Verifiable Privacy-Preserving Querying in wireless Sensor Networks

T Sravan, S.Suresh

1,2 Dept. of CSE, Sree Rama Institute of Technology & Science, SKuppenakuntla, Penuballi, Khammam, AP, India

Abstract- A large-scale wireless sensor network constructed in terms of two-tiered architecture, where cloud nodes take charge of storing sensed data and processing queries with respect to the sensing nodes and querists, incurs security breach. This is because the importance of cloud nodes makes them attractive to adversaries and raises concerns about data privacy and query result correctness. We consider a sensor network that is not fully trusted and ask the question how we preserve privacy for the collected data and how we verify the data reply from the network. We explore the problem in the context of a network augmented with storage nodes and target at range query, a very general and powerful type of query. We use bucketization to mix the data for a range, use message encryption for data integrity, and employ encoding numbers to prevent the storage nodes from dropping data.

I. Introduction

Cloud based online storage service is more and more popular not only in traditional networks but also in wireless sensor networks. Because cloud storage would involve storing data on multiple cloud nodes (CNs) (e.g., cloud servers or devices with abundant storage), data privacy and integrity preserving become important issues. We believe that pervasive computing systems, touching upon every aspect of our life, will be partially supported by the sensor network infrastructure, which is involved in two processes: monitoring the environment surrounding us (also including us), and providing information for us to analyze and respond. Both processes are exposed to potential risks for information security and privacy prohibiting the realistic sensor network deployment. On one hand, a sensor network may leak information about people to an unauthorized party, which leads to a privacy breaching. On the other hand, it may also lie about the collected data to a valid query making the network dysfunctional. In deploying such a realistic sensor network, a fundamental question is how much we should trust a sensor network and how we prevent, or at least, to detect the misbehavior of the sensor network. Unfortunately, little research work has targeted to solve the problem. This paper tries to address the problem in a setting of network enhanced by some nodes with large storage space and considers a powerful and typical sensor network operation: range query. The network setting, we believe, will be a natural enhancement to the future sensor network. Range query is powerful enough to cover many interesting types of queries including location based queries. Thus, our model is generalized enough for us to investigate the trust problem in a practical and also meaningful environment. We envision that future sensor network shall be augmented by sparsely deployed special nodes for data storage. Those storage nodes differ from the regular sensors with a larger storage space (e.g., with more enriched flash memory). Sensor network
generates a large amount of data, and, many times, the collected data has to be archived for future retrieval. Data can be stored in the sensor nodes or sent back to the base station, each of which has its limitation. To store data on the sensor nodes is prohibitive due to the limited storage space on each sensor node and the difficulty in collecting all the data to a central repository. Transmitting all the data to the base station, on the other hand, has to address the limited transmission rate that is especially throttled by the funnel effect around the base station and attenuated per node transmission bandwidth. The introduction of the storage nodes helps to alleviate the transmission bandwidth problem by distributing the local data transmission to the storage node. This hierarchical structure has been instantiated by the recently popular stargate device [1] and the memory-enhanced sensor nodes by UC Riverside [2]. Those special powerful nodes take advantage of their high transmission capability and storage and even computational capability to alleviate the cursed bandwidth limitation, and also provide auxiliary support for surrounding vulnerable sensors for data back-up. The introduction of the storage node is also spurred by the recent concept of “data-centric storage” [3]. Data-centric storage deterministically conducts a mapping between the name of a data (N) to the address associated with a specific node (A). All the data with name N generated by the network are accumulated to the node A and all queries about N go to A too. In this way, network-wide search for data query is avoided dramatically reducing communication cost in many scenarios. For example, a sensor network deployed for plant monitoring may forward queries about the humidity to a storage sensor directly, and the temperature to another storage sensor instead of querying the entire network. Storage nodes have to take care of regular real-time sensor network operations, for example, data query, so that network user may monitor the sensor field and respond with the environmental change in a timely fashion. This natural requirement implies that the storage node has to gain some understanding about the stored data for an energy-efficient data reply by avoiding sending all the collected data back. The practice would not be a problem if the storage nodes are trusted as most of the sensor network research assumes. It is not valid, however, if the storage nodes are susceptible to compromise and the disclosure of the information may endanger the crucial assignment for the users in the network. With more sensor network deployed for pervasive computing applications, this issue becomes even more serious if the user information is leaked through the storage nodes, which breaches the privacy requirement. Generally, an adversary is not able to compromise numerous deployed sensors. The limited number of compromised sensors do not affect the query reply seriously because of redundant sensor deployment and limited coverage of the compromised sensors. The storage nodes, which hold much data collected from many sensors, however, will be the target for compromise and have to be a great concern when privacy related information is collected and query is imposed to the collected data. In this paper, we are particularly interested in the privacy implication of this network architecture, which we believe will be prevalent in the future sensor network deployment. We focus on data range query, which asks the storage nodes to return the data in a range specified by [a, b] with the sensor ids attached to each data1. We would like the storage
nodes reply the range query without gaining too much information. Since the storage nodes are not trusted, it is very likely that they lie about the collected data or discard some data. To prevent the malicious or non-cooperative storage nodes from cheating or dropping the data is hard, but at least the user is entitled to know whether the data reply is intact. In a word, we require the range query in this network setting to be privacy preserving and verifiable. Even though we use storage node as part of our network structure, the concept and the techniques used in this paper can also be applied to the regular range query in which aggregating nodes accumulate the results from the sensors in their charge. In storage node case, data has already been transmitted to the storage nodes before the query arrival, while in aggregating node case, the queries will be dissipated to the sensors online and the aggregating nodes will simply accumulate the replies and forward to the base station.

