proof of Theorem 1. Second, $\mathcal{M}$ may omit all the qualified data items from some qualified sensor node(s). In this case, the omitted items must have scores higher than those of unqualified data items returned by $\mathcal{M}$ to replace them. If the query response contains any omitted node ID in the encrypted form, the network owner can detect $\mathcal{M}$'s misbehavior. Assume that $\mathcal{M}$ completely omits data items from λ qualified nodes, each broadcasting its highest score with probability p. If $p = 1$,

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2010 proceedings
This paper was presented as part of the main Technical Program at IEEE INFOCOM 2010.

each of the λ IDs appears at least once in unqualified data items, whereby the network owner can precisely detect the result incompleteness as discussed in Section IV-B3. If $p < 1$, however, the result incompleteness cannot be detected if none of the λ IDs is contained in the query response. This occurs with probability $(1-p)^\lambda$ when none of the λ sensor nodes broadcasts its highest score. Summarizing the above cases, we have $P_{\rm det} \geq 1-(1-p)^\lambda$ for Scheme 2. ■

Now we derive the in-cell communication cost $C_{\rm T}$ of Scheme 2. In contrast to Scheme 1, Scheme 2 has two additional costs: C_B incurred by the gossip phase and C_{ID} incurred by transmitting embedded node IDs to $\mathcal{M}$. We assume that the simplest broadcast scheme is used, in which each node forwards a received broadcast packet once. Also assume that the N sensor nodes all broadcast their respective highest score with probability p so that we can upper-bound C_B and C_{ID}. Let l_d denote the length of a data score. We then have $C_B = pN^2(l_d + l_{id})$. Since each node on average embeds $p(N-1)$ gossiped node IDs into its data items, $C_{ID} = pN(N-1)l_{id}\bar{L}$. Adding C_B and C_{ID} to $C_{\rm T}$ in Eq. (1), we have

$$\begin{aligned} C_{\rm T} &= \mu l_m \bar{L} + C_B + C_{ID} \\ &= \mu l_m \bar{L} + pN^2(l_d + l_{id}) + pN(N-1)l_{id}\bar{L}. \end{aligned} \tag{3}$$

Finally, we estimate the query communication cost $C_{\rm V}$. For simplicity, we assume that each node $S_i, i \in \mathcal{I}$, generates $\bar{\mu}$ data items and that each gossip-ID set $\mathcal{I}_{i,j}$ contains $p(N-1)/(\bar{\mu}+1)$ node IDs on average. As in the analysis of Scheme 1, there are totally $\delta_q = \delta(1-(1-1/\delta)^k)$ qualified sensor nodes, each generating k/δ_q qualified data items on average. For each qualified data item, one MAC and a gossip ID set need be transmitted, leading to a cost of $k(l_m + p(N-1)l_{id}/(\bar{\mu}+1))$. In addition, an unqualified data item need be transmitted for each qualified node, resulting in a cost of $\delta_q(p(N-1)l_{id}/(\bar{\mu}+1) + l_D)$. To sum up, we have

$$\begin{aligned} C_{\rm V} =& k(l_m + \frac{p(N-1)l_{id}}{\bar{\mu}+1}) \\ &+ \delta(1-(1-1/\delta)^k)(\frac{p(N-1)l_{id}}{\bar{\mu}+1} + l_D)\ . \end{aligned} \tag{4}$$

C. Scheme 3: Hybrid Crosscheck

Scheme 1 incurs a lower in-cell communication $C_{\rm T}$ but a higher query communication cost $C_{\rm V}$ than Scheme 2. Now we further propose Scheme 3 to strike a balance between $C_{\rm T}$ and $C_{\rm V}$. In Scheme 3, each cell $\mathcal{C}$ is virtually partitioned into s subcells, and each sensor node knows its affiliated subcell. We denote the s subcells and their respective node ID sets by $\{\mathcal{C}_y\}_{y=1}^s$ and $\{\mathcal{J}_y\}_{y=1}^s$, respectively.

