Secure Publish-Subscribe-based In-Network Data Storage Service in Wireless Sensor Networks

Muhammad Bashir Abdullahi and Guojun Wang*
School of Information Science and Engineering, Central South University,
Changsha, Hunan Province, P. R. China, 410083
*Correspondence to: csgjwang@mail.csu.edu.cn

***Abstract*—Recently, it was argued that the true potential of sensor network services can be realized only if they become an integral part of a larger context-aware distributed system. The benefit is that, the intergration allows sensor network services to be seamlessly accessible to remote users through devices that they are already familiar with. This integration can be simplified using a publish-subscribe system. However, the wireless medium facilitates eavesdropping and some data is sensitive. This makes the security extremely important. Furthermore, sensor nodes have limited power and processing resources, so standard security mechanisms, which are heavy in weight and resource consumption, are unsuitable. These challenges increase the need to develop comprehensive and secure solutions that achieve wider protection, while maintaining desirable network performance. In this paper, we propose a secure publish-subscribe-based in-network data storage service in WSNs to provide different security requirements such as confidentiality, authentication, integrity, availability, and authorized access.**

I. Introduction

A. Background and Motivation

Wireless sensor networks (WSNs) are deployed to intelligently monitor the surrounding environments, store, and make available the sensed data to the authorized network users upon their demands [1]. The collected data may be of interest to many network users from both public and private sectors ranging from individual users to universities, government research centers, and business companies. Thus, it is observed that the true potential of sensor network services can be realized only if they become an integral part of a larger context-aware distributed system [2]. The benefit is that, the integration allows users to communicate with the sensor nodes through diverse communication abstractions using devices such as mobile phones, personal digital assistants (PDAs), and laptops as shown in Figure 1. In addition, it enables sensor nodes across different networks to communicate with each other, as in any other distributed system [2].

This integration can be simplified using a publish-subscribe (pub-sub, for short) system. A pub-sub system is emerging as a very flexible communication paradigm that is reflecting the dynamic and decoupled nature of the applications [5].

The principle of pub-sub communication paradigm is that entities, which are interested in consuming certain information (*events*), register their interest. This process of registering an interest is called *subscription*; the interested party is therefore called a *subscriber*. Entities who want to produce certain information (*events*) do so by *publishing* their information. They are thus called *publishers*. The entity, which ensures that the data gets from the publishers to the subscribers, is called the *broker*. This process of delivering data to a subscriber if it matches a subscription is termed *notification* as shown in Figure 2. The broker coordinates subscriptions, and subscribers usually have to contact the broker explicitly to subscribe [6]. While this integration provides several benefits, there are essential security risks. This is due to the use of wireless medium that facilitates eavesdropping and adversarial intrusion and packet injection to disrupt the network functionality.

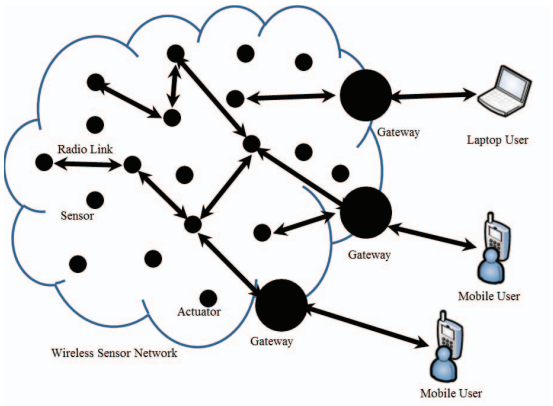

Fig. 1. Integrated Wireless Sensor Network

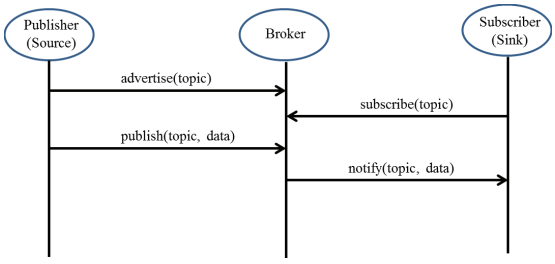

Fig. 2. Topic-based Pub-Sub Communication Model

Intuitively, there is need to provide basic security mechanisms such as authentication of publishers and subscribers and confidentiality of events and subscriptions in a pub-sub system. In particular, how to ensure:

- *Publication authenticity*: only the correct publications are

978-0-7695-4707-7/12 $26.00 © 2012 IEEE
DOI 10.1109/DCOSS.2012.58

delivered to the subscribers.

- *Subscription authentication*: only the authorized subscribers who subscribe (e.g., have paid) to the service received publications that matched their interest.
- *Publication and subscription integrity*: prevent unauthorized modifications of pub-sub messages.
- *Publication confidentiality*: perform content-based routing without the publishers trusting a pub-sub network.
- *Subscription confidentiality*: subscribers receive requested publications without disclosing their subscriptions to a pub-sub network.