To the best of our knowledge, this paper is the first in dealing with privacy issue in sensor network data query. Even though there are papers on providing secure sensor network operations and verifying sensor operation results, most of them are focused on data aggregation, but not on more complicated range query, which is common and important in many sensor network applications. Our work explores the privacy concerns in sensor network in a very general setting and provides meaningful and interesting results for data reply verification.

II. Related Work

A. Privacy and Integrity Preserving in WSNs
Privacy- and integrity-preserving range queries in WSNs have drawn people’s attention recently. Sheng and Li proposed a scheme to preserve the privacy and integrity of range queries in sensor networks. The basic idea is to divide the domain of data values into multiple buckets, the size of which is computed based on the distribution of data values and the location of sensors. In each time-slot, a sensor collects data items from the environment, places them into buckets, encrypts them together in each bucket, and then sends each encrypted bucket along with its bucket ID to a nearby storage node. For each bucket that has no data items, a data storage model of sensor networks has been widely discussed in prior research work. Early work on this area considers the extreme models, archiving all data on the sink [4] or each sensor locally [5]. In [3], [6]–[9], new data storage system is designed by introducing an intermediate tier between the sink and sensors. This tier can cache data, process query and provide a more efficient access to the data collected by sensor networks. This paper considers the same system model, that some storage nodes are deployed as the intermediate tier and in charge of data archival and query response. Data privacy and security have attracted lots of work in database system [10]–[16]. In [10], the authors considered privacy problems in “Database as a Service” model. The service provider might not be trusted and data owner encrypted the data before sending it out. The authors proposed a data partition/bucketization scheme to allow service provider to process queries without decryption. Finally, the results are decrypted and processed at the client site. The privacy issues of outsourced database are also discussed in [13]. B. Hore et al. investigated data bucketing scheme and analyzed the tradeoff between performance and privacy. They further gave two measurements of privacy and designed an algorithm to optimize the
In Section IV-A, we inherit the privacy measurements in [13], [14] and apply them on sensor networks. However, in DAS model, database provider may be curious about sensitive data, but will not act in a malicious way. Users do not have to verify the reply. This paper considers a sensor network which is deployed in a hostile environment. Besides privacy protection, the other goal is to detect malicious behaviors. In addition, in sensor networks, data is distributed and we need consider communication efficiency in our design. There is some other related research on privacy protection of documents stored on untrusted sites. D. Song et. al. [17] described several schemes for keyword searching on encrypted data. Similarly, they considered the privacy issues with an untrusted storage server. The same problem is also discussed in [18]. Y. Chang et. al. resolved the problem by using a dictionary and interactive protocol. In addition, P. Golle et.al. proposed protocols particularly for conjunctive keyword search in [19]. But basically, the scenario they considered is for keyword search, not range query. And the data submitter is supposed to be the same as the requestor. Prior research about privacy issue in sensor networks focus on the location privacy [20], [21]. However, their protection target is the source sensor, not the data information. In this paper, we consider a system model consisting of special sensor nodes with large storage space. In practice, this kind of special nodes have been manufactured. Stargate [1] and RISE [2] are representative products. In [22], G. Mathur et. al. also attach external flash memory to sensor nodes and give a comprehensive evaluation of the performance. In addition, MicroHash [23] is a file system specifically designed for sensor nodes with large storage size. In our previous work [24], we propose an optimal deployment strategy of storage nodes in order to maximize performance improvement. In sensor networks, secure aggregation [25], [26] is a similar topic with our work of reply verification. L. Hu and D. Evans [25] proposed a protocol to prevent intermediate aggregators transmitting false information by using MAC messages as a signature. In their design, one aggregator is able to verify the information from its children by the messages from its grandchildren. This scheme, however, does not work for the case where multiple nodes are compromised. In [26], B. Przydatek et. al. proposed an aggregate-commit-prove scheme to verify the aggregation result. Sampling theory is applied in the protocol, which enables the sink to estimate the probability that the result is within a tolerant error range. However, both approaches are not for privacy-sensitive data. In addition, the goal of [25] is to find malicious aggregators, not suspect data sources, and the scheme in [26] is designed for aggregation queries.