*1) **Data Submission**:* As in Scheme 2, each node S_i broadcasts its highest data value score (if any) along with its ID within its affiliated subcell during the gossip phase of each epoch t. Node S_i may receive some gossip messages during the gossip phase and will process them in the same way as in Section IV-B1. In particular, S_i will submit its own data mixed with gossiped node IDs to the master node $\mathcal{M}$.

*2) **Query Processing**:* Upon receiving a top-k query $\mathcal{Q}_t = \langle \mathcal{C}, t, k, \mathcal{I}_t \rangle$, $\mathcal{M}$ first needs to follow the procedure in Section IV-B2 to prepare a partial response which contains the top-k data items along with some auxiliary information for authenticity and completeness verifications.

Unfortunately, each node now only has information about the data items generated in its own subcell. If $\mathcal{M}$ does not return any qualified data item from one subcell, the network owner cannot differentiate whether that subcell indeed has no qualified data items or $\mathcal{M}$ omits them. In view of this situation, Scheme 3 requires $\mathcal{M}$ to return some additional information as in Scheme 1. Specifically, a subcell $\mathcal{C}_y$ is called unqualified if the following two conditions are both met.

- $\mathcal{C}_y$ contains some queried nodes, i.e., $\mathcal{J}_y \bigcap \mathcal{I}_t \neq \emptyset$.
- No node in $\mathcal{C}_y$ has data items satisfying $\mathcal{Q}_t$.

$\mathcal{M}$ is required to return the largest data items in the intersections between the query region and each unqualified subcell. Consider an unqualified subcell $\mathcal{C}_x$ as an example. There are two cases. If there is a node, say S_i, that generated the largest data item $D_{i,1}$ in epoch t among all the nodes in $\mathcal{J}_x \bigcap \mathcal{I}_t$, $\mathcal{M}$ need include the following information in the query response,

$$\mathcal{M} \rightarrow owner : i, \langle E_{i,1}, h_{K_i}(E_{i,0}||E_{i,1}||E_{i,2}) \rangle, E_{i,2}\ , \tag{5}$$

where $E_{i,j}$ is as defined in Section IV-B1. If no node in $\mathcal{J}_x \bigcap \mathcal{I}_t$ generated any data during epoch t, $\mathcal{M}$ need append the following aggregated proof to the query response,

$$\mathcal{M} \rightarrow owner : x, h(\mathop{||}_{i \in \mathcal{I}_t \bigcap \mathcal{J}_x} h_{K_i}(i||t))\ .$$

*3) **Query-result Verification**:* Upon receiving the query response, the network owner first verifies its authenticity as in Scheme 1 and Scheme 2. If successful, he constructs the top-k query result $\mathcal{R}_t$ and then verifies its completeness as in Section IV-B3. Finally, the network owner checks that the query response contains no data item from unqualified subcells whose value score is higher than the smallest one in $\mathcal{R}_t$. If so, the query result is considered complete and incomplete otherwise.

*4) **Performance Analysis**:*

THEOREM *3: Assume that none of the queried sensor nodes is compromised, Scheme 3 can detect any forged and/or incomplete top-k query result with probability* $P_{\rm det} = 1$.

Proof: (*Sketch*) Similar to the proofs of Theorems 1 and 2, it can be easily shown that Scheme 3 enable the network owner to deterministically detect any query result containing forged data items. Also note that the first completeness check corresponds to Scheme 2 with $p = 1$. If we view each unqualified subcell as a virtual sensor node, the second completeness check corresponds to Scheme 1. Both checks can succeed with probability 1. So we have $P_{\rm det} = 1$ for Scheme 3. ■

The in-cell communication cost $C_{\rm T}$ of Scheme 3 can be computed similarly as in Scheme 2. In particular, we have

$$\begin{aligned} C_{\rm T} &= \mu l_m \bar{L} + C_B + C_{ID} \\ &= \mu l_m \bar{L} + \frac{N^2(l_d + l_{id})}{s} + \frac{N(N/s-1)l_{id}\bar{L}}{s}\ . \end{aligned}$$

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2010 proceedings
This paper was presented as part of the main Technical Program at IEEE INFOCOM 2010.

