A Mobile Object-based Secret Key Distribution Scheme for Wireless Sensor Networks

Pardeep Kumar[1], Pawani Porambage[1]
[1]Centre for Wireless Communication
University of Oulu, Finland
pradeepkhl@gmail.com, pporamba@ee.oulu.fi

Mika Ylianttila[1], Andrei Gurtov[2]
[2]Computer Science and Engineering
Aalto University, Finland
mika.ylianttila@ee.oulu.fi, gurtov@hiit.fi

***Abstract*— Security is a paramount requirement in any modern technology. This also applies to wireless sensor networks (WSNs), where sensor nodes have severe resources scarcity. Recently the presence of mobile object (e.g., moving robot, vehicle, etc) has shown their great impact on WSNs. However, as the low-cost sensor networks become wide-spread, secret key distribution to the sensor nodes is a challenging task in such unattended WSNs. In this regards, this paper presents a mobile object-based secret key distribution scheme for resource hungry sensor nodes. The key idea is that a mobile object should visit all the sensor nodes along its pre-defined path and distribute secret keys within its broadcasting range. The proposed scheme exploits the symmetric cryptographic approach. The feasibility of proposed scheme is advocated using real testbed experiments. The analysis reveals that the proposed scheme attains high efficiency (in terms of computation and communication costs), and provides strong security.**

Keywords*— Authentication, key distribution, mobile object, and wireless sensor networks

I. INTRODUCTION

Wireless sensor networks (WSNs) are collections of distributed tiny-nodes, which are resource constraint in nature. Tiny sensors are used to monitor given field of interests and collectively sense the various environmental parameters and relay sensed data to a sink device. In general, most of traditional WSNs are stationary wireless sensor networks (SWSNs). Recently, the involvement of moving objects (moving robots, vehicle, animal, etc) have shown that how a mobile object can potentially improve the network performance in terms of network lifetime, network coverage and network connectivity, and manage the resources at sensor side [1][2]. Moreover, numbers of mobile object-based research interests have been shown in [3]-[6]. In [3] authors have proposed a network-assisted sink navigation (i.e., mobile sink) scheme that collects data from static nodes using a single hop communication. *Mikhaylov and Tervonen* have suggested and evaluated different approaches for collecting the data from isolated (static) sensors using a mobile ferry [4]. Using a mobile object, a different (sensor) data collection approach has been proposed in urban scenarios [5]. In this case, it is assumed that static sensors are deployed at a bus stop or anywhere along a road side street or a park, whereas the mobile object (e.g., a person) walking along the street and collects data from static sensors [5]. Moreover, in [6] and [7] authors have presented detailed studies on recently proposed mobile WSNs.

The smooth and secure network coordination among tiny sensor nodes requires the security mechanisms. Whereas security in the dense SWSNs is always a prime concern due to the resource constraint nature of sensor nodes. Apparently, providing security to WSNs, the secret key should be shared between the communicating entities (i.e., static sensors and mobile objects). During the last decade, numbers of key management schemes [8]-[15] have been proposed for securing large scale SWSNs. Each scheme has pros and cons. However, high computation and communication costs, keys storage overhead, and (security) protocols design weaknesses are always vulnerable to some kind of serious attacks or threats. *Bechkit et al* [8] presented a highly scalable key pre-distribution scheme, where each sensor node is preloaded with *(m+1)* disjoint keys. In [9], authors proposed a key management scheme for heterogeneous sensor networks. In the scheme, *Du et al* suggested *M* keys to each high-end sensor and *L* keys to each low-end sensor. In one hand, the high number of keys (as in [8]-[10][14] [15]) can affect the schemes due to the various adversary attacks, e.g., a node compromise attack. On the other hand, all the preloaded keys are generally not used throughout the network lifetime and hence the huge storage overhead at the sensors side.

