2008

A Security Framework for Wireless Sensor Networks Utilizing a Unique Session Key

Yong Wang
University of Nebraska - Lincoln, ywang@cse.unl.edu

Byrav Ramamurthy
University of Nebraska - Lincoln, bramamurthy2@unl.edu

Yuyan Xue
University of Nebraska - Lincoln, yxue@cse.unl.edu

Xukai Zou
Indiana University-Purdue University Indianapolis, xkzou@cs.iupui.edu

BROADNETS 2008. 5th International Conference on Broadband Communications, Networks and Systems, 2008.
Digital Object Identifier: 10.1109/BROADNETS.2008.4769130 Publication Year: 2008 , Page(s): 487 - 494

A Security Framework for Wireless Sensor Networks Utilizing a Unique Session Key

Yong Wang, Byrav Ramamurthy, Yuyan Xue
Department of Computer Science and Engineering
University of Nebraska-Lincoln
Lincoln, NE 68588 USA
Email: {ywang, byrav, yxue}@cse.unl.edu

Xukai Zou
Department of Computer and Information Science
Indiana University-Purdue University Indianapolis
Indianapolis, IN 46202 USA
Email: xkzou@cs.iupui.edu

***Abstract*— Key management is a core mechanism to ensure the security of applications and network services in wireless sensor networks. It includes two aspects: key distribution and key revocation. Many key management protocols have been specifically designed for wireless sensor networks. However, most of the key management protocols focus on the establishment of the required keys or the removal of the compromised keys. The design of these key management protocols does not consider the support of higher level security applications. When the applications are integrated later in sensor networks, new mechanisms must be designed. In this paper, we propose a security framework, uKeying, for wireless sensor networks. This framework can be easily extended to support many security applications. It includes three components: a security mechanism to provide secrecy for communications in sensor networks, an efficient session key distribution scheme, and a centralized key revocation scheme. The proposed framework does not depend on a specific key distribution scheme and can be used to support many security applications, such as secure group communications. Our analysis shows that the framework is secure, efficient, and extensible. The simulation and results also reveal for the first time that a centralized key revocation scheme can also attain a high efficiency.**

I. Introduction

Wireless sensor networks (WSNs) are promising solutions for many applications and security is an essential requirement of WSNs. Among all security issues in WSNs, key management is a core mechanism to ensure the security of applications and network services in WSNs [1]. It includes two aspects: key distribution and key revocation. The goal of the key distribution is to establish the required keys between sensor nodes which must exchange data. Key revocation is used to remove compromised sensor nodes from the network.

Many key management protocols have been specifically designed for wireless sensor networks. However, most of the key management protocols focus on the establishment of the required keys or the removal of the compromised keys. The design of these schemes does not consider the support of higher level security applications. When these applications are later integrated in sensor networks, new mechanisms must be designed. For example, secure group communication (SGC) is an important application and it is also very useful in wireless sensor networks [2]. Although the secure group communication problem has been extensively studied in the context of secure multicast in wired networks, most of them are not suitable for WSNs due to the constraints on the sensor nodes. A few papers [3], [4] address the secure group communication problem in the context of sensor networks. In these papers, new group rekeying schemes are designed in addition to the key management protocols in the sensor networks. In the literature there is a lack of a security framework which provides an integrated solution for key management and higher level security applications for sensor networks.

In this paper, we propose a security framework, uKeying, for wireless sensor networks. The proposed framework includes three components: a security mechanisms to provide confidentiality, authentication, and integrity for the communication, an efficient session key distribution scheme, and a centralized key revocation scheme. The security of the communication is ensured by two types of keys: *encryption key*, and *message authentication code (MAC) key*. Both of the two keys are bound to a globally distributed *session key*. The session key can be distributed in the network using a broadcasting message in one round. The proposed framework does not depend on a specific key distribution scheme and can be used to support many secure applications, for example, secure group communication (SGC), in wireless sensor networks.

Our contributions in this paper include the following:

1) We propose a security framework which can be easily extended to support higher level security applications in wireless sensor networks.
2) We show for the first time that centralized key revocation scheme can also attain a high efficiency in wireless sensor networks.

The remainder of the paper is organized as follows: Section II discusses the related work. Section III introduces our proposed key management framework. The security and performance analyses are presented in Section IV, and followed by the simulation experiments and results in Section V. Section VI concludes the paper.

II. Related work

As discussed earlier, key management includes two aspects: key distribution and key revocation. Many key distribution schemes have been proposed in sensor networks. According to the network structure, the schemes can be divided into centralized key distribution schemes [5] and distributed key distribution schemes [6], [7]. According to the probability of key sharing between a pair of sensor nodes, the key distribution schemes can be classified into deterministic approaches [8], [9]

and probabilistic approaches [6], [7]. An investigation of key distribution schemes for WSNs can be found in [1].

In this paper, we propose a security framework, uKeying, for wireless sensor networks. uKeying does not depend on a specific key distribution scheme as long as the key distribution scheme provides pairwise keys for sensor nodes which must exchange data. The establishment of the pairwise keys among sensor nodes is one of the main tasks for a key distribution scheme and has been extensively studied in the literature. For example, the key distribution scheme in [6] consists of three phases: key pre-distribution, shared-key discovery, and path key establishment. In the key pre-distribution phase, each sensor is equipped with a *key ring* held in the memory. The key ring consists of *k* keys which are randomly drawn from a large pool of *P* keys. In the shared key discovery phase, each sensor discovers its neighbors within its wireless communication range with which it shares keys. Finally, in the path-key establishment phase, a path-key is assigned between sensor nodes which are within wireless communication range but do not share a key at the end of the second phase.

