ISSN 2395-695X (Print)
ISSN 2395-695X (Online)
International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)

Vol. 2, Special Issue 10, March 2016

RESOURCE CONSTRAINED SECURE DISTRIBUTED DATABASE MANAGEMENT SCHEME FOR WSN

M.Dinesh Kumar[1], M.Somu[2]

[1,2]Department of Computer Science and Engineering
[1,2]KSR College of Engineering, Tiruchengode

ABSTRACT

Environment monitoring is the main operation for the sensor nodes. Sensor networks are building with a group of sensor nodes. Sensor networks are constructed with energy and storage constraints. Data collection points are referred as sink nodes. Location and time factors are considered in the data query schemes. Distributed database management schemes are building to manage sensor network data collection tasks.

Sensor network is a huge volume of data observing environment. All the user queries are processed by the sensor databases. Centralized and distributed database approaches are adapted in the wireless sensor network environment. Data query operations are performed in the distributed data transmission model. Data transmission load is reduced using data compression and prediction methods. Optimizer is used to analyze the user query values. Query evaluation is carried out using the meta data information. Meta data based query optimization is performed in the query execution process.

Resource constraints are adapted to improve the distributed database management scheme. Tree structured data process is supported using the XML schema. The security schemes are designed with resource factors. Load distribution mechanism is integrated with the distributed query process under the sensor networks.

1. INTRODUCTION

Wireless sensor networks (WSNs) provide the flexibility of untethered sensing, but pose the challenge of achieving long lifetime with a limited energy budget, often provided by batteries. It is well-known that communication is the primary energy drain, which is unfortunate, given that the ability to report sensed data motivates the use of WSNs in several pervasive computing applications.

An approach to reduce communication without compromising data quality is to predict the trend followed by the data being sensed, an idea at the core of many techniques [1]. This data prediction approach1 is applicable when data is reported periodically—the common case in many pervasive computing applications. In these cases, a model of the data trend can be computed locally to a node. This model constitutes the information being reported to the data collection sink, replacing several raw samples. As long as the locally-sensed data are compatible with the model prediction, no further communication is needed: only when the sensed data deviates from the model, must the latter be updated and sent to the sink.

The aforementioned approach is well-known and has been proposed by several works we concisely survey. Nevertheless, to the best of our knowledge none of these works has been verified in practice, in a real-world WSN deployment. The techniques employed are relatively complex and their effectiveness is typically evaluated based on implementations in high-level languages on mainstream hardware platforms. Their feasibility on resource-scarce WSN devices remains unascertained. Moreover, the works in the literature typically evaluate the gains only in terms of messages suppressed w.r.t. a standard approach sending all samples. This data-centric view is quite optimistic. WSNs consume energy not only when transmitting and receiving data, but also in several continuous control operations driven by the network layer protocols, e.g., when maintaining a routing tree for data collection, or probing for ongoing communication at the MAC layer.

2. RELATED WORKS

The problem is inspired by the work in [5], which studies the problem of data sharing among multiple applications. It assumes each application only needs discrete data point samplings. While in our problem, the applications may require a continuous interval of data. The proposed solution in [5] cannot be applied to our problem. Our solution can solve their problem.

The problem is a novel one in WSNs. It tries to collect as little data as possible. Query optimization in WSNs [2] tries to find in-network schemes or distributed algorithms to reduce communication cost for aggregation queries. Our work focuses on reducing the amount of transmitted data for each node.

Multi-query optimization in database systems studies how to efficiently process queries with common sub expressions. It aims at exploiting the common sub-expression of SQLs to reduce query cost, while our problem aims at reducing data volume. Krishnamurthy et al. considered the problem of data sharing in data streaming systems for aggregate queries. They studied the min, max, sum and count-like aggregation queries. A stream is scanned at least once and is chopped into slices. Only the slices that overlap among multiple queries could be shared. Their studied problems are different from ours. We expect to reduce the number of sensor samplings at each individual node resulting in less communication cost. Our problem differs in that we want to provide each application enough sampled data while minimizing the total number of sampling times.