In this paper, we propose a secure publish-subscribe-based in-network data storage service in WSNs. In our scheme, we use Bloom filters (BFs) [3] to represent interest. This not only reduces bandwidth requirement for interest propagation, but also provides confidentiality of the interest. Identity-based-encryption (IBE) [4],[8] is used to provide authentication of publishers and subscribers and confidentiality of events. The network controller (NC) divides all the registered users into groups. The users in the same group have the same access privileges.

B. Classification of Publish-Subscribe Systems

Pub-sub systems are classified into three types based on the different ways in which events of interest are specified: topic-based, type-based, and content-based [5].

- *Topic-based systems*: the list of topics or subjects is usually known in advance, e.g., during the design phase of an application. Subscriptions and publications can only be made on a specified set of topics.
- *Type-based systems*: a subscriber states the type of data it is interested in (e.g., temperature data). Type-based systems are not very common.
- *Content-based systems*: these are the most versatile ones. A subscriber describes the content of messages it wants to receive. Such a subscription could be for any messages containing both temperature and light readings where the temperature is below a certain threshold and the light is on.

The topic-based systems are considered to be the most appropriate for WSNs due to its simplicity when compared with other types of pub-sub systems [6].

II. Reference Communication Model

We use the following scenario of a smart hospital communication abstraction prototype developed in [2] to have a quick overview of seamless integration. The abstraction for remotely monitoring the temperature of blood samples of a patient is described as follows in two flavors, namely, push or pull:

Before users can interact with a sensor node, the node needs to register its services. An individual node offering a service, or a manager that represents a group of nodes offering the service, can register the service by sending a TINYSIP-REGISTER request to the gateway. The registration message includes the validity period of the registration, attributes of the service (e.g., instantaneous temperature of blood samples), and the identity of the node that is registering (e.g., mote-id, group-id, or a location identifier). When the TinySIP (Tiny session initiation protocol) gateway receives a TINYSIP-REGISTER request from a sensor node, it records the information in a local database and uses the information to route a message to the appropriate sensor node.

After the services are registered, a user can use TinySIP to interact with the sensor nodes with the help of the graphical user interface (GUI) client on the mobile device. For example, using the prototype in [7], a hospital staff member, such as a nurse, can use the SUBSCRIBE method in order to be notified when the temperature of the blood samples in a blood bank exceeds a certain threshold. This is useful to detect bacterial growth and prevent transfusion errors. When temperature sensors detect the event, they PUBLISH the event to the TINYSIP gateway, which then uses the NOTIFY method to deliver the notification as a text message on the mobile device of the subscribers. The MESSAGE method can be used to send an instant message to query the vital statistics of a patient or to control smart devices deployed in the hospital facility. Session semantics allows a phenomenon to be continuously monitored over a period of time and the corresponding data to be streamed from the sensor end point to the user.

A nurse or administrator uses the GUI client on the mobile phone to initiate a session with the temperature sensors that monitor the temperature of a patient, who is performing a treadmill test and to request the temperature data to be streamed to him for the duration of the test period. The request from the GUI client is first translated into a SIP INVITE request by the GUI application and sent to the contact unified resource identifier (URI) for the sensors (i.e., of the sensor gateway). When the sensor gateway receives the INVITE, it maps the request to a TINYSIP-INVITE message and routes the request to the appropriate sensor nodes. The sensor node that is responsible for providing the requested information accepts the invitation by returning an affirmative response OK. The gateway node forwards the response from the sensor network to the client. In turn, the client acknowledges the response from the sensor nodes by sending an ACK. After this handshake, the sensor nodes begin the media transfer according to the parameters agreed upon during the session establishment.

III. Models and Assumptions

A. Network Model

We consider a large wireless sensor network with densely deployed sensor nodes, many gateways, and many network users. Each sensor node has a unique identifier in the network. The network users access the sensed data using devices such as laptop PCs or PDAs. The NC generates and assigns the keying materials to access devices to enforce the access control policy. In addition, the NC groups together network users with similar access right to ensure a fine-grained data access control.

B. Threat Model

We assume that an adversary can launch both inside and outside attacks. The adversary may eavesdrop, copy, and replay transmitted messages in WSNs. A compromised publisher might try to fabricate some events or discovers a subscriber's subscription. A compromised subscriber might try to claim and decrypt subscription for events it did not subscribe to. We assume that the gateways are honest-but-curious entities, which will try to learn the contents of message (interest or event). However, the gateways will neither drop a message (interest or event), nor delete it. Therefore, our goals are:

- To protect the integrity of the message in transit.
- To protect the content such that only the authorized subscribers can read it.
- To authenticate the source of messages from publishers, subscribers, and brokers to one another.

IV. Secure Publish-Subscribe-Based In-Network Data Storage Service in WSNs

Our proposed scheme consists of initialization phase where the network controller (NC) bootstraps the keying materials and the registration of network users with the certificate authority (CA). The CA is a trusted third party that helps the NC to store and regulate access to decryption keys. This is followed by data publishing phase that involves advertising, generation, encryption, and transfer of the encrypted data to the gateway. Finally, the event data is delivered to the subscribers.