Key revocation refers to the task of securely removing keys which are known to be compromised. To detect a compromised sensor, intrusion detection techniques are employed. Intrusion detection is out of the scope of this paper. We assume that there are some methods [10], [11], [12] which can be used to detect a compromised sensor node. Recent work conducted on key revocation for WSNs include [6], [13], [14], [15], [16]. These key revocation schemes can be divided into two categories: the centralized key revocation schemes, such as EsRev scheme [6], GPSRRev scheme [16], and the distributed key revocation schemes, DistRev scheme [13], [14].

Although a few schemes [6], [14] have been proposed to address the key revocation problem in WSNs, these schemes encounter various difficulties when used in sensor networks. For example, the EsRev scheme proposed in [6] requires a signature key to be distributed to the non-revoked sensor nodes. However, the signature key can only be distributed by unicasting which causes severe performance issues in large scale sensor networks. In GPSRRev scheme [16], the revocation area is divided into sub-areas. For each sub-area, a revocation message is sent to a certain node within that area using GPSR protocol [17], and then the revocation message is multicasted to the remaining sub-area. However, additional information, such as location of the sensor nodes, must be used. Further, the multicast of the revocation message in the sub-area is implemented using message flooding and it is still time and energy consuming. The distributed key revocation scheme, DistRev, proposed in [14] is based on some simple assumptions such as each node knowing its neighboring nodes before the sensor network is deployed.

In [15], we proposed a centralized key revocation scheme, KeyRev, for wireless sensor networks. In this paper, we utilize the KeyRev scheme to provide revocation capability for the framework and we further analyze, evaluate and compare the performance of the KeyRev scheme with that of the three other key revocation schemes.

Secure group communication is an important application in wireless sensor networks [2]. The obvious benefit of secure group communication to WSNs is that outside nodes are unable to obtain any messages transmitted to the group. Secure group communication is also attractive for in-network processing and data aggregation [18]. In [2], we proposed two SGC schemes (SGC-unicasting and SGC-broadcasting) for wireless sensor networks based on using any sensor node member as a group controller. In this paper, we further present two new secure group communication schemes for WSNs (the SGC-overlapping and the SGC-preloading schemes) using the proposed key management framework.

III. uKeying: a Security Framework for WSNs

In this section, we propose our security framework, uKeying, for wireless sensor networks. This framework includes three components, namely, a security mechanism to provide secrecy, an efficient session key distribution scheme, and a centralized key revocation scheme. We further demonstrate how to use the framework to support an important applications in WSNs, secure group communication.

A. A security mechanism to provide secrecy

The lifetime of a WSN is partitioned into time intervals called $sessions$. The duration of sessions can be fixed or dynamic depending on the applications. The base station is responsible for distributing *session keys* to the sensor nodes. We use K_j to denote the j-th session key where $j \in \{1, 2, \cdots, m\}$ and m is the number of sessions. We assume that each sensor is uniquely identified by an ID number i, where $i \in \{1, \ldots, n\}$ and n is the largest ID number.

We use two kinds of keys for secure communication in the sensor network: the *encryption key*, K_{encr} and the *message authentication code (MAC) key*, K_{mac}. For any message transmitted in the network, authentication, confidentiality, and integration are required. Let A and B be two entities in a WSN, the complete message A sends to B is:

$$A \longrightarrow B : E(K_{encr}, M||T_s), MAC(K_{mac}, \{M||T_s\}_{K_{encr}})$$

where M is the message, T_s is the timestamp when sending the message, $E(K, R)$ denotes the encryption of the message R with key K, and $MAC(K, R)$ denotes the computation of the message authentication code of message R with key K.

Let K_j be the current session key and $K_{A,B}$ represent the pairwise key shared between the entity A and the entity B. The encryption key and the MAC key used in session j can be generated as follows:

$$K_{encr} = F(MAC(K_{A,B}, K_j), 1)$$
$$K_{mac} = F(MAC(K_{A,B}, K_j), 2)$$

where $F(K, x)$ is a pseudo-random function and x is an integer 1 or 2 for generating K_{encr} or K_{mac} respectively.

The security of the communication between A and B is ensured by the encryption key, K_{encr} and the MAC key, K_{mac}. Any message that A sends to B is encrypted by the encryption

key K_{encr} and signed by the MAC key K_{mac}. For any message that B receives from A, B always verifies the message first and then decrypts it. Further, a sensor node always uses the encryption key and the MAC key corresponding to the current session key to encrypt and sign the outgoing messages or decrypt and verify the incoming messages.

The pairwise key between A and B does not depend on a specific key distribution scheme and thus, the framework can be extended for implementation with other key distribution schemes. Next, we introduce our session key distribution scheme.

B. Session key distribution scheme

The session key distribution is based on the personal key share distribution scheme in [19]. It can be divided into three phases, viz., setup, broadcast, and session key recovery.

