



IWNest PUBLISHER

## Journal of Industrial Engineering Research

(ISSN: 2077-4559)

Journal home page: <http://www.iwnest.com/AACE/>

# Maintaining Privacy and Integrity in two-tiered Wireless Sensor Network using Advanced SafeQ Protocol

<sup>1</sup>A. Emimanothaya and <sup>2</sup>Dr. R. Suresh Babu

<sup>1</sup>PG student, Communication and Networking, Department of Electronics and Communication Engineering, Kamaraj College of Engineering and Technology, Virudhunagar, 626001, Anna University, Chennai, Tamilnadu, India.

<sup>2</sup>Associate professor, Department of Electronics and Communication Engineering, Kamaraj College of Engineering and Technology, Virudhunagar, 626001, Anna University, Chennai, Tamilnadu, India.

### ARTICLE INFO

#### Article history:

Received 22 February 2015

Accepted 20 March 2015

#### Keywords:

Integrity, Privacy, Range queries, Storage nodes, Sensor networks.

### ABSTRACT

Maintaining Privacy and Integrity in two-tiered Wireless Sensor Network using Advanced SafeQ Protocol. To protect the Privacy and Integrity of two-tiered Wireless Sensor Network using Advanced SafeQ Protocol in Transport layer. Advanced SafeQ significantly outperforms prior art for multidimensional data in terms of both power consumption and storage space. Merkle hash tree and neighborhood chain are used for handling integrity for verifying related queries. Bloom filters technique is used for reduce communication cost on sensor network. Finally, the solution proposed is to adapt Advanced SafeQ protocol for event driven sensor networks. The upper bound of the false positive rate decreases as the number of hash function increases or the number of data items increases.

© 2015 IWNest Publisher All rights reserved.

**To Cite This Article:** A. Emimanothaya and Dr. R. Suresh Babu., Maintaining Privacy and Integrity in two-tiered Wireless Sensor Network using Advanced SafeQ Protocol. *J. Ind. Eng. Res.*, 1(3), 16-23, 2015

## INTRODUCTION

Wireless sensor networks (WSNs) have been widely deployed for various applications, such as environment sensing, building safety monitoring, earthquake prediction, etc. The security of data is a major threat in wireless communication environment. It is a very big challenge to the society to transfer the peoples data in a speedy manner with accuracy. To avoid the active and passive attacks of wireless communications, this paper introduces a new protocol which is called as Advanced SafeQ, and it will provide the solution for data integrity and privacy. Considering a two-tiered sensor network architecture in which storage nodes gather data from nearby sensors and answer queries from the sink of the network[1]. The storage nodes serve as an intermediate tier between the sensors and the sink for storing data and processing queries. Storage nodes bring three main benefits to sensor networks. First, sensors save power by sending all collected data to their closest storage node instead of sending them to the sink through long routes. Second, sensors can be memory-limited because data are mainly stored on storage nodes. Third, query processing becomes more efficient because the sink only communicates with storage nodes for queries[2].

As storage nodes store data received from sensors and serve as an important role for answering queries, they are more vulnerable to be compromised, especially in a hostile environment. The storage node imposes the significant threats to a sensor network. The attackers may obtain sensitive data that has been stored in the storage node. The storage node may return the forged data for the query. This storage node may not include all data items that satisfy the query. We want to design a protocol that prevents attackers from gaining information from both sensor-collected data, sink issued queries, and allows the sink to detect compromised storage nodes when they misbehave. For *Privacy*, compromising a storage node should not allow the attacker to obtain the sensitive information that has been stored in the node. As well as the queries that the storage node has received, and will receive. For *Integrity*, the sink needs to detect whether a query result from a storage node includes forged data items or does not include all the data that satisfy the query. For solving the privacy and integrity, there are two key challenges. A storage node needs to correctly process encoded queries over encoded data without knowing their actual values. A sink needs to verify that the result of a query contains all the data items that satisfy the query and does not contain any forged data. Advance SafeQ uses a novel technique to encode