To estimate the query communication cost $C_{\rm V}$, we assume that each node in cell $\mathcal{C}$ generates $\bar{\mu}$ data items during epoch t. Each $\mathcal{I}_{i,j}$ thus contains $(N/s-1)/(\bar{\mu}+1)$ node IDs on average. Recall that δ denotes the cardinality of the query region $\mathcal{I}_t$. Let $s_c \in [1, \min(s, \delta)]$ be the number of candidate subcells with nodes in $\mathcal{I}_t$. Each candidate subcell is then unqualified with probability $(1-1/s)^k$, so there are on average $s_c(1-1/s)^k$ unqualified subcells, each of which requires transmitting at most an unqualified node ID, two unqualified enhanced data items, and one MAC (see Eq. (5)). In addition, one MAC and a gossip node set need be transmitted for each of the k qualified data items. An unqualified enhanced data item need be transmitted for each of the $\delta_q = \delta(1-(1-1/\delta)^k)$ qualified sensor nodes. We then have

$$C_{\rm V} = k(l_m + \frac{(N/s-1)l_{id}}{\bar{\mu}+1}) + \delta_q(\frac{(N/s-1)l_{id}}{\bar{\mu}+1} + l_D) + s_c(1-1/s)^k(\frac{2(N/s-1)l_{id}}{\bar{\mu}+1} + l_{id} + 2l_D + l_m)\ .$$

V. Defenses against Compromised Sensor Nodes

So far we have not considered the impact of compromised sensor nodes for the sake of clarity. In practice, however, the adversary may also compromise some sensor nodes among $\{S_i\}_{i=1}^N$ to facilitate the attack, e.g., help the master node escape the detection. We assume that compromised sensor nodes will fully collaborate with the compromised $\mathcal{M}$, e.g., by disclosing their keys. Now we further consider the following issues.

Issue 1: Forge data items with extremely high scores.

The simplest attack by compromised sensor nodes on top-k queries is to forge data items with extremely high scores. In particular, compromised sensor nodes in cell $\mathcal{C}$ can submit many arbitrarily fake large data items to $\mathcal{M}$, which are properly authenticated and chained together using MACs. If any compromised node appears in the query region, data from non-compromised sensor nodes will have little chance to appear in the query result. Such fake query results are not easy to detect by the network owner.

We now briefly evaluate the impact of such attacks. Assume that the adversary has compromised c out of N sensor nodes, which are uniformly distributed in cell $\mathcal{C}$. For a given top-k query $\mathcal{Q}_t = \langle \mathcal{C}, t, k, \mathcal{I}_t \rangle$ with $|\mathcal{I}_t| = \delta$, the probability that none of the compromised sensor nodes is queried is given by $\prod_{i=1}^{c}(1 - \min\{1, \frac{\delta}{N-i+1}\})$. Therefore, at least one compromised node will be included in the query region $\mathcal{I}_t$ with probability $1 - \prod_{i=1}^{c}(1 - \min\{1, \frac{\delta}{N-i+1}\})$.

One countermeasure against Issue 1 is to use smaller query regions so as to reduce the possibility of querying compromised sensor nodes. In addition, if a small set of sensor nodes always have data satisfying many diverse top-k queries, the network owner may suspect those nodes as being compromised and then use some tool like software attestation [13] to testify his hypotheses and remove those compromised. Also note that such compromised yet undetected sensor nodes are open challenges for almost all security mechanisms, as they can misbehave arbitrarily.

TABLE I
Default Evaluation Parameters

Para.	Val.	Para.	Val.	Para.	Val.	Para.	Val.
N	400	k	10	δ	100	$\bar{\mu}$	10
p	0.9	L	8	s	10	l_{id}	10
l_m	160	l_D	400	l_d	20	λ	1
c	10	s_c	2				

Issue 2: Do not cooperate with non-compromised nodes.