3. DATA QUERY MANAGEMENT SYSTEM FOR WSN

As in traditional database systems, the sensor databases try to create an abstraction between the end-users and the sensor nodes. This abstraction aims to permit the users to only concentrate on the needed data to be collected rather than bothering with the complexities of mechanisms deciding how to extract data from a network. As such, the sensor databases have been subject to two main approaches to data storage and query in WSNs [7]: the warehousing approach and the distributed approach.

1. In the warehousing approach, the sensors act as collectors. The data gathered by sensors are periodically sent to a central database where user queries are processed. This model is the most used one in data storage and query processing. It has some drawbacks, such as eventually wasting resources and creating a bottleneck with an immense amount of transmitted data. This approach is unsuitable for real-time processing.

2. The distributed approach is the alternative, where each sensor node is considered as a data source and then the WSN forms a distributed database where the sensed data are in the form of rows with columns representing sensor attributes. In this second approach, the sensed data are not periodically sent to the database server. They remain in the sensor nodes and some queries are injected in the network through the base station. These queries are disseminated into the network according to the routing techniques as per [3] and the sensors, thanks to their processing and storage capabilities, process them. The sensors send their data to their parent nodes whenever they correspond to the query requirements. The parent nodes combine this coming data with their own data and transmit to their parent nodes and so on until the data reaches the gateway. This approach that consists to process the data inside the sensor nodes themselves is called in network processing and it reduces the amount and size of transmitted data and the latency.

As largely detailed are four essential methods to design a distributed data management system: in-network processing, acquisition query processing, cross layer optimization and data-centric data/query dissemination. The in-network processing technique [4] generally includes the different types of operation that are traditionally done on the server, for instance, aggregations to inside the sensor nodes themselves. The acquisitional query processing permits to minimize energy consumption in the network by reducing the number of sensor nodes participating in the query processing. This reduction is done by expressing in the query when or what sensors to sample. Unlike the traditional computer networks in which layers in the conventional OSI model are separated and isolated, the cross- layer optimization [12] permits to combine information available on these different layers and profit from this information sharing. For instance, in wireless sensor networks the routing takes care of, among others, the quality of service (QoS) parameters of the network, network connectivity, the power available on the node and the network lifetime. In traditional computer networks the routing is done by the network layer only considering the destination address of a packet.

In contrast with traditional networks, nodes in WSNs usually do not have a single identifier because of data centric nature of sensor applications as well as the large number of sensors deployed. Generally, applications are not interested in specific sensors, but rather in data, which they generate. For example, a query as ‘‘which is the temperature measurement of the sensor with the ID XXXX’’ does not have much interest for a sensor application, but a query like ‘‘in which region, sensors measure fewer than 7^0C’’ is more significant. Routing protocols must take these characteristics into account.

4. DATA QUERY ISSUES IN WSN

Wireless Sensor Networks (WSNs) are composed of a large number of devices, called sensor nodes, which are able to sense, process and transmit information about the environment on which they are deployed. These devices are usually distributed in a geographical area to collect information for users interested in monitoring and controlling a given phenomenon. This information is transferred to a sink node to be accessible by remote users through generally application-level gateway, e.g. global sensor network (GSN). To obtain the data, these applications should also provide supports of efficient queries, which allow communication with the network [6].

In wireless sensor networks, the sensor nodes are battery powered and are considered intelligent with acquisitional, processing, storage and communication capacities [8]. These resources are generally very limited, especially in terms of storage and energy and the sensor nodes activities are sometimes not negligible in energy consumption [9], [10]. One of the most used techniques to save power is to activate only necessary nodes and to put other nodes to sleep[11]. Some authors 3 dimensional sensor field can be efficiently partitioned into cells to save energy.

Sensors can be placed anywhere there is data that should be collected, what makes information omnipresent. Consequently, systems based on sensor networks are increasingly common in many areas of the knowledge, giving rise to several flavors of WSNs [13], [14]. These numerous WSNs have allowed the development of many applications. In addition to data gathering and data replication issues such applications, a database oriented approach of WSNs has proven to be useful to manage the large amount of data generated by the sensors. According to this approach, a WSN is viewed as a distributed database where sensor nodes are considered as data sources with sensed data stored in the form of rows of a relation distributed across a set of nodes in the network. This database-oriented approach has motivated the design of WSN data acquisition with two fundamental objectives: similarly to traditional database systems, a WSN database should provide SQL-like abstractions so that nodes can be easily programmed for simple data sensing and collection. In addition, the data collection process should minimize the energy consumption in the network.