1) Setup: The setup server randomly picks m $2t$-degree masking polynomials, $h_j(x) = h_{j,0} + h_{j,1}x + \cdots + h_{j,2t}x^{2t}, j \in \{1, 2, \cdots, m\}$, over a finite field F_q where q is a sufficiently large prime number. For each sensor node A_i, the setup server loads the personal secrets, $\{h_1(i), h_2(i), \cdots, h_m(i)\}$, on to the node A. The setup server also loads the polynomial, $h_j(x)$, on to the base station. For each session key K_j, the setup server randomly picks a t-degree polynomial $p_j(x)$ and constructs $q_j(x) = K_j - p_j(x)$.
2) Broadcast: Given a set of revoked sensor nodes, $R = \{r_1, r_2, \cdots, r_w\}$, $w \leq t$ in session j, the base station distributes the shares of t-degree polynomial $p_j(x)$ and $q_j(x)$ to non-revoked sensors via the following broadcast message:

$$\begin{aligned} B &= \{R\} \\ &\cup \{P_j(x) = g_j(x)p_j(x) + h_j(x)\} \\ &\cup \{Q_j(x) = g_j(x)q_j(x) + h_j(x)\} \end{aligned} \quad (1)$$

where the revocation polynomial $g_j(x)$ is constructed as $g_j(x) = (x - r_1)(x - r_2)\cdots(x - r_w)$. The authenticity of the broadcast message is ensured by broadcast authentication schemes such as $\mu Tesla$ [20].
3) Session key recovery: If any non-revoked sensor node A_i receives such a broadcast message, it evaluates the polynomial $P_j(x)$ and $Q_j(x)$ at point i and gets $P_j(i) = g_j(i)p_j(i) + h_j(i)$ and $Q_j(i) = g_j(i)q_j(i) + h_j(i)$. Because A_i knows $h_j(i)$ and $g_j(i) \neq 0$, it can compute $p_j(i) = \frac{P_j(i) - h_j(i)}{g_j(i)}$ and $q_j(i) = \frac{Q_j(i) - h_j(i)}{g_j(i)}$. A_i can then compute the new session key $K_j = p_j(i) + q_j(i)$. The revoked sensors cannot recover $p_j(i)$ and $q_j(i)$ because $g_j(i) = 0$ and thus they cannot compute the new session key.

Since the communication among sensor nodes depends on their processing the correct session key, the sensor network must be synchronized to use the same session key.

C. A centralized key revocation scheme

We proposed an efficient key revocation scheme, KeyRev, for wireless sensor networks and evaluated its performance against other centralized key revocation schemes in [15]. The KeyRev scheme can also be integrated in the proposed framework. We consider two situations here:

1) In case there are no compromised sensors in the network, the base station selects a constant $c, c \notin \{1, \cdots, n\}$, and adds c to the revocation list such as $R = \{c\}$. Then, the base station broadcasts the message as shown in Equation 1.
2) In case a set of sensor nodes $\{r_1, r_2, \cdots, r_w\}$ are compromised, the base station sets $R = \{r_1, r_2, \cdots, r_w\}$ and broadcasts the message as shown in Equation 1.

Without obtaining the new session key, the compromised sensor cannot derive the encryption key, K_{encr} and the MAC key, K_{mac} and thus cannot decrypt new messages and authenticate itself to other sensor nodes in the network. The compromised sensor nodes can thus be removed from the network.

To demonstrate the session key distribution process and the key revocation scheme, an example is given below. We consider three sensors with ID numbers 1, 2, and 3 respectively. We assume sensor 2 is compromised in session 5 and will be revoked in session 6. In the setup phase, the setup server picks the masking polynomial $h_6(x) = 1 + x^8$ for session 6 and each sensor receives a secret $h_6(1) = 2$, $h_6(2) = 257$, and $h_6(3) = 6562$ respectively. Let $K_6 = 101$, $p_6(x) = 1 + x^4$ and thus we have $q_6(x) = 100 - x^4$ and $g_6(x) = x - 2$. In session 6, the base station broadcasts a message:

$$\begin{aligned} B &= \{2\} \\ &\cup \{P_6(x) = (x-2)(1+x^4) + 1 + x^8\} \\ &\cup \{Q_6(x) = (x-2)(100-x^4) + 1 + x^8\} \end{aligned}$$

When sensor 1 receives the message, sensor 1 calculates: $P_6(1) = 0$, $Q_6(1) = -97$ and thus $p_6(1) = 2$ and $q_6(1) = 99$. Sensor 1 computes the session key $K_6 = p_6(1) + q_6(1) = 101$; Similarly, sensor 3 calculates: $P_6(3) = 6644$, $Q_6(3) = 6581$ and thus $p_6(3) = 82$ and $q_6(3) = 19$. Sensor 3 can also compute the session key $K_6 = p_6(3) + q_6(3) = 101$. However, sensor 2 cannot calculate $p_6(2)$ and $q_6(2)$ because $g_6(2) = 0$ and thus sensor 2 cannot derive the new session key.

D. Secure group communication

The proposed framework can be easily extended to support secure group communication using the unique session key. The group key K_G can be generated using a function $MAC(K, M)$ over two secrets, a group key share K_s, and a session key K_j:

$$K_G = MAC(K_j, K_s) \quad (2)$$

The session key distribution has been described in Section III-A. The group formation and the group key share distribution process are described below:

1) Solicitation of interest: The base station broadcasts a message soliciting expression of interest in event E to the sensor network:

$$B \longrightarrow * : gid||E$$

The authentication of the broadcast message is ensured by broadcast authentication schemes such as $\mu Tesla$ [20]. No one can impersonate a base station and broadcast an authenticated message.