In addition, *Chatzigiannakis et al* proposed an agent-based distributed group key establishment scheme in WSN where a mobile agent (i.e., software, mobile code) traverse the network [23]. In [24], *Nehra and Patel* developed a similar research to *Chatzigiannakis et al*. However, the security is in fact a significant concern with a mobile (software) agent, for instance, as a sensor node receiving a mobile (software) agent for execution, it may require strong authentication. In another research [11], authors proposed a simple mobile assisted key distribution scheme, which is easy and efficient to implement in real WSN. However, we have pointed out some practical issues, as analyzed in the next Section-II. Thus, efficient secret key distribution schemes are still need to be designed for the real-time WSNs.

This paper proposes a mobile object-based secret key distribution scheme with efficient use of resources such as, a small number of message transmissions (secret key broadcasting). The proposed scheme exploits the symmetric cryptography, as in [11]. Each low-cost static sensor shares a master key with the mobile object. The idea of proposed scheme is straight-forward, i.e., a mobile object broadcasts secret keys to the static sensor nodes within its broadcasting range. Upon receiving the mobile object message, sensor nodes authenticate to the mobile object and retrieve the secret key.

The proposed scheme regards the entity authentication (i.e., mobile object) and message authentication through the implementation on Telos B wireless sensor nodes. In addition, the analysis shows how a single broadcast message can significantly reduce the communication cost between the mobile object and the sensor nodes.

The rest of the paper is structured as follows. Section - II briefly reviews the *Tas and Tosun's* scheme. Section - III presents the proposed secret key distribution scheme. Section - IV demonstrates the security and performance analysis of the proposed scheme. Conclusions are drawn in Section - V.

II. Brief Review of Tas and Tosun's Scheme

This section briefly reviews and analyzes the work presented in [11]. *Tas and Tosun* presented a mobile assisted key distribution scheme, where a mobile element (ME) broadcasts secret key messages within its broadcasting range *(*r)* and distributes the secret key to the static sensor nodes. The route of the ME and secret key broadcast locations are pre-defined. A master key (K_M) is shared between the static sensor nodes and the mobile element. Authors assume that the mobile element travels to sensors terrain and broadcasts the secret keys to the static nodes. It broadcasts two messages: (1) secret key broadcast message (SKBM); and (2) authenticated key disclosure message (AKDM).

SKBM: ME $\rightarrow$ $*r: E_{KM}(K_S||SN), MAC(K_A, E_{KM}(K_S||SN))$. Here, E is an encryption using key (K_M), K_S is a new secret key, SN is a sequence number, and K_A is an authentication disclosure key.

AKDM: ME $\rightarrow$ $*r+\varepsilon: E_{KM}(K_A||SN)$.

In this scheme, each SKBM is followed by the corresponding AKDM, i.e., message authentication codes (MACs) are followed by its disclosure/verification key. A sensor first buffers the SKBM message and then upon receiving the corresponding AKDM, it verifies the authenticity of ME. If it holds then it decrypts SKBM and retrieves the secret key (K_S) from the buffered message $(i.e., D_{KM}(K_S||SN))$. In addition the broadcasting range $(*r+\varepsilon)$ of AKDM is higher than the SKBM. The corresponding AKDM of a SKBM should be broadcast after the ME moves on its route ε unit, so that all the sensors receive corresponding key disclosure broadcasts.

Analysis: Assumed that a mobile adversary (Tom) can follow to a ME. Tom has full control over the wireless channels, such as he can replay the old messages, block the messages, and may create delay in broadcasting order.

Generally sensor nodes have two modes (sleep and active) for saving energy. In *Tas and Tosun's* scheme, a sensor first buffers the SKBM and then it waits for the AKDM (i.e., disclosure key message). In the real practice, it is possible that after receiving the SKBM at the sensor side, Tom can make delay or block corresponding AKDM. Consequently sensor can go into sleep mode and never receive the corresponding AKDM broadcasts. By doing so, SKBM cannot be verified without the key disclosure message (i.e., AKDM) and Tom can easily pose the denial-of-service threat to the application. In addition, it is very practical that if the disclosure key message (i.e., AKDM) received before the SKBM, then a message forgery is possible by using the AKDM (i.e., disclosure key). To apply possible message forgery attack, refer to [13]. Therefore, *Tas and Tosun's* scheme could have some practical issues for the real-time applications.