To avoid becoming suspicious, compromised sensor nodes may faithfully generate data but refuse to cooperate with non-compromised sensor nodes, e.g., by not embedding other sensor node IDs in Scheme 2 and Scheme 3. Such attacks, however, are much less harmful. The reason is that a qualified sensor node's ID may appear in multiple sensor nodes' data items, depending on its data items' ranks in the top-k query result. As long as its ID appears at least once, both $\mathcal{M}$'s misbehavior and the compromised sensor nodes can be detected. Therefore, this attack is less likely to occur in practice.

Issue 3: Frame legitimate master nodes.

Our previous discussion focuses on detecting a compromised master node $\mathcal{M}$, possibly assisted by some compromised sensor nodes. The adversary, however, may exploit our techniques to frame some legitimate master nodes. For example, the adversary may compromise some sensor nodes in cell $\mathcal{C}'$ with a legitimate master node $\mathcal{M}'$. The compromised sensor nodes can frame $\mathcal{M}'$ by sending its data authenticated using incorrect keys. Since $\mathcal{M}'$ does not know the correct keys, it cannot detect such misbehavior. Consequently, the network owner will falsely identify $\mathcal{M}'$ as malicious. An effective countermeasure against the framing attack is to let each sensor node and master node digitally sign every message transmitted and received. In case of dispute, the network owner can detect the misbehaving entities by analyzing related messages and signatures. This solution requires public-key operations which, however, have been shown to be quite viable in WSNs [14].

VI. Performance Evaluation

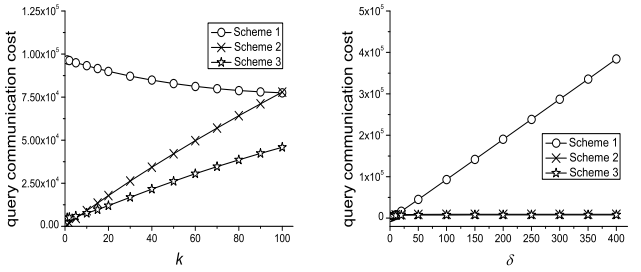
In this section, we use numerical results to evaluate the performance of the proposed schemes for a given top-k query $\mathcal{Q}_t = \langle \mathcal{C}, k, t, \mathcal{I}_t \rangle$ with $|\mathcal{I}_t| = \delta$. We assume a cell with 400 sensor nodes and a master node, and the average distance between a sensor node and the master node is 8 hops. We also assume error-free and collision-free packet transmissions. Table I summarizes the default evaluation parameters unless specified otherwise.

A. Evaluation Results

1) Impact of k and δ: We first examine the impact of k and δ on $C_{\rm V}$. According to our previous analysis, $P_{\rm det}$ and $C_{\rm T}$ are both independent of k and δ.

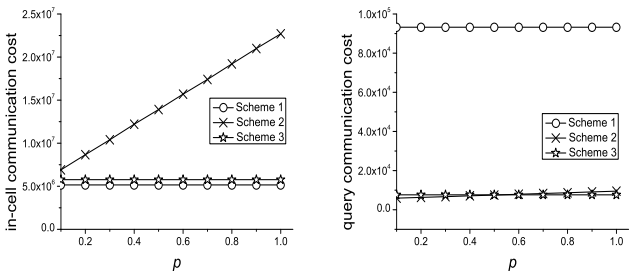
Fig.2(a) shows that Scheme 1 has the highest $C_{\rm V}$, followed by Scheme 2 and then Scheme 3, as too much information need be transmitted for unqualified sensor nodes. In addition, the $C_{\rm V}$ of Scheme 1 decreases as k increases. This is anticipated because a larger k implies more qualified sensor nodes

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2010 proceedings
This paper was presented as part of the main Technical Program at IEEE INFOCOM 2010.