The main goal of distributed database management on WSNs is to support the management of the huge amount of sensed data in an energy-efficient manner. In fact, research into sensor hardware has shown that the energy depletion in the network is mainly due to the data communication tasks among the nodes. To deal with this problem, various data reduction techniques exist, including data aggregation, packet merging, data compression techniques, data fusion and approximation based techniques. The data compression techniques are also used to reduce the amount of data transmitted between the nodes, but they involve data encoding at the source nodes, data decoding at the sink node. The data fusion techniques refer to more complex operations on a data set and are usually used in multimedia data processing. The approximation based techniques use statistical techniques to approximate the queries results. These techniques provide, among other advantages, the reduction of the size of the transmitted data, the communication tasks, the network load and the data transmission time.

The aim of this paper is to show how distributed database techniques are adapted to wireless sensor networks to improve the management of the great amount of sensed data in an energy-efficient way by presenting and classifying the most recent and relevant proposals of distributed database management on WSNs. A discussion and open issues on distributed database management techniques for wireless sensor networks are identified to facilitate further contributions.

Large amount of sensed data are generated in sensor networks. Sensor databases create an abstraction between the end users and the sensor nodes. Warehouse approach based sensor database is constructed under the centralized environment to store and distribute the data values. In distributed database approach each sensor node is considered as a data source and the WSN forms. Distributed database approach is used for sensor networks to support energy efficient data storage and query operations. Data aggregation, packet merging, compression, data fusion and approximation techniques are served for data reduction process. User query is decomposed, optimized and distributed by the optimizer across the network. Meta data based query optimization is performed in the query execution process. The following issues are identified from the existing system. Load balancing policy is not considered. Hierarchical data organization is not supported. Data and query security is not provided. Multi query processing is not adapted.

5. RESOURCE CONSTRAINED SECURE DISTRIBUTED DATABASE MANAGEMENT SCHEME

Distributed Database management system is enhanced with power aware load balancing policies. Sensor data values are organized and processed in XML based model to adapt flexible data representations. Resource constraint based security services and multi query execution methods are integrated with the distributed database model. The system is improved to manage node failures in query process. The distributed database management scheme is adapted with security features. Node level and network level load distribution techniques are integrated with the system. Hierarchical data organization mechanism is employed in the system. The system is divided into six major modules. They are Distributed Database Construction, Meta Data Management, Structured Query Process, Hierarchical Query Process, Security Services and Load Distribution Process.

Distributed database construction module is designed to integrate the sensor databases. Sensor databases and attribute details are maintained under the meta data management process. SQL based data request operations are carried out under structured query process. XML based data access tasks are carried out under hierarchical query process. Data and query security operations are handled under the security services. Load distribution process is designed to manage query processing loads. Each sensor node is considered as single database. Sensor databases are grouped to construct the distributed database framework. Sensed data values are updated into the local database environment. Node properties are distributed to the network area. Meta data provides the sensor node and database information. Database name, IP address and capture scheme details are maintained in the database meta data. Data sensing properties are also maintained in the meta data model. Meta data values are distributed with reference to the request information

Structured Query Language (SQL) based data requests are processed by the nodes. Conditional query values are supported by the system. Parsing mechanism is used to verify the query syntax. Query assistance support model is provided to construct the query values. The distributed database system is enhanced to manage hierarchical data models. XML is used to organize and query the data values. Multi node data retrieval is supported by the hierarchical query process. Region based data aggregation is provided in the system.

Source Anonymous Message Authentication (SAMA) scheme is used for the security process. Elliptic Curve Cryptography technique is adapted for the security process. Query and data values are protected by the security services. Gamal signature scheme is adapted in the security process. The system performs the load distribution to manage resource consumption. Computational and energy loads are distributed in the node level load distribution process. Data transmission loads are shared under the network level load distribution models. Multi query execution is also managed with load balancing policies.