III. Proposed Secret-key Distribuition Scheme

This section first describes the overview of the system model and later presents the proposed scheme.

A. System model

In this model, we consider that a WSN consists of large *(P)* number of static sensor nodes, which are deployed over the interest of field for continuously monitoring the environment. As shown in figure 1, a deployed sensors region is divided into Voronoi cells *(i.e., a,b,c,…, and so on)* using [16]-[18]. A mobile object (MO) roams/visits to the sensor networks and collects the sensor data. The MO's visits (e.g., i^{th}) would be defined by the gateway/remote server or applications specific. However, the aim of mobile object is to distribute the new secret keys (for every i^{th} visit) to the static nodes [11] and collect the sensed data from nodes [1]. In order to accomplish the task, MO needs to route through the center of the Voronoi cells. In addition, it can detect the misbehavior of malicious nodes or compromised keys using [25] and revoke them. Note that, for the security purposes, we consider that nodes in one cell cannot directly communicate with the other cells' nodes.

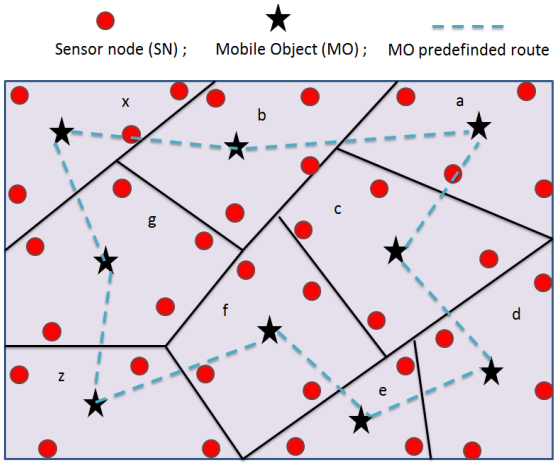

Fig. 1. System Model: a network is divided into *N* cells *(e.g., a, b, c ….)*.

We make few network assumptions for the proposed scheme, as follows:

- MO is a trusted node and the locations of static nodes are known by MO. It is a resource-rich node, and can compute complex operations.
- Each Voronoi cell (e.g., cell *a* in fig 1.) has an unique master key (M_{Key}), which is shared between the mobile object and the members of a cell.
- MO secret-key broadcasting locations (i.e., center of each cell in our scheme) and routes are pre-defined, as in [11].