Initialization: In this phase, the NC initially selects an elliptic curve E defined over a finite field $GF(p)$ of size p, where p is a large prime number and a system base point G ($G \neq O$, where O is a point at infinity) of order q, where q is also a large prime number on that curve. The NC then generates n secret keys $x_1, x_2, \cdots, x_n$ from $GF(p)$ to compute the master secret key $X = (x_1, x_2, \cdots, x_n)$. The n corresponding public keys are then generated to make up the master public key $Y = (y_1, y_2, \cdots, y_n)$, where $y_i = x_i G$, $1 \leq i < n$. Finally, the NC selects a collision resistant one-way hash function $h: \{0,1\}^* \rightarrow \{0,1\}^n$. The NC then preloads the public domain parameters (Y, G, p, q, h) into every sensor node. The NC then register the master secret key $X = (x_1, x_2, \cdots, x_n)$ with the CA.

Every network user will register with the CA through submitting its information. The CA will verify the identities of each user upon registration. Thereafter, every user will deliver its payment information and the targeted access privileges to the NC including verification evidence from the CA. Upon receiving this message, the NC will put each validated network user into a group according to its access privilege.

Data Publishing: We use BFs to encode a user's interest and the brokers only query BFs to match contents. This approach not only saves bandwidth required for interest transmission, but also protects the secrecy of a user's interest. When a sensor node generates data, d, it first computes a public key based on the attributes α, of the data, and then it encrypts the data as follows: compute public key $y_\alpha = \sum_{i=1}^{n} h_i(\alpha) y_i$, and then compute $c = Encrypt(d, y_\alpha)$.

Data Notification: When a subscriber wants to receive its requested data, it first contacts either the NC or the CA to obtain decryption key. After the subscriber is authenticated, the NC or CA will run $x_\alpha = keygen(\alpha)$ to obtain the corresponding private key x_α to decrypt the data. The subscriber then executes $d = Decrypt(x_\alpha, c)$ to obtain the data.

Security Analysis: The security of this scheme is due to the intractability of the discrete logarithm problem. The encryption of the data prevents eavesdropper from learning anything from the ciphertext. When an adversary compromised a sensor node and has access to key materials preloaded, it cannot decrypt the already encrypted data. This is because the sensor node is preloaded with only public keys without the corresponding private keys. A subscriber cannot have access to a decryption key for the event data it did not subscribe to. This is because, each subscriber must be authenticated with respect to its access privilege before a decryption key is issued.

V. Conclusion

In this paper, we presented a secure publish-subscribe-based in-network data storage service in WSNs. We described a reference communication model based on the prototype and scenario for which we provided security. We presented a key agreement scheme that guarantees data confidentiality and enforces access control. We evaluated the security of the scheme, which shows the scheme is protected against eavesdropping and node compromise attacks.

Acknowledgments

This work is supported by the National Natural Science Foundation of China under grant numbers 61073037 and 61103035, and the Ministry of Education Fund for Doctoral Disciplines in Higher Education under grant number 20110162110043.

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, "Wireless Sensor Networks: A Survey," *Computer Networks: The International Journal of Computer and Telecommunications Networking*, 38(4): 393-422, 2002.

[2] S. Krishnamurthy and L. Lange, "Enabling Distributed Messaging with Wireless Sensor Nodes Using TinySIP," *Proc. of the Ubiquitous Intelligence and Computing* (UIC 2007), LNCS 4611, J. Indulska et al. (Eds.), 610-621, 2007.

[3] B. H. Bloom, "Space/Time Trade-offs in Hash Coding with Allowable Errors," *Communications of the ACM*, 13(7): 422-426, 1970.

[4] D. Boneh and M. Franklin, "Identity-Based Encryption from the Weil Pairing," *Proc. of CRYPTO*, 213-229, 2001.

[5] P. T. Eugster, P. A. Felber, R. Guerraoui, and A.-M. Kernmarrec, "The Many Faces of Publish/Subscribe," *ACM Computing Surveys*, 35(2): 114-131, 2003.

[6] U. Hunkeler, H. L. Truong, and A. Stanford-Clark, "MQTT-S - A Publish/Subscribe Protocol for Wireless Sensor Networks," *Proc. of the 3rd International Conference on Communication Systems Software and Middleware and Workshops* (COMSWARE 08), 791-798, 2008.

[7] S. Krishnamurthy, "TinySIP: Providing Seamless Access to Sensor-Based Services," *Proc. of the 1st International Workshop on Advances in Sensor Networks* (IWASN 2006), 2006.

[8] C. C. Tan, H. Wang, S. Zhong, and Q. Li, "IBE-Lite: A Lightweight Identity-Based Cryptography for Body Sensor Networks," *IEEE Transactions on Information Technology in Biomedicine*, 13(6): 926-932, 2009.