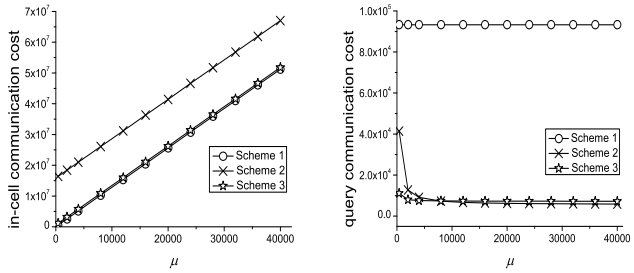
(a) C_{V} vs. k (b) C_{V} vs. δ

Fig. 2. Impact of query parameters k and δ on C_{V}.



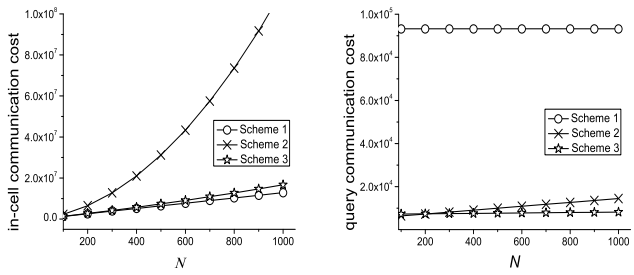
(a) C_{T} vs. p (b) C_{V} vs. p

Fig. 3. Impact of the broadcast probability p on Scheme 2.



(a) C_{T} vs. μ (b) C_{V} vs. μ

Fig. 4. Impact of μ.



(a) C_{T} vs. N (b) C_{V} vs. N

Fig. 5. Impact of N on C_{T} and C_{V}.

but fewer unqualified sensor nodes, and each unqualified node incurs two outlier transmissions, while each qualified node incurs one. In contrast, as k increases, C_{V} increases with k for both Scheme 2 and Scheme 3 and more rapidly for Scheme 2. The reason is that more sensor node IDs are returned with each qualified data item in Scheme 2 than in Scheme 3, the number of which will increase with k.

Fig. 2(b) shows that the C_{V} of Scheme 1 grows rapidly with the increase of δ, while the C_{V}s of Scheme 2 and Scheme 3 are both insensitive to δ. The reason is obvious: the larger δ, the more unqualified sensor nodes, and the more information for unqualified sensor nodes need be returned in Scheme 1. In contrast, Scheme 2 and Scheme 3 both require much less information to be sent for unqualified sensor nodes.

2) Impact of p: Fig. 3 shows the impact of the broadcast probability p on Scheme 2, where the results for Scheme 1 and Scheme 3 are just given for reference. We can see that the C_{T} of Scheme 2 increases linearly with p, as the number of gossip messages grows linearly with p. In addition, the C_{V} of Scheme 2 is relatively insensitive to p and is closer to that of Scheme 3. The reason is that p only affects the number of sensor node IDs in the query response, which are a relatively small part of C_{V}. Still, Scheme 1 has the highest C_{V}.

3) Impact of μ: Fig. 4(a) shows the impact of $\mu = N\bar{\mu}$, the number of data items generated in a cell every epoch, on C_{T}. We vary μ by changing $\bar{\mu}$. We can see that the C_{T}s of all three schemes are directly linear to μ, among which the C_{T} of Scheme 2 is the highest. Such results are of no surprise because the number of MACs submitted to the master node in all three schemes increases linearly with μ.

Fig. 4(b) shows the impact of μ on C_{V}. Despite the independence of μ, Scheme 1 has the highest C_{V}. In addition, the C_{V}s of Scheme 2 and Scheme 3 are both inversely proportional to μ. This is understandable because the larger μ is, the fewer the gossip IDs bound to each data item in both schemes, and vice versa. In addition, each node in Scheme 2 receives gossip messages from the whole cell instead of from a smaller subcell as in Scheme 3, so more gossiped node IDs are bound to and returned with each data item in Scheme 2. The C_{V} of Scheme 2 is thus higher than that of Scheme 3.