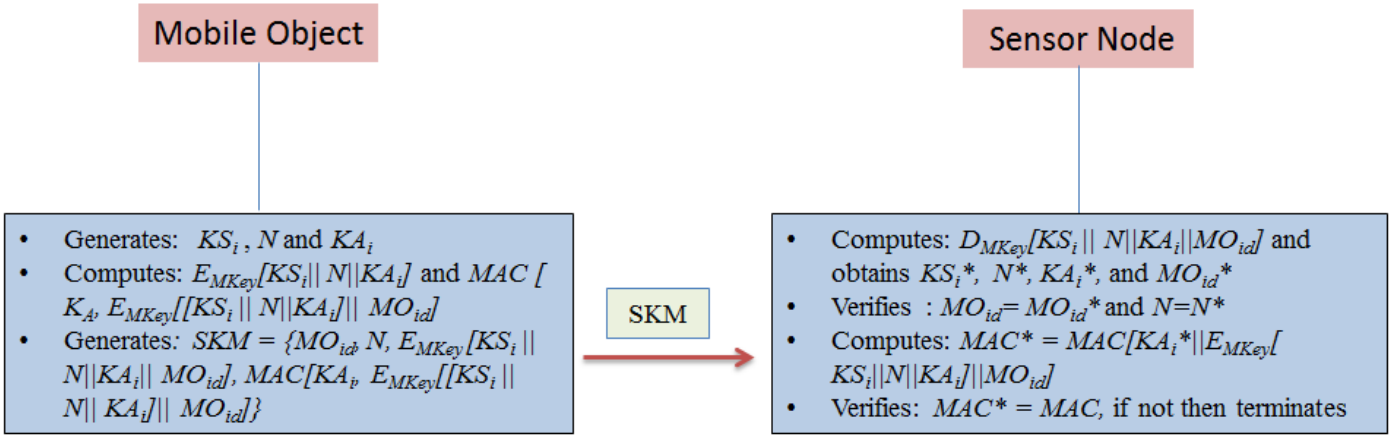

Fig. 2. Secret key distribution using a mobile object.

The notations used now onwards in this paper are listed in Table-I.

TABLE I. LIST OF NOTATIONS AND DESCRIPTIONS

Notations	Description
MO_{id}	Mobile object identity
SN	Static sensor node
M_{Key}	Master key
KS_i	New secret key for MO's i^{th} visit, here $i = 1,2,3...n$
MAC	Message authentication code
KA_i	MAC disclosure key for MO's i^{th} visit, here $i = 1,2,...n$
N	True nonce
$E_K[x]$	Encrypt message x using symmetric key K
$D_K[x]$	Decrypt message x using symmetric key K
$MAC(K, m)$	Message authentication code (MAC) using symmetric key K on message m
$\|\|$	concatenation operation

B. *Proposed scheme*

As shown in figure 1, the mobile object should route through the center of each cell and broadcast the new secret keys to member nodes for its i^{th} visit. To illustrate overall scheme, a mobile object performs as following:

- Generates: KS_i, N, *and* KA_i here KS_i is a new secret key for MO's i^{th} visit, N is a random number, and KA_i is an authentication disclosure key.
- Computes: $E_{MKey}[KS_i|| N||KA_i]$ and $MAC[KA_i, E_{MKey}[[KS_i || N||KA_i]|| MO_{id}]$.
- Generates a message, i.e., $SKM = \{MO_{id}, N, E_{MKey}[KS_i || N||KA_i|| MO_{id}], MAC[KA_i, E_{MKey}[[KS_i || N|| KA_i]|| MO_{id}]\}$

Now, it broadcasts the secret key message *(SKM)* from its predefined location (i.e., through cell center) to the cell's nodes, as follows:

$$MO \longrightarrow SNs: SKM$$

Upon receiving the message *(SKM)*, the static sensor node performs the followings:

- It decrypts the sub-message *(i.e.,* $D_{MKey}[KS_i || N || KA_i || MO_{id}]$*)* using shared master key (M_{Key}) and obtains K_{Si}*, N*, KA_i*, and MO_{id}*.
- Now SN's verifies $MO_{id} = MO_{id}$* and $N=N$*, if it holds then MO is an authentic object and the received message is fresh; otherwise, it terminates the system.
- In order to check the message authenticity and integrity of *SKM*, SN's computes MAC using authenticator disclosure key (KA_i*) and verifies it. If it holds then received *SKM* is an authentic message; otherwise, it aborts the whole system.

Finally, upon receiving the new secret key, *i.e.*, KS_i, cell's nodes have to delete their shared master key (M_{Key}) and the new secret key, KS_i, will be used as a master key for the next $(i.e., i+1^{th})$ visit of MO's. Figure 2 depicts the flow of proposed scheme.

By doing so, a mobile object can securely achieve the fundamental network security goals (e.g., confidentiality, strong authentication, integrity, and message freshness) [12].

IV. SECURITY AND PERFORMANCE ANALYSIS

A. *Security Analysis*

It is assumed that an attacker (Tom) can overhear on wireless messages, intercept messages, inject new messages, and delay or block the wireless messages. Moreover, Tom can physically capture the node and extract all the secrets. Under this attack model, the proposed scheme offers strong security services at minimal computation and communication cost.