III. Privacy Preserving Storage
We first discuss the protection of data privacy, i.e., preventing data from being disclosed to storage nodes. For this purpose, storing plaintext data on storage nodes will leak information to the storage nodes. Instead, data is encrypted by each sensor before it is sent to the storage nodes so that storage nodes are not able to access the collected data. Here every sensor and the sink share a secret key for an epoch, which makes up a one-way key chain. That is, Let $k_{i,t}$ be the secret key of sensor $s_i$ at epoch $t$, $k_{i,t} = \text{hash}(k_{i,t-1})$. After an epoch, the old key is erased from the sensor and a new key is generated by the embedded hash function. Thus, compromising a sensor $s_i$ as well as a storage node does not
lead to the disclosure of the information stored before the current epoch. Each sensor possesses a distinct key chain so that compromising one sensor does not affect the security of another sensor's data. After sink receives the query reply from storage nodes, the shared key between the sink and the corresponding sensor aids to decrypt the received data. Leaking no information to the storage nodes provides good privacy, but does not help with range query: The storage nodes have to send all the stored data back to the sink for a range query request, which consumes too much energy. The solution is to expose some information to the storage nodes while a good level of privacy is still maintained. We adopt bucketing scheme to associate a tag with each encrypted data as in [10], [13]. In this approach, the value domain is divided into multiple buckets and each bucket is assigned with a tag. There is no overlap or gap between consecutive buckets, i.e., every value is covered by exactly one bucket. Sensors and the sink have agreed on the same range partition, which is unknown to storage nodes. When sending data to the storage nodes, sensors attach the corresponding tag to every encrypted data based on which bucket the data falls into. The data values assigned with the same tag can be encrypted as a block. For example, a sensor may send the following to the storage node: sensor → Storage Node : sid, t, \{Tag1, \{data1, data2\}ki,t\}, \
\{Tag2, \{data3\}ki,t\}, . . . , where sid is the sensor ID and t is the value of current epoch counter. For a range query \([a, b]\), the sink first translates it into a list of tags which are associated to the smallest set of buckets that cover range \([a, b]\). Therefore, query sent to storage nodes is composed of this list of eligible tags, instead of a and b, for example: Sink → Storage Node : t, \{Tag1, Tag2, . . .\}, where t is the epoch specified by users. Storage nodes will look up all tags of the data generated in epoch t and return the data whose tag is listed in the query. Verify Reply The bucketing scheme discussed above achieves privacy preserving storage such that little information about data is exposed to storage nodes. However, as we mentioned, if storage nodes behave maliciously, they may send back arbitrary data as the query reply. In the following, we discuss counter schemes to detect false reply of range query. More precisely, there are three possibilities for a storage node to cheat on a range query reply. First, a storage node can forge a data value for query reply. The forged data can be easily detected because each valid data is encrypted by a key shared by the sink and the sensor who generates the data. Second, a storage node can launch incorrect reply attack, in which the storage node replies with an encrypted data that lies out of the required query range. The sink can also easily detect the cheating by decrypting the data and comparing with the query range. Third, a storage node may return partial portion of the desired data, which constructs an incomplete reply. In this section, we focus on detecting the incomplete reply.