**Corresponding Author:** A. Emimanothaya, PG student, Communication and Networking, Department of Electronics and Communication Engineering, Kamaraj College of Engineering and Technology, Virudhunagar, 626001, Anna University, Chennai, Tamilnadu, India.

both data and queries such that a storage node can correctly process encoded queries over encoded data without knowing their actual values to preserve the integrity. We propose two schemes. One using Merkle hash trees. Second a new data structure called neighbourhood chains. A solution to adapt Advanced SafeQ for eventdriven sensor networks then a sensor submits data to its nearby storage node only when a certain event happens and the event may occur infrequently. Our results show that the power and space savings of Advanced SafeQ over prior art grow exponentially with the number of dimensions[2]. Advanced SafeQ consumes 184.9 times less power for sensors and 76.8 times less power for storage nodes for three-dimensional data.

#### *Related Work:*

##### *A. Preserving the Integrity and Privacy in WSN's:*

In our scheme, we use bucket-partitioning idea for database privacy. Privacy- and integrity-preserving range queries in WSNs have drawn people's attention recently. The size of which is computed based on the distribution of data values and the location of sensors. A sensor collects data items from the environment encrypts them together in each bucket and then sends each encrypted bucket along with its bucket ID to a nearby storage node. The bucket is empty, when sensor sends an encoding number to a nearby storage node that can be used by the sink to verify that the bucket is empty. When the sink wants to perform, a range query it finds the smallest set of bucket IDs that contains the range in the query. The S&L scheme has two main drawbacks inherited from the bucket-partitioning technique[2]. The bucket-partitioning technique allows compromised storage nodes to obtain a reasonable estimation on the actual value of both data items and queries. For multidimensional data as well as the space consumption of storage nodes, the power consumption of both sensors and storage nodes increases exponentially with the number of dimensions due to the exponential increase of the number of buckets. The basic idea of their optimization is that each sensor uses a bit map to represent which buckets have data and broadcasts its bit map to the nearby sensors. An optimized version of S&L's integrity preserving scheme aiming to reduce the communication cost between sensors and storage nodes[4],[6]. The sink verifies query result integrity for a sensor by examining the bit maps from its nearby sensors.

##### *B. Privacy preservice in the databases:*

We have observed the proposed bucket partitioning idea for querying encrypted data in the database-as-service model (DAS) where sensitive data are outsourced to an untrusted server. It is used the bucket-partitioning idea to investigate range queries on numerical data. It cannot be used to solve our privacy problem because it is too expensive for sensor networks. Hence, sensor to perform  $O(zD)$  encryption for each data submission that is the number of dimensions and is the domain size of each dimension.

##### *C. Preserving integrity in the database:*

The Merkle hash trees have been used for the authentication of data elements and they were used for verifying the integrity of database queries. In a database signature of the each tuple by signing the concatenation of the digests of the tuple itself as well as the tuple's left and right neighbors. Moreover, our neighborhood chaining technique seems similar to the above signature aggregation and chaining technique. For storing the counting information for multidimensional data such that this counting information can be used for integrity verification without leaking boundary information. The result requires each sensor to compute and send an encrypted multidimensional CRT with approximately  $n(\log D)z$  overhead to a storage node.

##### *D. Untrusted servers for Secure File System:*

Our aims to design a system where users can store their files on an untrusted server and the server cannot read the content of the files. The files are not able to process the query in the untrusted server. Our main design goal for Advanced SafeQ processing queries in a privacy-preserving manner at storage nodes.

#### *Problem Statement:*

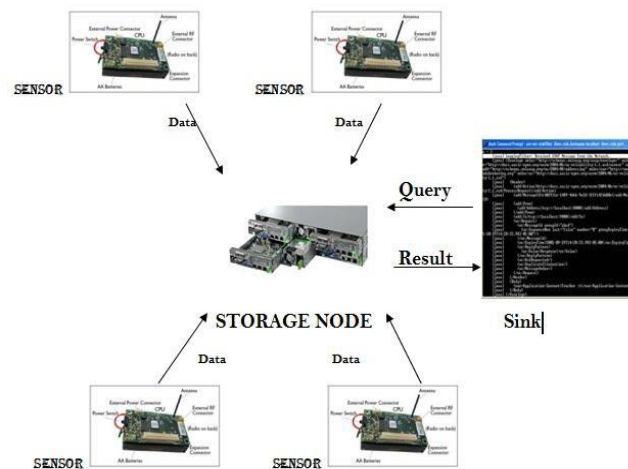
As shown in the fig.1 two-tiered sensor networks as illustrated.