Message Confidentiality: Assuming that Tom eavesdrops on SKM *(i.e.,* MO_{id}, N, $E_{MKey}[KS_i || N||KA_i||MO_{id}]$, $MAC[KA_i, E_{MKey}[[KS_i || N|| KA_i]|| MO_{id}]$*)*and tries to extract some secret information. However, Tom will not succeed in this operation, since all the secrets are confidential using master key, which is only shared between the legitimate entities (mobile object and sensor nodes). Hence the proposed scheme achieves confidentiality.

Strong Authenticity and Integrity: In the proposed scheme, MO's secret key message *(SKM)* is strongly authenticated by each static sensor. Assume that mobile Tom broadcasts some altered messages to static nodes. However, in the proposed scheme the authenticity and integrity of each message is verified using message authentication code *(i.e.,* $MAC^* = MAC\ [KA_i^*, E_{MKey}\ [[KS_i \ ||\ N||\ KA_i]\ ||MO_{id}])$, it is computed over the KA_i, which is known to only the legal static nodes and the mobile object. Moreover, MAC itself ensures the message integrity. Hence, the proposed scheme retains strong authentication and integrity.

Message Freshness: In the proposed scheme, each new secret key broadcast's *(i.e., SKM)* freshness is verified using true nonce *(i.e.,* $N = N^*$*)*, hence Tom cannot succeed in replaying the previously broadcasted *SKM* to the static nodes.

Mobile object impersonation attack: In real-time it is very practical that Tom tries to impersonate as a legal mobile object (with MO_{idTOM}) to the static sensors. However, Tom cannot impersonate as a legal MO in the proposed scheme, since the message SKM contains original MA's identity (MO_{id}), which is encrypted with M_{Key}. Therefore, sensors will not verify MO_{idTOM} ($MO_{idTOM} \neq MO_{id}$) and discards the message.

Resilience against node capturing: It is possible that Tom can physically capture a static node and extract the secrets from the node. It is widely known that if a node is compromised then there is no meaning for application security. However, in the proposed scheme each cell has a unique master key, which is only shared among its nodes. It is assuming that if a cell node is compromised with Tom then it does not affect to the rest of other cells nodes. Hence, the proposed scheme is resilient to some extent in the node capture attack.

B. Performance Analysis

This subsection discusses the performance analysis of the proposed scheme. As shown in figure 3, we conducted experiments on a single-hop test-bed using TelosB nodes and evaluated the performance measurements including the code size and the running time at sensors side, only.

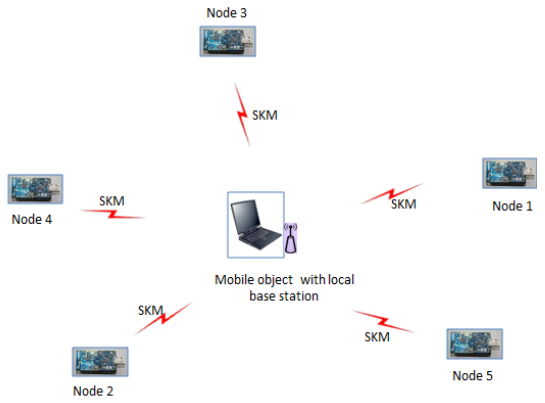

Fig. 3. Experimental setup with TelosB sensor motes

Implementation: As shown in figure 3, the experimental setup consists of five TelosB sensor nodes [20], with IEEE 802.15.4 compliant RF transceivers. The hardware includes 8 MHz, 16-bit MCU with 10 Kbyte RAM and 48 Kbyte ROM. More details of TelosB mote are shown in Table –II.

TABLE II. TELOSB SPECIFICATION

ITEMs	**DESCRIPTIONs**
Processor	16-bit RISC
Internal memory	10-Kbytes RAM
Flash memory	48-Kbytes ROM
Multi-channel radio	2.4-GHz(CC2420)
Data rate	250kbps
Interface	UART
Sensors	Temperature, humidity, light, etc.