6. SECURITY MODEL FOR DATA QUERY PROCESS

Privacy is sometimes referred to as anonymity. Communication anonymity in information management has been discussed in a number of previous works. It generally refers to the state of being unidentifiable within a set of subjects. This set is called the AS. Sender anonymity means that a particular message is not linkable to any sender and no message is linkable to a particular sender. We will start with the definition of the unconditionally secure SAMA. A SAMA consists of the following two algorithms:

- Generate $(m,Q_1,Q_2, . . .,Q_n)$. Given a message m and the public keys $Q_1,Q_2, . . .,Q_n$ of the AS S = $\{A_1,A_2, . . .,A_n\}$, the actual message sender $A_{t,1} \leq t \leq n$, produces an anonymous message S(m) using its own private key dt.
- Verify S(m). Given a message m and an anonymous message S(m), which includes the public keys of all members in the AS, a verifier can determine whether S(m) is generated by a member in the AS. The security requirements for SAMA include:
- Sender ambiguity. The probability that a verifier successfully determines the real sender of the anonymous message is exactly 1=n, where n is the total number of members in the AS.
- Unforgeability. An anonymous message scheme is unforgeable if no adversary, given the public keys of all members of the AS and the anonymous messages $m_1,m_2, . . .,m_n$ adaptively chosen by the adversary, can produce in polynomial time a new valid anonymous message with non-negligible probability.

In this paper, the user ID and the user public key will be used interchangeably without making any distinctions. The appropriate selection of an AS plays a key role in message source privacy, since the actual message source node will be hidden in the AS. We will discuss techniques that can prevent the adversaries from tracking the message source through the AS analysis in combination with local traffic analysis.

Before a message is transmitted, the message source node selects an AS from the public key list in the SS as its choice. This set should include itself, together with some other nodes. When an adversary receives a message, he can possibly find the direction of the previous hop, or even the real node of the previous hop. The adversary is unable to distinguish whether the previous node is the actual source node or simply a forwarder node if the adversary is unable to monitor the traffic of the previous hop. Therefore, the selection of the AS should create sufficient diversity so that it is infeasible for the adversary to find the message source based on the selection of the AS itself. Some basic criteria for the selection of the AS can be described as follows:

- To provide message source privacy, the message source needs to select the AS to include nodes from all directions of the source node. In particular, the AS should include nodes from the opposite direction of the successor node. In this way, even the immediate successor node will not be able to distinguish the message source node from the forwarder based on the message that it receives.

- Though the message source node can select any node in the AS, some nodes in the AS may not be able to add any ambiguity to the message source node. For instance, the nodes that are apparently impossible or very unlikely to be included in the AS based on the geographic routing. Therefore, these nodes are not appropriate candidates for the AS. They should be excluded from the AS for energy efficiency.
- To balance the source privacy and efficiency, we should try to select the nodes to be within a predefined distance range from the routing path. We recommend selecting an AS from the nodes in a band that covers the active routing path. AS does not have to include all the nodes in the routing path.
- The AS does not have to include all nodes in that range, nor does it have to include all the nodes in the active routing path. In fact, if all nodes are included in the AS, then this may help the adversary to identity the possible routing path and find the source node.

As an example, suppose we want to transmit a packet from source node S to destination node D. We select the AS to include only nodes marked with while nodes marked as will not be included in the AS. Of all these nodes, some of them are on the active routing path, while others are not. All these nodes are located within the shaded band area surrounding the active routing path. Suppose node A is compromised, unless node A collaborates with other nodes and can fully monitor the traffic of the source node S, it will not be able to determine whether S is the source node, or simply a forwarder. Similar analysis is also true for other nodes.

Any node in the active routing path can verify the contents' authenticity and integrity. Anybody who receives a packet in the transmission can possibly exclude some of the nodes in the WSNs as the possible source node. Inclusion of these nodes in the AS will not increase the source privacy. Nevertheless, the more the nodes included in the AS are, the higher the energy cost will be. Therefore, the selection of the AS has to be done with care so that the energy cost and the source privacy can both be optimized. In addition, to balance the power consumption between authenticity and integrity verification and the possibility that corrupted messages are being forwarded, the verification service may not have to take place in every hop; instead, it may be configured to take place in every other hop, for instance.