There are three types of nodes considered in the two tiered sensor network. They are

- a. Sensor
- b. Storage node
- c. A sink (Base Station)

Limited storage and the computation power for the sensors, which are the inexpensive sensing device. Storage nodes are powerful wireless devices that are equipped with much more storage capacity and computing power than sensors. The contact of user is done by the sink or the base station. First translates the question into multiple queries and then disseminates the queries to the corresponding storage nodes. The query result is unified from the multiple storage nodes into the final answer and sends it back to the user. With loose synchronization and every sensor collects data once per *time interval* we divide time into fixed duration

intervals. A *Time-slot* is formed, from the starting time that all sensors and the sink agree upon every time intervals. After a sensor collects data for  $n$  times it sends the message that contains a 3-tuple  $(i, t, \{d1\text{---}dn\})$ . A range query “finding all the data items collected at time-slot in the range  $[a, b]$ ” is denoted as  $\{t, [a, b]\}$ . We address privacy and integrity-preserving ranges queries for event-driven sensor networks.



**Fig. 1:** Architecture of two-tiered sensor networks.

**Table 1:** Summary of Notation.

Symbol	Description
$S_i$	A sensor with ID $i$
$K_i$	The secret key of sensor $s_i$
$T$	The sequence number of a time slot
$d1\text{---}dn$	$n$ 1-dimension data items
$D1\text{---}Dn$	$n$ z-dimension data items
H,G,E	Three “magic” functions
$F(x)$	The prefix family of $x$
$S(d1,d2)$	The minimum set of prefixes converted from $[d1,d2]$
$N$	A prefix numericalization function
$HMAC_g$	An HMAC function with key $g$
QR	A query result
VO	An integrity verification object

The sensors and the sink are trusted for a two-tiered sensor network[5]. The sensors and the storage nodes are compromised in the hostile environment. The subsequent collected data of the sensor will be known to the attacker and the compromised sensor may send forged data to its closest storage node. Compromising a storage node can cause much greater damage to the sensor network than compromising a sensor. The data from one sensor constitute a small fraction of the collected data of the whole sensor network. The large quantity of data stored on the node will be known to the attacker.

A falsified result can be returned by the compromised storage node formed by including forged data or excluding legitimate data. In the two-tier architecture the fundamental problem statement is: “*How can we design the storage scheme and the query protocol in a privacy- and integrity-preserving manner?*” A satisfactory solution to this problem should meet the following two requirements.

### 1. Data and query privacy:

Data privacy means that a storage node cannot know the actual values of sensor collected data. In addition, query privacy means that a storage node cannot know the actual value of sink issued queries.

### 2. Data integrity:

If a query result that a storage node sends to the sink includes forged data or excludes legitimate data. Still, the query result is guaranteed to be detected by the sink as invalid.

### System Model:

We consider a Storage nodes are powerful wireless devices that are equipped with much more storage capacity and computing power than sensors. Each sensor periodically sends collected data to its nearby storage node. The sink is the point of contact for users of the sensor network. Each time the sink receives a question from a user; it first translates the question into multiple queries and then disseminates the queries to the

corresponding storage nodes which process the queries based on the storage node sends the results back to the sink. The sink unifies the query answers which are obtained from multiple storage nodes into a single answer and sends them to the user.