The proposed scheme is implemented at both the sender (i.e., mobile object) and the receiver sides with the respective roles of the mobile object and the sensor nodes. At the mobile object side programs are running on a local base station, which is serially connected to a 2.50GHz laptop. Due to the high demand of resource saving at a sensor node side, we have presented the experimental results only at the receiver's point of view. Whereas at MO's side, it is assumed that MO's is a resource rich device and it can compute complex (mathematical and cryptographic) operations efficiently. Therefore, we performed our measurements at sensors side, only. The encryption and decryption operations (i.e., message confidentiality) performed using a non-optimized software-based advanced encryption standard (AES) algorithm at sensors side [22]. For the sake of security comparisons purpose, we evaluated the authentication and integrity operations with three security mechanisms, namely, chaining block cipher-MAC (CBCMAC), SHA-1, and hashed message authentication code (HMAC). The proposed scheme is developed using NesC on TinyOS 2.x [21].

As figure 3 depicts, a mobile node (sender) broadcasts a secret key message ($SKM = \{MO_{id}, N, E_{MKey}\ [KS_i \ ||\ N||K_A\ ||MO_{id}],\ MAC[K_A, E_{MKey}[[KS_i||\ N||\ K_A]||\ MO_{id}]\ \}$) and the receivers node 1, 2, 3, 4, and 5 can receive it.

Computation overhead: In the experiments we have taken the memory consumption values and execution timing values for message decryption and verification (i.e., authentication and integrity) at the receiver side.

For the sake of comparisons we have varied all the keys *(i.e.,* master key (M_{Key}), authentication key (K_A), and new secret key (KS_i)*)* sizes as 64 bits, 128 bits and 256 bits.

The memory consumptions of proposed scheme are shown in Table-III. For the 64 bits key size: AES+CBCMAC requires 2.72KB of RAM and 14.5KB of ROM; AES+SHA-1 takes 2.73KB of RAM and 15.6KB of ROM; and AES+HMAC requires 2.75KB of RAM and 16.3B of ROM. For the 128 bits key size: AES+CBCMAC takes 3.11KB of RAM and 14.5KB of ROM; AES+SHA-1 requires 3.18KB of RAM and 15.7KB of ROM; and AES+HMAC incurs 3.20KB of RAM and 16.4KB of ROM. Moreover, as we can see from table-III, the key size of 64bits key and 128bits key do not require much memory as compared to the key size of 256bits.

TABLE III. CODE SIZE OF PROPOSED SCHEME (MEMORY USES) IN BYTES

Key Size (Bits)	AES+CBCMAC		AES+SHA-1		AES+HMAC	
	RAM	ROM	RAM	ROM	RAM	ROM
64	2726	14572	2736	15672	2758	16382
128	3110	14572	3184	15736	3206	16446
256	3878	14572	4080	15864	4102	16574

Likewise we have taken up execution timing values for AES+CBCMAC, AES+SHA-1, and AES+HMAC and, as shown in Table-IV. For the 64 bits key size, AES+CBCMAC AES+SHA1, and AES+HMAC requires 4ms (milliseconds), 24ms, and 57ms, respectively. For the 128 bits key size, AES+CBCMAC needs 4ms, AES+SHA-1 require 40ms, and AES+HMAC requires 57ms of time to execute the decryption, authentication and integrity of broadcasted message. Similarly for the 256 bits key size, AES+CBCMAC takes 4ms, AES+ SHA-1 takes 75ms and AES+HMAC takes 91ms of time for executing the cryptographic operations.

TABLE IV. IMPLEMENTATION RESULTS (TIMING VALUES) IN MS(MILLISECONDS)

Key Size (Bits)	AES+CBCMAC	AES+SHA-1	AES+HMAC
64	4	24	41
128	4	40	57
256	4	75	91

As shown in figure 4, one of the observations is that the execution time of AES+CBCMAC operation remains unchanged for 64 bits key, 128 bits key and 256 bits key. However the timing values for AES+SHA-1 and AES+HMAC operations are increasing, while extending the key size, with an equal ratio.