IV. Performance Analysis

We evaluate our scheme based on simulations. The resulting encoding length is sufficient to protect query reply. Furthermore, we use real datasets to simulate. We show the communication cost incurred by this approach. The following parameters are involved in the simulation: We first run Algorithm 1 to estimate the optimal encoding length for a single tag query.
By default, we set \{n, s, \_ , \_ , PTi\} to \{100, 10, 0.1, 0.9, 0.1\}. In the simulation, the experimental results from our side-by-side comparison show that SafeQ significantly outperforms the S&L scheme for multidimensional data in terms of power and space consumption. For the two integrity-preserving schemes, the neighborhood-chaining technique is better than Merkle hash tree technique in terms of both power and space consumption. The rationale for us to include the Merkle hash-tree-based scheme is that Merkle hash trees are the typical approach to achieving integrity. For power consumption, SafeQ-NC+ consumes about the same power for sensors and 0.7 times less power for storage nodes; SafeQ-MHT+ consumes about the same power for sensors and 0.3 times less power for storage nodes; SafeQ-NC consumes 1.0 times more power for sensors and 0.7 times less power for storage nodes; and SafeQ-MHT consumes 1.0 times more power for sensors and 0.3 times less power for storage nodes. For space consumption on storage nodes, SafeQ-NC+ and SafeQ-MHT+ consume about the same space, and SafeQ-NC and SafeQ-MHT consume about 1.0 times more space. show that the power and space savings of SafeQ over prior art grow exponentially with the number of dimensions. For power consumption, for three-dimensional data, SafeQ consumes 184.9 times less power for sensors and 76.8 times less power for storage nodes. For space consumption on storage nodes, for three-dimensional data, SafeQ uses 182.4 times less space. Our experimental results conform with the analysis that the power and space consumption in the S&L scheme grow exponentially with the number of dimensions, whereas those in SafeQ grow in early with the number of dimensions times the number of data items.

\section*{V. Conclusion}

This paper proposed a technique for ensuring query privacy for on-demand WSN access. we consider an important problem in real sensor network deployment: how do we preserve the privacy and verify the query reply for a common type of query: range query. We build our scheme in a network augmented with storage nodes that are equipped with more storage space. To preserve privacy, we use bucketization to obscure the view of the storage nodes to the data stored. The collected data in a range is encrypted without being identified by the storage node, but the data is associated with a tag that tells the storage node which range it belongs to. In this way, storage node may reply to the range query without knowing the exact value of the data. The query reply from the storage nodes will be verified by examining the attached certificate so that the storage node is unable to forge data for reply. To prevent storage node from dropping data, an encoding number is generated on each sensor if no data in a range is collected on that sensor. The storage node has to provide the encoding number if it does not send the data for a range from a sensor, analysis, and simulation results on our proposed scheme.

\section*{References}


Mr. T Sravan is a student of Sree Rama Institute of Technology & Science, Kuppenakunta, Penuballi, Khammam, AP, India. Presently he is Pursuing his M.Tech (CN & IS) from this college. His area of interests includes Information Security, Cloud Computing, Data Communication & Networks.

Mr. S. Suresh well known Author and excellent teacher. He is working as a H.O.D for the Department of M.Tech., Computer Science & Engineering, Sree Rama Institute of Technology & Science, Kuppenakunta, Penuballi, Khammam. He has vast teaching experience in various engineering colleges. To his credit couple of publications both National & International conferences / journals. His area of Interest includes Data Warehouse and Data Mining, information security, Data Communications & Networks and other advances in Computer Applications. He has guided many projects for Engineering Students.