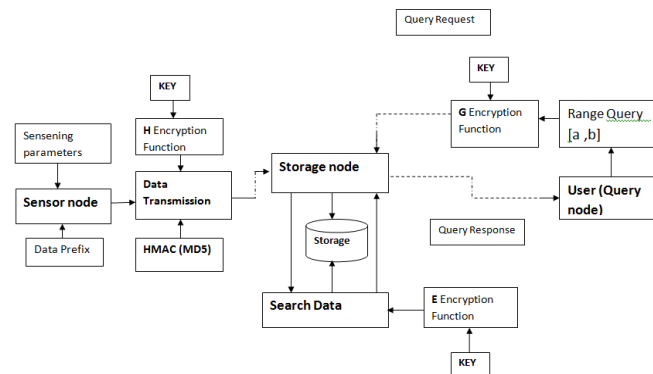


Fig. 2: Block diagram of two-tiered sensor networks.

Privacy & Integrity for One-Dimensional Data:

Each sensor  $s_i$  encrypts data items  $d_1, \dots, d_n$  using its secret key  $k_i$  and it is denoted as  $(d_1)k_i, \dots, (d_n)k_i$  to preserve privacy.  $k_i$  is a shared secret key with the sink. The key challenge is how a storage node processes encrypted queries over encrypted data without knowing their values. The values of data items and range queries to prevent a storage node from knowing the values. Consider sensor collected data  $\{1, 4, 5, 7, 9\}$  and a sink issued query  $[3,6]$  in Figure 2.

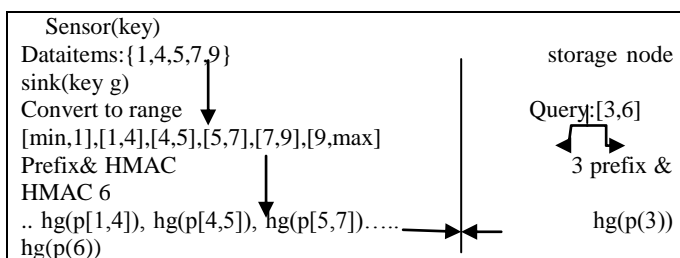


Fig. 2: SafeQ Privacy preserving scheme.

The sensor first converts the collected data to ranges  $[min,1], [1,4], \dots, [9,max]$ , where  $min$  and  $max$  denote the lower and upper bound for all possible data items. The sensor converts each range  $[d_j, d_{j+1}]$  to prefixes and is denoted as  $p([d_j, d_{j+1}])$ , and then apply HMAC to each prefix in  $p([d_j, d_{j+1}])$ . The sink performs query first converts 3 and 6 to prefixes. The query denoted as  $p(3)$  and  $p(6)$  and then apply HMAC to each prefix in  $p(3)$  and  $p(6)$  and denoted as  $hg(p(3))$  and  $hg(p(6))$ . On consequent receiving query  $hg(p(3))$  and  $hg(p(6))$  from the sink the storage node checks which  $hg(p([d_j, d_{j+1}]))$  has common elements with  $hg(p(3))$  or  $hg(p(6))$ . The query result of  $[3,6]$  includes two data items 4 and 5 to find in the storage node. The storage node sends  $(4)k_i$  and  $(5)k_i$  to the sink.

The query response from a storage node to the sink consists of two parts:

a. The query result QR:

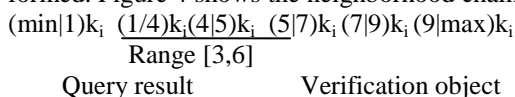
It includes all the encrypted data items that satisfy the query

b. The verification object VO:

It includes information for the sink to verify the integrity of QR

Neighborhood chaining technique is presented to preserve integrity of a query result. If a storage node excludes any data item that satisfies the query the sink can detect it, as sensor encrypts each item with its left neighbor.

As shown in the figure 3 the storage node, sink and sensor shows the how the neighborhood chain is formed. Figure 4 shows the neighborhood chain with the considered example.



For the range query [3,6] and the query result QR is  $\{(1|4)ki, (4|5)ki\}$  and the verification object V O is  $\{(5|7)ki\}$ . The items in QR and V O do not form a neighborhood chain when the sink can detect this error.