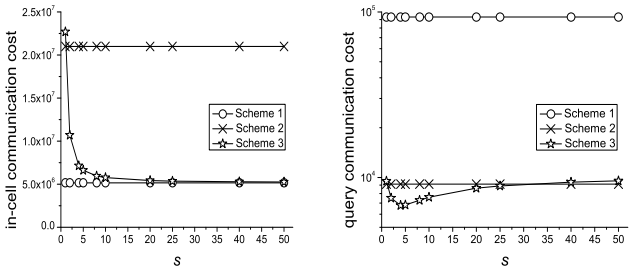
4) Impact of N: Fig. 5(a) shows the impact of the number N of sensor nodes in a cell on C_{T}. All three schemes have a C_{T} directly proportional to N because larger N means that more nodes submit data to the master node. In addition, under the default configuration, Scheme 2 has the highest C_{T}, followed by Scheme 3 and then Scheme 1. The reason is that nodes in Scheme 2 and Scheme 3 need broadcast their highest data scores to the whole cell and the subcell, respectively, leading to some overhead directly proportional to N^2.

Fig. 5(b) shows the impact of N on C_{V}. Though independent of N, Scheme 1's C_{V} is still the highest because too much information need be returned for unqualified sensor nodes. In addition, Scheme 3 has a higher C_{V} than Scheme 2, as the master node need return some proof for each unqualified subcell to enable completeness verification.

5) Impact of s: Fig. 6 shows the impact of s on the C_{T} and $C_{\mathrm{V}}$of Scheme 3, where the results of Scheme 1 and Scheme 2 are independent of s and only plotted for reference.

From Fig. 6(a), we can see that Scheme 3's C_{T} is higher than that of Scheme 2 when s is small. The reason is that

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2010 proceedings
This paper was presented as part of the main Technical Program at IEEE INFOCOM 2010.



(a) C_{T} vs. s (b) C_{V} vs. s

Fig. 6. Impact of the number s of subcells.

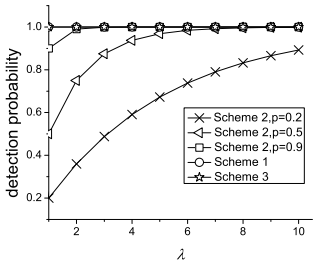


Fig. 7. Impact of λ and p on P_{det}.

in such cases each node broadcasts its highest data score with probability 1 to its subcell of size comparable to the whole cell, while each node in Scheme 2 only broadcasts its highest data value score with probability p which is set 0.9 in Fig. 6(a). As s increases, the subcell size drops, so does the C_{T} of Scheme 3 which quickly converges to that of Scheme 1. In particular, when $s = N$, Scheme 3 and Scheme 1 are equivalent because each subcell contains one sensor node.

Fig. 6(b) shows that as s increases, Scheme 3's C_{V} first decreases, then increases and exceeds that of Scheme 2 when s is larger than 30. The reason can be explained as follows. On the one hand, the larger s is, the fewer the gossip IDs bound to each data item, whose number is inverse proportional the number of subcells, leading to lower C_{V}. This effect is particularly dramatic when s is small. On the other hand, the number of unqualified subcells also increases as s increases, so more information for unqualified subcells need be returned, which results in higher C_{V}. The overall C_{V} is the interplay between these two factors. Still Scheme 1 has the highest C_{V}.

6) Impact of λ: Fig. 7 shows the detection probability $P_{\mathrm{det}} = 1-(1-p)^{\lambda}$ of Scheme 2 for various combinations of the broadcast probability p and λ which denotes the number of nodes whose qualified data items are all not returned. Here we assume that the master node is smart enough to return only authenticated data items because fake items can be detected immediately. λ and p do not affect Scheme 1 and Scheme 3 whose P_{det}s are both 1 and plotted as a reference. As p and/or λ grow large, more and more traces (the IDs of affected nodes embedded in returned data items) are available, in which case the P_{det} of Scheme 2 quickly approaches 1.

B. Summary

We summarize the evaluation results as follows.