2) Join: All the receivers observing the same event E send a Join Request to the base station:

$$A \longrightarrow B : ID_A||gid||E, MAC(K_{A,B}, ID_A||gid||E)$$

where $K_{A,B}$ is the pairwise key shared by the sensor node A with the base station B.

3) Group key share distribution: Once the base station authenticates the join request, the base station unicasts the group key share K_s to the sensor A using the Secret Share message:

$$B \longrightarrow A : \{K_s\}_{K_{A,B}}$$

along the routing paths set up during the transmission of the Interest and the Join Request messages (Please refer to [2] for the details).

With the group key share K_s and the current session key K_j, each group member can calculate the shared group key K_G as described in Equation 2.

The proposed scheme, which is referred to as SGC-overlapping scheme, is much simpler when a sensor node wants to leave the group. The leave operation can be reduced to the session key update problem and can be completed using one broadcast message.

Notice that the group formation phase in the SGC-overlapping scheme may take a long time due to the use of unicasting to distribute the group key share. If we know the group membership during the pre-distribution stage, we can load a group key share on to the sensor nodes before the sensor network is deployed. Thus, the group formation phase can be simplified as the distribution of the session key only. The new scheme, SGC-preloading, is thus a specific instance of the SGC-overlapping scheme.

Note that the solution of preloading a single mission key on to the sensor nodes in the group does not work. Once a sensor node is compromised, the single mission key is exposed and cannot ensure the security of the group communication. However, in the SGC-preloading scheme, although an adversary may compromise the sensor node to steal the group key share, the adversary cannot participate in the group communication until it obtains the session key. By preloading a group key share in sensor nodes before the sensor network is deployed, the SGC-preloading scheme greatly reduces the group formation time. It requires only one broadcast message to set up the group and update the group key.

IV. Security and performance analysis

In this section we first discuss the security of the framework. Then, we analyze the computation, the communication costs, and the storage requirements of the framework.

A. Security analysis

The proposed framework satisfies the following properties.

Property 1 The session key distribution process is secure.

The session key is distributed using the personal key distribution scheme [19]. To restore the session key, it is required that some personal secret be pre-distributed among the sensor nodes. Outsiders cannot recover the session key without the pre-distributed secret. Further, as we show in Section III-A, the revoked sensors cannot recover the new session keys either. Thus, the session key distribution process is secure.

Property 2 The KeyRev scheme is secure inspite of the non-removal of the pre-distributed key materials at a compromised sensor node.

Although, due to the non-removal of the pre-distributed key materials, the compromised sensor may retain the pairwise keys, the adversaries cannot figure out the encryption key, K_{encr} and the MAC key, K_{mac} if the session key is updated. In the worst case, an adversary might use a chosen plaintext attack to crack the session key; however, the attack itself is also time consuming. As long as the duration of sessions is less than the session key cracking time, the proposed key revocation scheme is secure.

B. Performance analysis

The performance of the framework depends mainly on the session key distribution process. Thus, we focus on the session key distribution scheme.

1) Computation cost: To restore the session key, each sensor node must evaluate the polynomial $P_j(x)$ and $Q_j(x)$ at point i. The polynomial evaluation is fast and thus the session key recovery is efficient in computation.

2) Communication cost: The session key can be updated in one round using broadcasting. The maximum size of the broadcast message in bits is decided by S:

$$S = (5t + 2) \log q$$

Let B indicate the transmission rate of the base station, L be the maximum range between the base station and the sensor nodes. The session key distribution time can be calculated as:

$$t_s = \frac{S}{B} + \frac{L}{3 * 10^8}$$

Compared with the transmission time, the propagation delay is very small. Thus, we can approximately estimate the session key distribution time as:

$$t_s \approx \frac{(5t + 2) \log q}{B}$$

3) Storage requirement: To restore the session key, each sensor node needs to be loaded with m personal secrets. Since the encryption key and the message authentication code key can be set up on-the-fly, the extra storage needed to implement the KeyRev scheme is $m \log q$.

C. Comparison

The proposed framework does not depend on a specific key distribution scheme. In the remainder of this section, we compare the KeyRev scheme and the secure group communication schemes with the existing schemes in the literature.

1) KeyRev: The KeyRev scheme is a centralized key revocation scheme. It depends on an efficient session key distribution scheme which can be implemented in one round using a broadcast message. In contrast, in case a sensor node is compromised, the EsRev scheme requires two rounds of communications: distributing a signature key to the non-revoked sensors, followed by broadcasting a message containing a list of revoked key identifiers. Since the signature key is distributed to the network using unicasting, the EsRev scheme may cause heavy traffic in large scale sensor networks. Note that since there is no need for the unicasting and the session key can be updated in one round using broadcasting, the KeyRev scheme performs much better than the EsRev scheme.

Although the GPSRRev scheme performs better than the EsRev scheme by dividing the revocation field into sub-areas and using multiple revocation messages, the multicast of the revocation message in the sub-area is still time and energy consuming. The KeyRev scheme is more efficient than the GPSRRev scheme since it uses broadcast instead of multicast.