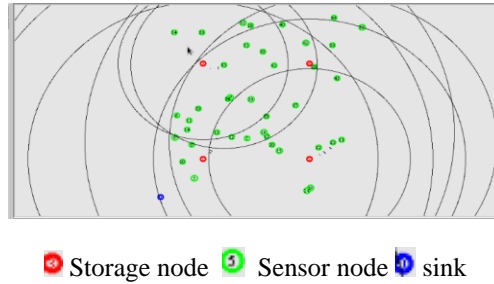


Fig. 3: Example neighborhood chain in WSN.

*Privacy and Integrity for Multi-Dimensional Data:*

As in the single dimensional privacy technique, each dimension in multi-dimensional is applied[6]. Sensor  $s_i$  collects 5 two-dimensional data items (1,11), (3,5), (6,8), (7,1) and (9,4), it will apply the 1-dimensional privacy preserving techniques to the first dimensional values  $\{1, 3, 6, 7, 9\}$  and the second dimensional values  $\{1, 4, 5, 8, 11\}$ . To preserve the integrity of multi-dimensional data we build a multi-dimensional neighborhood chain. Fig.5 illustrates this chain the two grey points denote the lower and upper bounds.

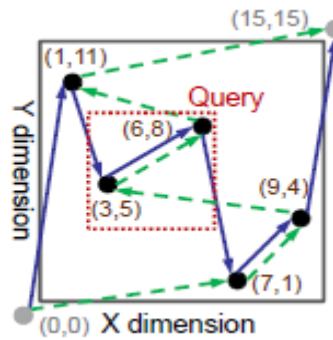


Fig. 4: A 2-dimensional neighborhood chain.

The dashed arrows illustrate the chain along the Y dimension and solid arrows illustrate the chain along the X dimension[6]. Bloom filters introduce false positives in the result of a query, i.e., the data items that do not satisfy the query. We can control the false positive rate by adjusting Bloom filter parameters.

Let  $eff$  denote the average false positive rate and  $w_h$  denote the bit length of each number in  $HMAC_g(N(S([d_j, d_{j+1}])))$ . For simplicity, we assume that each set contains the same number of values, which is denoted as  $n$ . The upper bound of the average false positive rate( $eff$ ) is given in (1).

$$eff < \frac{1}{3} \frac{(n+2)(n+3)}{(n+1)^{k-1}} (1 - e^{-\frac{k(n+1)q}{c}}) \tag{1}$$

Where, n-no of data items, k-no of hash function, q-query, c-communication cost.

To represent  $HMAC_g(N(S([d_0, d_1]))), \dots, HMAC_g(N(S([d_n, d_{n+1}])))$ , without Bloom filters,

the total number of bits required with Bloom filters, the total number of bits required is  $w_h(n+1)q$  at most  $c+2k(n+1)q[\log_2(n+1)]$ , the calculation of which is in Section VI. Therefore, our optimization technique reduces the communication cost if

$$w_h(n+1)q > c+2k(n+1)q[\log_2(n+1)] \tag{2}$$

the upper bound of the false positive rate decreases as the number of data items increases or the number of hash functions increases.

*Advanced Safeq Optimization:*

To reduce the communication cost between sensors and storage nodes we present an optimization technique based on Bloom filters. This cost can be significant because of two reasons.

- A sensor needs to convert each range  $[d_j, d_{j+1}]$
  - The sensor applies HMAC to each prefix number that results in 128-bit string
- Our basic idea is to use a Bloom filter to represent  $\text{HMACg}(N(S[d_0, d_1])), \dots, \text{HMACg}(N(S[d_n, d_{n+1}]))$ . A sensor only needs to send the Bloom filter instead of the hashes to a storage node. Consider the data items  $d_1, \dots, d_n$  use a Bloom filter to represent  $\text{hg}(p([min, d_1])), \text{hg}(p([d_1, d_2])), \dots, \text{hg}(p([d_n-1, d_n])), \text{hg}(p([d_n, max]))$ . Bloom filter is send instead of the sensor of the hashes to a storage node. Number of bits needed to represent the Bloom filter is much smaller than that needed to represent the hashes. Consider  $\text{hg}(p([4, 5]))=\{v_1\}$  and  $\text{hg}(p([5, 7]))=\{v_2, v_3\}$ . We can control the false positive rate by adjusting Bloom filter parameters.