7. CONCLUSION

Wireless sensor networks are considered as distributed database model. Distributed database management scheme is employed to perform data storage and query processing tasks. The system is enhanced with security and load balancing features. Multi query processing, meta data management and node failure handling schemes are also integrated with the system. Load balanced query processing environment is supported for the wireless sensor networks.

Hierarchical data management scheme supports multi node based data access features. The system provides security for query and data values. Energy and bandwidth consumption is reduced in the system.

REFERENCES

[1] T. Palpanas, “Real-time data analytics in sensor networks,” in Managing and Mining Sensor Data, C. Aggarwal, Ed. New York, NY, USA: Springer, 2012.

[2] Hong Gao, Xiaolin Fang, Jianzhong Li and Yingshu Li, “Data Collection in Multi Application Sharing Wireless Sensor Networks”, IEEE Transactions On Parallel And Distributed Systems, Vol. 26, No. 2, February 2015

[3] C. Li, H. Zhang, B. Hao and J. Li, ‘‘A Survey on Routing Protocols for Large-Scale Wireless Sensor Networks,’’ Sensors, vol. 11, no. 4, pp. 3498-3526, Mar. 2011.

[4] P. Andreou, D. Zeinalipour-Yazti, P.K. Chrysanthis and G. Samaras, ‘‘In-Network Data Acquisition and Replication in Mobile Sensor Networks,’’ Distrib. Parallel Databases, vol. 29, no. 1/2, pp. 87-112, Feb. 2011.

[5] A. Tavakoli, A. Kansal and S. Nath, ‘‘On-Line Sensing Task Optimization for Shared Sensors,’’ in Proc. 9th ACM/IEEE Int’l Conf. IPSN, 2010, pp. 47-57.

[6] L.M.L. Oliveira and J.J.P.C. Rodrigues, ‘‘Wireless Sensor Networks: A Survey on Environmental Monitoring,’’ J. Commun., vol. 6, no. 2, pp. 143-151, Apr. 2011.

[7] O. Diallo, J.J. Rodrigues and M. Sene, ‘‘Real-Time Data Management on Wireless Sensor Networks: A Survey,’’ J. Netw. Comput. Appl., vol. 35, no. 3, pp. 1013-1021, May 2012.

[8] A. Nayak and I. Stojmenovic, Wireless Sensor and Actuator Networks: Algorithms and Protocols for Scalable Coordination and DataCommunication. Hoboken, NJ, USA:Wiley-Interscience, 2010.

[9] K. Lin, M. Chen, S. Zeadally and J.J.P.C. Rodrigues, ''Balancing Energy Consumption with Mobile Agents in Wireless Sensor Networks,'' Future Gener. Comput. Syst., vol. 28, no. 2, pp. 446-456, Feb. 2012.

[10] K. Lin, J.J.P.C. Rodrigues, H. Ge, N. Xiong and X. Liang, ''Energy Efficiency QoS Assurance Routing in Wireless Multimedia Sensor Networks,'' IEEE Syst. J., vol. 5, no. 4, pp. 495-505, Dec. 2011.

[11] S. Sendra, J. Lloret, M. Garcia and J.F. Toledo, ''Power Saving and Energy Optimization Techniques for Wireless Sensor Networks,'' J. Commun., vol. 6, no. 6, pp. 439-459, Sept. 2011.

[12] L.D. Mendes and J.J. Rodrigues, ''A Survey on Cross-Layer Solutions for Wireless Sensor Networks,'' J. Netw. Comput. Appl., vol. 34, no. 2, pp. 523-534, Mar. 2011.

[13] P.A. Neves, J.J.P.C. Rodrigues, M. Chen and A.V. Vasilakos, ''A Multi-Channel Architecture for IPv6-Enabled Wireless Sensor and Actuator Networks Featuring PnP Support,'' J. Netw. Comput. Appl., vol. 37, pp. 12-24, Jan. 2014.

[14] L.M.L. Oliveira, J.J.P.C. Rodrigues, A.F. de Sousa and J. Lloret, ''Network Access Control Framework for 6LoWPAN Networks,'' Sensors, vol. 13, no. 1, pp. 1210-1230, Jan. 2013.