- Scheme 1 can detect any fake and/or incomplete top-k query result with probability 1 and has minimal C_{T} but possibly high C_{V} when the query region, i.e., δ, is large.
- Scheme 2 can detect any fake and/or incomplete top-k query result with very high probability and has high C_{T} but minimal C_{V}.
- Scheme 3 can detect any fake and/or incomplete top-k query result with probability 1 and has moderate communication costs C_{T} and C_{V}.

Scheme 1 is most suitable for infrequent top-k queries with small query regions, while Scheme 2 is more preferable with frequent top-k queries with large query regions. In practice, Scheme 3 may be the best choice whose performance can be adjusted as needed. Built upon symmetric cryptographic primitives, our schemes are very suitable and practical for resource-constrained sensor networks.

ACKNOWLEDGMENT

This work was supported in part by the US National Science Foundation under grants CNS-0716302 and CNS-0844972 (CAREER). We would also like to thank anonymous reviewers for their constructive comments and helpful advice.

REFERENCES

[1] W. Zhang, H. Song, S. Zhu, and G. Cao, "Least privilege and privilege deprivation: towards tolerating mobile sink compromises in wireless sensor networks," in *ACM MobiHoc'05*, Urbana-Champaign, IL, USA, May 2005, pp. 378–389.
[2] M. Shao, S. Zhu, W. Zhang, and G. Cao, "pDCS: Security and privacy support for data-centric sensor networks," in *IEEE INFOCOM'07*, Anchorage, Alaska, USA, May 2007, pp. 1298–1306.
[3] B. Sheng and Q. Li, "Verifiable privacy-preserving range query in sensor networks," in *IEEE INFOCOM'08*, Phoenix, AZ, Apr. 2008, pp. 46–50.
[4] J. Shi, R. Zhang, and Y. Zhang, "Secure range queries in tiered sensor networks," in *IEEE INFOCOM'09*, Rio de Janeiro, Brazil, Apr. 2009.
[5] R. Zhang, J. Shi, and Y. Zhang, "Secure multidimensional range queries in sensor networks," in *ACM MobiHoc'09*, New Orleans, LA, May 2009, pp. 197–206.
[6] O. Gnawali, *et al.*, "The tenet architecture for tiered sensor networks," in *SenSys'06*, Boulder, Colorado, USA, Oct. 2006, pp. 153–166.
[7] P. Desnoyers, D. Ganesan, and P. Shenoy, "TSAR: A two tier sensor storage architecture using interval skip graphs," in *ACM SenSys'05*, San Diego, California, USA, Nov. 2005, pp. 39–50.
[8] A. S. Silberstein, R. Braynard, C. Ellis, K. Munagala, and J. Yang, "A sampling-based approach to optimizing top-k queries in sensor networks," in *ICDE '06*, Lisboa, Portuga, Apr. 2006, p. 68.
[9] G. Das, D. Gunopulos, N. Koudas, and D. Tsirogiannis, "Answering top-k queries using views," in *VLDB'06*, Sep. 2006, pp. 451–462.
[10] D. Liu, P. Ning, A. Liu, C. Wang, and W. Du, "Attack-resistant location estimation in wireless sensor networks," *ACM TISSEC*, vol. 11, no. 4, pp. 1–39, July 2008.
[11] D. Liu and P. Ning, "Multilevel μTESLA: Broadcast authentication for distributed sensor networks," *Trans. on Embedded Computing Sys.*, vol. 3, no. 4, pp. 800–836, 2004.
[12] P. Rogaway, M. Bellare, and J. Black, "OCB: A block-cipher mode of operation for efficient authenticated encryption," *ACM Trans. Inf. Syst. Secur.*, vol. 6, no. 3, pp. 365–403, Aug. 2003.
[13] A. Seshadri, A. Perrig, L. van Doorn, and P. K. Khosla, "SWATT: Software-based attestation for embedded devices," in *IEEE S&P'04*, Berkeley, CA, USA, May 2004, pp. 272–282.
[14] A. Liu and P. Ning, "TinyECC: A configurable library for elliptic curve cryptography in wireless sensor networks," in *IPSN'08*, St. Louis, MO, Apr. 2008, pp. 245–256.