Advanced Encryption Standard (AES) is a specification for the encryption of electronic data established by the U.S. AES is based on a design principle known as a substitution-permutation network, and is fast in both software and hardware.

AES (Advanced Encryption Standard) is a light weight algorithm. It is used because it consumes less processing power. Sensors should transmit using less power hence AES is used.

#### Event-Driven Networks Queries:

We have assumed that at each time-slot a sensor sends to a storage node the data that it collected at that time-slot. This assumption does not hold for event-driven networks that a sensor only reports data to a storage node when a certain event happens. The sink cannot verify whether a sensor collected data at a time-slot when if we directly apply our solution. We address the above challenge by sensors reporting their idle period to storage node each time when they submit data after an idle period or when the idle period is longer than a threshold. Hence, storage nodes can use such idle period reported by sensors to prove to the sink that a sensor did not submit any data at any time-slot in that idle period. *Sensors: An idle period* for a sensor is a time-slot interval  $[t_1, t_2]$  that indicates that the sensor has no data to submit from  $t_1$  and  $t_2$ . Let be the threshold of a sensor being idle without reporting to a storage node. *Storage Nodes:* When a storage node receives a query from the sink then first it checks weather si has submitted data at time-slot. *Sink:* Changes on the sink side are minimal.

#### Security & Complexity Analysis:

A protected two-tiered sensor network compromising a storage node does not allow the attacker to obtain the values of sensor-collected data and sink issued queries in the Advanced SafeQ[5]. A storage node only receives encrypted data items and the secure hash values of prefixes converted from the data items only in the submission on the protocol. It is computationally infeasible to compute the actual values of sensor collected data, without knowing the keys used the corresponding prefixes in the encryption and secure hashing. The key used in the secure hashing is without knowing the computationally infeasible to compute the actual values of sink issued queries. The result of query can be detected by the sink, which contains all the data items that satisfy the query and whether it contains forged data.

**Table 2:** Complexity analysis of SafeQ.

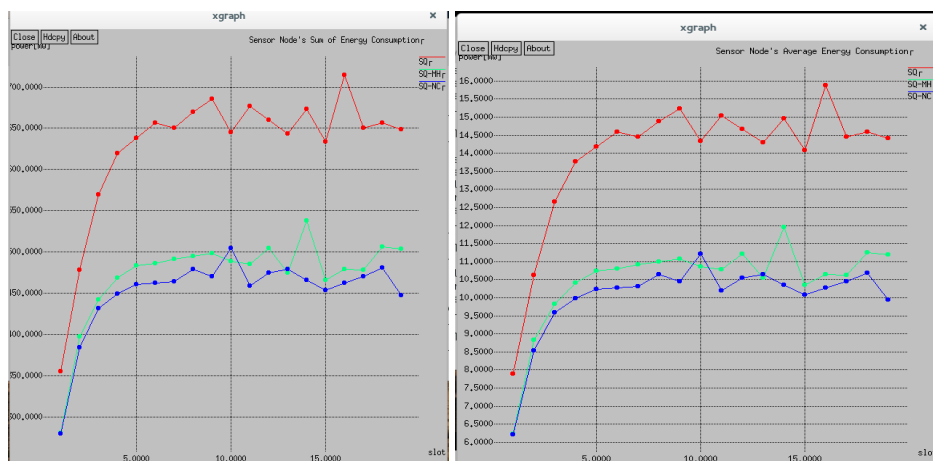
	Computation	Communication	Space
Sensor	$O(zn)$ hash $O(n)$ encryption	$O(zn)$	-
Storage node	$O(z)$ hash	$O(zn)$	$O(zn)$
Sink	$O(z)$ hash	$O(zn)$	-

Excluding any item in the middle or changing any item violates the chaining property. Based on the three properties QR and V O, the correctness of this claim satisfy for a query. As shown in the table2  $n$   $z$ -dimensional data items that a sensor collects in a time slot and storage space of Advanced SafeQ are described. The communication cost denotes the number of bytes sent for each submission or query.