The distributed key revocation scheme, DistRev, has been regarded to be faster than the centralized key revocation schemes due to the fact that it requires only broadcast messages of a few hops that reach the local destinations [14]. However, in case a sensor node is compromised and revoked successfully from the network, the DistRev scheme requires four rounds of communications as follows (from [14]):

1) Neighboring nodes exchange the masks to decrypt the votes for the current revocation sessions at the connection time.
2) At least t sensor nodes cast their votes against the target node (compromised node) in the current session.
3) The voting nodes also cast their votes against the target node on the next session.
4) If a sensor node receives at least t revocation votes, a hash value containing the compromised sensor node information needs to be broadcasted through the entire network.

Although the first three rounds of the communications are local broadcast, the last one involves a broadcast through the entire network. The broadcast message can either be flooded from the sensor node which receives t revocation votes or be forwarded to the base station and broadcasted to the network by the base station. Either way, the KeyRev scheme is much better than the DistRev scheme since it requires only one broadcast and no local communication is required. Further, the DistRev scheme is also built on some simplifying assumptions, for example, each node knows its neighboring nodes before deployment, which are hard to satisfy in many sensor network applications.

Table I compares the four revocation schemes discussed in the paper, where n is the number of sensor nodes in the network, d is the number of sub-areas in the GPSRRev scheme, and t is the number of votes which a sensor node has to collect to revoke a compromised node in the DistRev scheme. We consider the situation when a single node is compromised and revoked successfully from the network.

TABLE I
COMPARISON OF THE KEY REVOCATION SCHEMES IN WIRELESS SENSOR NETWORKS.

	Scheme	Rnds	Unicast	Broadcast	Local Broadcast	Scalability
I	EsRev	2	n	1	0	Low
	GPSRRev	1	d	0	d	Medium
	KeyRev	1	0	1	0	Good
II	DistRev	4	0	1	$2 * t$	Good

Category I denotes centralized key revocation schemes and category II denotes distributed key revocation schemes. The GPSRRev scheme requires the location information of the compromised sensor nodes.

The comparison in Table I shows that the KeyRev scheme is better than other schemes in reducing the communication overhead caused by the revocation protocol. Notice that the KeyRev scheme requires a session key to be distributed to the network during each session. The duration of the session time could be set and adjusted dynamically according to the application to reduce the background traffic in the sensor network.

2) Secure group communication: In [2], we propose two secure group communication schemes, SGC-unicasting and SGC-broadcasting for wireless sensor networks. In this paper, we present two other secure group communication schemes, SGC-overlapping and SGC-preloading for wireless sensor networks.

In SGC-unicasting and SGC-broadcasting schemes, the group controller is a sensor node. Thus, the sensor node must be granted local broadcast capability to start a group. However, the capability can be abused by an adversary to start inside flooding attacks. Further, since the group controller needs to broadcast a message to its neighboring nodes to request expression of interest, the authenticity of the local broadcast message must be ensured to avoid the impersonation of the group controller. However, most of proposed broadcast authentication scheme such as [21], [22] are used for authenticating broadcast messages from the base station and cannot be used for local broadcast. Without an efficient local broadcast authentication scheme, the sensor nodes must be trustworthy in Scheme-unicasting and Scheme-broadcasting. In case a sensor node in the group is compromised, the adversary can know the group keys which it possesses. If the compromised sensor node is the group controller, the compromised sensor node may start revocation attacks to remove non-revoked sensor nodes from the group.

In SGC-overlapping and SGC-preloading schemes, the base station is the group controller to initialize and start the group. Thus, it is not necessary to grant the local broadcast capability to sensor nodes. The authenticity of the broadcast message can be ensured by schemes such as [21], [22]. Since the base

station is trustworthy, no adversary can impersonate the base station and start the revocation attack either. In case a sensor node in the group is compromised, the adversary may steal the group key and eavesdrop on the communication until the compromised sensor node is removed from the network.

The communication overhead of the proposed schemes is compared in Table II. Since the SGC-unicasting scheme depends on a key tree to distribute the group key, the scheme must maintain the integrity of the key tree when sensors join or leave the group. With the SGC-broadcasting, the SGC-overlapping and the SGC-preloading schemes, the distribution of the group key does not depend on the key tree. Thus, it is more easier to handle the join and the leave operations.

V. Simulation Experiments and Results

We evaluated the performance of the KeyRev scheme and the secure group communication schemes in SENSIM, a component-based discrete-event simulator for sensor networks. We consider two sensor network experimental settings: a small-scale sensor network with 100 nodes uniformly dispersed in a field with dimensions $100m \times 100m$ and a large-scale sensor network with 1000 nodes uniformly dispersed in a field with dimensions $2000m \times 2000m$. In both the networks, we set the base station at the center of the field and we assume that all the sensor nodes are within reach of the base station.

A. KeyRev scheme

We compare the KeyRev scheme with the centralized key revocation schemes, the EsRev scheme and the GPSRRev scheme, and the distributed key revocation scheme, the DistRev scheme. The evaluation metrics include the key revocation time t_v and the average energy consumption e_v per node to revoke a compromised sensor in the network. The key revocation time is the time duration from when the key revocation protocol starts until all the uncompromised sensor nodes receive the key revocation message.

Table III shows the key revocation time to revoke a compromised sensor node in the two networks. As the table shows, in the 100-node sensor network, the key revocation times using the EsRev scheme and the GPSRRev scheme are about 83 times and 1.6 times that of the KeyRev scheme. In the 1000-node sensor network, the key revocation times using the EsRev scheme and the GPSRRev scheme are 800 times and 6.5 times that of the KeyRev scheme. The KeyRev scheme is much better than the EsRev scheme and the GPSRRev scheme in terms of the key revocation time.