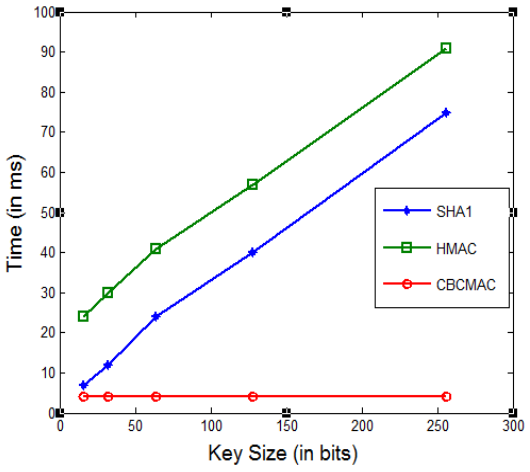

Fig. 4. Execution time for different number of key size.

It is obvious that increase in the key sizes at mobile object side also lengthens the storage (memory) and execution time at sensor side, as shown in Table-III and Table-IV. In addition, it is justifiable that our proposed scheme provides more security in terms of strong authentication and integrity, confidentiality, and message freshness at reasonable costs.

Communication cost: In order to compare the communication cost, we consider that how many broadcast messages are required to execute the whole protocol. As shown in Table V, the proposed scheme requires only one broadcast message *(i.e., SKM)* whereas *Tan and Tosun's* scheme requires two broadcast messages *(SKBM and AKDM).*

TABLE V. COMMUNICATION COST COMPARISON WITH [11]

Scheme	Number of broadcasts
Tan and Tosun's scheme [11]	2
Proposed scheme	**1**

Hence, the proposed scheme requires the least secret key broadcast messages (i.e., only one).

V. CONCLUSIONS AND FUTURE DIRECTIONS

In this paper, we have proposed a mobile object based secret key distribution framework for WSNs. The proposed scheme exploits the symmetric cryptography and broadcasts symmetric secret keys to the static nodes. We have investigated and implemented the proposed scheme on a real-time test-bed. The experimental results show the additional security overhead and its feasibility to the real-world applications.

We have not done the measurements on the authentication delay and total network overhead. However, the authentication delay depends on many factors such as mobility of MA, the distance between mobile agent and sensor nodes, multi-hop, speed of mobile agent, link quality, etc. We will focus on the above mentioned issues and other security threats countermeasure in the future version of this paper.

ACKNOWLEDGMENT

This paper work has been funded by Tekes under Massive Scale Machine-to-Machine Service (MAMMotH) project.

REFERENCES

[1] J.H. Jun, B. Xie, D. Agarwal, "Wireless Mobile Sensor Networks: Protocols and Mobility Strategies, " *Computer Communications and Networks*, 2009, pp. 607-634.

[2] D. G. Stratogiannis, G. I. Tsiropoulos, J. D Kanellopoulos, P. G. G. Cotties, "4G Wireless Networks: Architectures, QoS support and Dynamic Resource Management," Wireless Netwrok Traffic and Quality of Service Support: Trends, and Standards, 1st ed., T. Lagkas, P. Anglidis, and L. Georgiadis, Ed. New York: IGI Global, 2010.

[3] J. Rao, T. Wu, and S. Biswas, "Network-Assisted Sink Navigation Protocols for Data Harvesting in Sensor Networks," *in proceeding of IEEE Conference on Wireless Communications and Networking (WCNC'08*) 2008, pp. 2887-2892.

[4] K. Mikhaylov and J. Tervonen, "Data Collection from Isolated Clusters in Wireless Sensor Netwroks Using Mobile Ferries," *in proceeding of FINA'2013(AINA'13)*.

[5] G. Anastasi, M. Conti, E. Gregori, C. Spagoni, G. Valente,"Motes Sensor Networks in Dynamic Scenarios: an Experimental Study for Pervasive Applications in Urban Environments," http://info.iet.unipi.it /~anastasi/papers/juci06.pdf

[6] M. D. Franscesco, S. K. Das, G. Anastasi, "Data Collection in Wireless Sensor Netwroks with Mobile Elements: A Survey," *ACM Trans. on Sensor networks*, vol. 8, No. 1, Article 7, 2011.