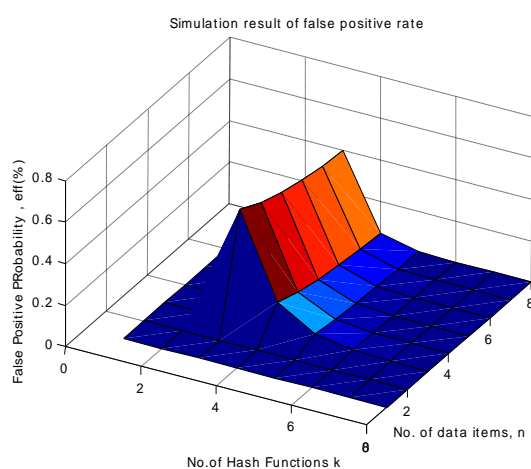
#### Experimental Analysis:

Our experimental results shows the Advanced SafeQ-Bloom consumes 184.9 times less power for sensors and 182.4 times less space for storage nodes. We implemented both Advanced SafeQ and the state-of-the-art on a large real data set. For 3-dimensional data, SafeQ Bloom consumes 10.3 times less power for sensors and 10.2 times less space for storage nodes. As shown in the fig.5 the average power and space consumption for 3-dimensional.

The three-dimensional shows the Advanced safeQ-NC+ consumes 182.4 times less space and Adnanced SafeQ-MHT+ consumes 169.1 times less space. As shown in the fig.5 the average space consumption of storage nodes for each data item versus the number of dimensions of the data item. Fig 6 show that the upper bound of the false positive rate decreases as the number of hash function increases or the number of data items increases.



**Fig. 5:** Ave. power and space consumption for 3-dimensional data.



**Fig. 6:** Simulation result of false positive rate.

#### Conclusion:

In our scheme, we mainly contribute three factors. They are Advanced SafeQ, Optimized technology using bloom filters and Adaption of Advanced safeQ. It is a novel and efficient protocol for handling range queries in two-tiered sensor networks in a privacy- and integrity-preserving fashion. Prefix membership verification, Merkle hash trees, and neighborhood chaining are used in the Advanced safeQ. It significantly strengthens the security of two-tiered sensor networks. Our results show that Advanced SafeQ significantly outperforms prior art for multidimensional data in terms of both power consumption and storage space. Using the bloom filters, an optimized technology is significantly used to reduce the communication cost between sensors and storage nodes.

#### REFERENCES

- [1] Sheng, B. and Q. Li, 2008. "Verifiable privacy-preserving range query in two tiered sensor networks," in Proc. IEEE INFOCOM, pp: 46-50.
- [2] Bo Sheng, Qun Li, 2011. "Verifiable privacy-preserving sensor network storage for range query" IEEE Transaction on Mobile Computing, 10(9).
- [3] Chen, F. and A.X. Liu, 2010. "SafeQ: Secure and efficient query processing in sensor networks," in Proc. IEEE INFOCOM, pp: 1-9.
- [4] Fei Chan and Alex X. Liu, 2012. "Privacy-and Integrity Preserving Range Queries in Sensor Networks," IEEE Transaction on networking, pp: 99.
- [5] Shi, J., R. Zhang and Y. Zhang, 2009. "Secure range queries in tiered sensor networks," in Proc. IEEE INFOCOM, pp: 945-953.
- [6] Jing Shi, Rui Zhang, Yanchao Zhang, 2011. "A Spatiotemporal approach for secure range queries in tiered sensor networks", IEEE Transaction on Wireless Communication, 10(1).
- [7] Zhang, R., J. Shi and Y. Zhang, 2009. "Secure multidimensional range queries in sensor networks," in Proc. ACM MobiHoc, pp: 197-206.

- [8] Thuc D. Nguyen, Thach V. Bui, Van H. Dang and Deokjai Choi, 2012. "Efficiently Preserving Data-Privacy Range Queries in Two-Tiered Wireless Sensor Networks," IEEE International Conference on Ubiquitous Intelligence and Computing.