Table IV shows the average energy consumption to revoke a compromised sensor in the 100-node and 1000-node sensor networks. As the table shows, in the 100-node sensor network, the average energy consumption to revoke a single node using the EsRev scheme and the GPSRRev are about 71 times and 19 times that of the KeyRev scheme. In the 1000-node sensor network, the average energy consumption to revoke a single sensor using the EsRev scheme and the GPSRRev scheme are about 714 times and 29 times that of the KeyRev scheme. The KeyRev scheme is much better than the EsRev scheme and the GPSRRev scheme in terms of the average energy consumption.

In both the experimental settings, the KeyRev scheme performs very well compared with the EsRev scheme and the GPSRRev scheme. Further, Tables III and IV also show that the key revocation time and the average energy consumption to revoke a single sensor node by using the KeyRev scheme have only a slight difference between the 100-node sensor network and the 1000-node sensor network, which indicates that the KeyRev scheme is scalable to large-scale sensor networks. However, due to the long key revocation delay caused by the EsRev scheme, the EsRev scheme is not scalable to large-scale sensor networks. The performance of the GPSRRev scheme is better than the EsRev scheme but not as good as that of the KeyRev scheme.

TABLE III
KEY REVOCATION TIME.

Scheme	100-node WSN Time (seconds)	1000-node WSN Time (seconds)
EsRev	49.63	496.06
GPSRRev	1.02	4.04
KeyRev	0.59	0.62

TABLE IV
AVERAGE ENERGY CONSUMPTION PER NODE TO REVOKE A COMPROMISED SENSOR.

Scheme	100-node WSN Energy (joules)	1000-node WSN Energy (joules)
EsRev	0.71	7.14
GPSRRev	0.19	0.29
KeyRev	0.01	0.01

To evaluate the performance of the KeyRev scheme, we also compare the KeyRev scheme with the DistRev scheme. The metrics we evaluate include the key revocation time and the average energy consumption. Each revocation session in the DistRev scheme consists of three states: pending, active, and completed. The critical part of the three states which decides the key revocation time is the active state. In the active state, a sensor node casts a vote and the vote is broadcasted locally among the neighboring nodes. Assume that the active state lasts for Δ_s time for each node and Δ_c is the maximum time that a message needs to completely propagate in a local neighborhood broadcast. We have $t_v > \Delta_s$ and $\Delta_s > 2\Delta_c$ since each sensor has to vote both in the current session and in the next session. Therefore, the key revocation time t_v of the DistRev scheme is at least twice that of Δ_c, thus $t_v > 2\Delta_c$. Similarly, let e_{Δ_s} be the energy consumption during the active state and e_{Δ_c} be the energy consumption consumed during the Δ_c period of time, We have $e_v > e_{\Delta_s}$, $e_{\Delta_s} > te_{\Delta_c}$ (to revoke a compromised sensor node, the sensor node must receive at least t revocation votes) and thus, $e_v > te_{\Delta_c}$.

The duration of Δ_c is decided by a maximum count L (max-hops), over which the vote can be broadcasted to ensure complete dissemination in the neighborhood of a compromised sensor node (four-six hops can cover this area with high probability [6]). We test the Δ_c in the 100-node and the 1000-

TABLE II
COMPARISON.

		SGC-unicasting		SGC-broadcasting		SGC-overlapping		SGC-preloading	
	message	nums	size	nums	size	nums	size	nums	size
Group formation	unicast	$2n$	$O(\log q)$	$2n$	$O(\log q)$	$2n$	$O(\log q)$	0	n/a
	broadcast	1	$O(\log q)$	2	$O(t \log q)$	2	$O(t \log q)$	1	$O(t \log q)$
Group key update	unicast	n	$O(\log q)$	0	n/a	0	n/a	0	n/a
	broadcast	0	n/a	1	$O(t \log q)$	1	$O(t \log q)$	1	$O(t \log q)$

node sensor networks. The sensor node casting the vote is set to the center of each testbed. Table V shows the number of sensor nodes in the coverage area when the max-hops changes.

TABLE V
THE NUMBER OF NODES IN THE COVERED AREA.

L (max-hops)	1	2	3	4	5	6
100-node WSN	100	n/a	n/a	n/a	n/a	n/a
1000-node WSN	15	44	85	142	219	299

Note: All the sensor nodes in the 100-node sensor network are in the cover area when the max-hops is set to 1.

In the 100-node sensor network, the simulation results show that $\Delta_c = 0.035$ seconds and $e_{\Delta_c} = 995$ nano-joules. Thus, we have $t_v > 0.070$ and $e_v > 995t$ nano-joules. Compared with the KeyRev scheme in the 100-node sensor network as shown in Tables III and IV, the DistRev scheme might be better than the KeyRev scheme but the performance of the KeyRev scheme is also very good in the 100-node sensor network.

Figure 1 shows the key revocation time of the DistRev scheme in the 1000-node sensor network when the max-hops changes. Note that the column value is not the real key revocation time t_v of the DistRev scheme but the value of the $2\Delta_c$. The actual key revocation time is $t_v > 2\Delta_c$. The dotted horizontal line shows the key revocation time of the KeyRev scheme in the 1000-node sensor network. From the figure, we can draw the conclusion that the KeyRev scheme is better than the DistRev scheme in terms of the key revocation time since the max-hops is definitely greater than one in the DistRev scheme to ensure full coverage of the neighboring nodes of the target node (compromised node).

Figure 2 shows the average energy consumption per node in the DistRev scheme in the 1000-node sensor network when the max-hops changes. The column value is also not the real average energy consumption e_v of the DistRev scheme but the value of $2e_{\Delta_c}$ (we set t to the minimum value 2, $t = 2$). The actual average energy consumption is $e_v > te_{\Delta_c}$. The dotted horizontal line shows the average energy consumption of the KeyRev scheme in the 1000-node sensor network. The figure indicates that the KeyRev scheme is better than the DistRev scheme even if we set the number of votes to revoke a sensor node to the minimum value of two.

To ensure that the neighborhood of the target node (compromised node) is fully covered, the max-hops cannot be set too small. Thus, our proposed scheme, KeyRev, is better than the DistRev scheme. From Figures 1 and 2, we can estimate the performance of the KeyRev scheme and the DistRev scheme. For example, if the max-hops is set to five, the key revocation

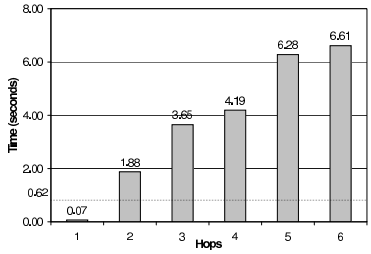

Fig. 1. Key revocation time in the 1000-node sensor network. The column value is not the real key revocation time t_v of the DistRev scheme but the value of the $2\Delta_c$.

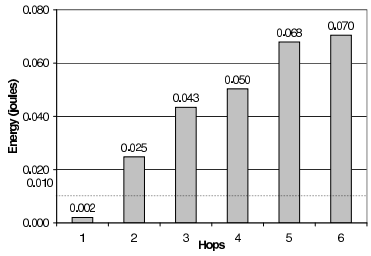

Fig. 2. Average energy consumption per node to revoke a compromised sensor in the 1000-node sensor network. The column value is also not the real average energy consumption e_v of the DistRev scheme but the value of $2e_{\Delta_c}$.

time of the DistRev scheme is at least 10.1 times that of the KeyRev scheme and the average energy consumption of the DistRev scheme is at least 6.8 times that of the KeyRev scheme.

Overall, the KeyRev scheme is much better than the previously proposed centralized key revocation schemes, such as the EsRev scheme and the GPSRRev scheme. It is also superior to the distributed key revocation scheme, the DistRev scheme.

B. Secure group communication schemes

We also evaluate the performance of the two SGC schemes (SGC-overlapping and SGC-preloading) with the schemes (SGC-unicasting and SGC-broadcasting) proposed in [2]. Due to the page limitations of this paper, we briefly introduce our simulation and results here. Please refer to [23] for the details.

We test the four schemes for different group sizes in a wireless sensor networks. The group size is decided by a maximum count (max-hops) along the routes in which the interest message is forwarded and we assume that all sensor nodes which hear the message become group members. We use the same group of sensor nodes for all the four schemes in each test scenario.

We evaluate the group formation time, the group key update time for each of the four schemes. We also evaluate the average

group member energy consumptions and the average group controller energy consumptions during the group formation phase and the group key update phases. Our simulation results indicate that, with respect to the group formation time and the energy consumption at the group controller and the group member sensor nodes, the SGC-unicasting scheme performs slightly better than the other schemes in the group formation phase; however, the SGC-broadcasting, the SGC-overlapping, and the SGC-preloading schemes perform far better than the SGC-unicasting scheme in the group key update phase. Our simulation also shows that the energy is distributed more evenly in the SGC-broadcasting and the SGC-overlapping schemes in both the group formation and group update phases. To summarize, if the group is set up on-the-fly, the SGC-overlapping scheme is a better solution for secure group communication in sensor networks. If the group membership can be decided in the pre-distribution stage, the SGC-preloading scheme is the best solution for secure group communication in sensor networks.

Note that the performance of the SGC-unicasting scheme represents that of a general category of secure group communication schemes using a key tree structure.

VI. Conclusion and Future Work

In this paper, we propose a security framework, uKeying, for wireless sensor networks utilizing a globally distributed session key. uKeying does not depend on a specific key distribution scheme and can be used to support many other security applications. Our analysis shows that the framework is secure, efficient, and extensible. The simulation and results also reveal for the first time that a centralized key revocation scheme can also attain a high efficiency.

uKeying depends on a globally distributed session key in the network, which requires that the sensor network be synchronized. Since most broadcast authentication schemes, such as $\mu Tesla$, require the synchronization of all sensor nodes in the network, it is not a problem if such broadcast authentication schemes are used. Our future work will extend the framework to scenarios where the sensor network is not synchronized. Further investigation will also be conducted to include more secure applications in the framework.

References

[1] Y. Wang, G. Attebury, and B. Ramamurthy, "A survey of security issues in wireless sensor networks," *IEEE Communications Surveys and Tutorials*, vol. 8, no. 2, 2006.

[2] Y. Wang and B. Ramamurthy, "Group rekeying schemes for secure group communication in wireless sensor networks," in *ICC '07: Proceedings of IEEE International Conference on Communications*, Glasgow, Scotland, U.K., June 2007.

[3] N. Thepvilojanapong, Y. Tobe, and K. Sezaki, "A proposal of secure group communication for wireless sensor networks," in *The 23th Computer Security (CSEC) Group Meeting, IPSJ*, Tokyo, Japan, Dec. 2003, pp. 47–52.

[4] M. Tubaishat, J. Yin, B. Panja, and S. Madria, "A secure hierarchical model for sensor network," *SIGMOD Rec.*, vol. 33, no. 1, pp. 7–13, 2004.

[5] R. D. Pietro, L. V. Mancini, Y. W. Law, S. Etalle, and P. J. M. Havinga, "LKHW: A directed Diffusion-Based secure multicast scheme for wireless sensor networks," in *ICPPW '03: Proceedings of the 32nd International Conference on Parallel Processing Workshops*. IEEE Computer Society Press, 2003, pp. 397–406.

[6] L. Eschenauer and V. D. Gligor, "A key-management scheme for distributed sensor networks," in *CCS '02: Proceedings of the 9th ACM conference on Computer and communications security*. New York, NY, USA: ACM Press, 2002, pp. 41–47.

[7] W. Du, J. Deng, Y. S. Han, S. Chen, and P. K. Varshney, "A key management scheme for wireless sensor networks using deployment knowledge," in *Proceedings of IEEE INFOCOM*, 2004, pp. 586–597.

[8] S. Zhu, S. Setia, and S. Jajodia, "LEAP: efficient security mechanisms for large-scale distributed sensor networks," in *CCS '03: Proceedings of the 10th ACM conference on Computer and communications security*. New York, NY, USA: ACM Press, 2003, pp. 62–72.

[9] S. A. Cametepe and B. Yener, "Combinatorial design of key distribution mechanisms for wireless sensor networks," in *Proceedings of 9th European Symposium on Research Computer Security*, Sophia Antipolis, France, 2004.

[10] S. Zhu, S. Setia, S. Jajodia, and P. Ning, "An interleaved hop-by-hop authentication scheme for filtering of injected false data in sensor networks," in *Proceedings of IEEE Symposium on Security and Privacy*, Oakland, CA, USA, May 2004, pp. 259–271.

[11] F. Ye, H. Luo, S. Lu, and L. Zhang, "Statistical en-route filtering of injected false data in sensor networks," in *Proceedings of IEEE INFOCOM*, 2004.

[12] G. Wang, W. Zhang, C. Cao, and T. L. Porta, "On supporting distributed collaboration in sensor networks," in *Proceedings of MILCOM*, 2003.

[13] H. Chan, A. Perrig, and D. Song, "Random key predistribution schemes for sensor networks," in *Proceedings of IEEE Symposium on Security and Privacy*, Oakland, CA, USA, May 2003.

[14] H. Chan, V. Gligor, A. Perrig, and G. Muralidharan, "On the distribution and revocation of cryptographic keys in sensor networks," *IEEE Transactions on Dependable and Secure Computing*, vol. 2, no. 3, pp. 233–247, July-Sept. 2005.

[15] Y. Wang, B. Ramamurthy, and X. Zou, "KeyRev: An efficient key revocation scheme for wireless sensor networks," in *ICC '07: Proceedings of IEEE International Conference on Communications*, Glasgow, Scotland, U.K., June 2007.

[16] W. Zhang, H. Song, S. Zhu, and G. Cao, "Least privilege and privilege deprivation: towards tolerating mobile sink compromises in wireless sensor networks," in *MobiHoc '05: Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing*. New York, NY, USA: ACM Press, 2005, pp. 378–389.

[17] B. Karp and H. T. Kung, "GPSR: greedy perimeter stateless routing for wireless networks," in *MobiCom '00: Proceedings of the 6th annual international conference on Mobile computing and networking*. New York, NY, USA: ACM Press, 2000, pp. 243–254.

[18] J.-H. Huang, J. Buckingham, and R. Han, "A level key infrastructure for secure and efficient group communication in wireless sensor network," in *SECURECOMM '05: Proceedings of the First International Conference on Security and Privacy for Emerging Areas in Communications Networks (SECURECOMM'05)*. Washington, DC, USA: IEEE Computer Society, 2005, pp. 249–260.

[19] D. Liu, P. Ning, and K. Sun, "Efficient self-healing group key distribution with revocation capability," in *CCS '03: Proceedings of the 10th ACM conference on Computer and communications security*. New York, NY, USA: ACM Press, 2003, pp. 231–240.

[20] A. Perrig, R. Szewczyk, V. Wen, D. Culler, and J. D. Tygar, "SPINS: Security protocols for sensor networks," *Wireless Networks*, vol. 8, no. 5, pp. 521–534, September 2002.

[21] B. Przydatek, D. Song, and A. Perrig, "SIA: secure information aggregation in sensor networks," in *SenSys '03: Proceedings of the 1st international conference on Embedded networked sensor systems*. New York, NY, USA: ACM Press, 2003, pp. 255–265.

[22] D. Liu and P. Ning, "Multi-level μTESLA: Broadcast authentication for distributed sensor networks," *Trans. on Embedded Computing Sys.*, vol. 3, no. 4, pp. 800–836, 2004.

[23] Y. Wang, "Key management protocols in hybrid wireless sensor networks," Ph.D. dissertation, University of Nebraska-Lincoln, December 2007.