[7] X. Li, R. Falcon, A. Nayak, I. Stojmenovic, "Servicing Wireless Sensor Networks by Mobile Robots," *IEEE Communications Magazine*, July 2012, pp. 147-154.

[8] W. Bechkit, Y. Challal, A. Bouabdallah, V. Tarokh, "A Highly Scalable Key pre-Distribution Scheme for Wireless Sensor Networks," *IEEE Trans. on Wireless Communications*, vol. 12, no. 2, February 2013, pp. 948-959.

[9] X. Du, Y. Xiao, M. Guizani, H.H. Chen, "An Effective Key Management Scheme for Heterogeneous Sensor Networks," *Ad Hoc Networks*, vol. 5, Issue.1, 2007, pp. 24-34.

[10] H. Lee, K. Shin, D.H. Lee, "PACPs: Practical Access Control Protocols for Wireless Sensor Networks," *IEEE Trans.on Consumer Electronics*, vol. 58, No.2, May 2012, pp. 491-499.

[11] B. Tas, and A. S. Tosun, "Mobile Assisted Key Distribution in Wireless Sensor Networks," *in proceeding of IEEE ICC 2011*, 5-9 June 2011, Kyoto, Japan.

[12] A. Perrig, R. Szewczyk, V. Wen, D. Culler, J. D. Tygar, "SPINS: Security Protocols for Sensor Networks," *Wireless Networks*, vol. 8, September 2002, pp. 521-534.

[13] P. Zeng, K. R. Choo, D. Sun, "On the Security of an Enhanced Novel Access Control Protocol for Wireless Sensor Networks," *IEEE Trans.on Consumer Electronics*, vol. 56, no, 2, May 2010.

[14] A. K. Das, "An Unconditionally Secure Key Management Scheme for large-Scale Heterogeneous Sensor Networks," *in proceeding of 1st International Conference on COMmunication System and And NETworks*, 5-10 January, 2009, pp. 365-662.

[15] L. Eschenauer and V. D. Gligor, "A Key-management scheme for Distributed Sensor Netwroks," *in proceeding of ACM conference on computer and communications security*, November 2002, pp. 41-47.

[16] M. O. Rahman, M. A. Razzaque, C.S Hong, " Probabilistic Sensor Deployment in Wireless Sensor Network: A New Approach," *IEEE ICACT 2007*, pp. 1419-1422.

[17] H. Mahboubi, J. Habibi, A. G. Adhdam and K. S. Pour, "Cooperative Self-Deployment Strategies in a Mobile Sensor Network with Non-Uniform Coverage Priority," *IEEE Globecom* 2011.

[18] A. Boukerche, X. Fei, "A Voronoi Approach for coverage protocols in wireless sensor networks," *IEEE Globecom* 2007, pp. 5190-5194.

[19] X. Du, Y. Xiao, "Energy Efficient chessboard clustering and routing in heterogeneous sensor networks," *International Journal of Wireless and Mobile Computing*, Vol.1, No.2/2006, pp. 121-130.

[20] TelosB Datasheet; http://www.willow.co.uk/TelosB_Datasheet.pdf

[21] TinyOS 2.x; www.tinyos.net/tinyos-2.x/

[22] Advanced Encryption Standard: http://tinyos.cvs.sourceforge.net/viewvc/tinyos/tinyos-2.x-contrib/crypto/index.html.

[23] I. Chatzigiannakis, E. Konstantinou, V. Liagkou, and P. Spirakis, "Agent-based Distributed Group Key Establishment in Wireless Sensor Networks," *IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM 2007)*, 18-21 June 2007, Espoo, Finland.

[24] N. Nehra and R.B. Patel, "MASLKE: Mobile Agent Based Secure Location Aware Key Establishment in Sensor Networks," *In the proceeding of IEEE International Conference on Networks, (ICON 2008)*, 12-14 December 2008, New Delhi, India.

[25] M. Drozda, S, Schaust, H. Szczerbicka, "AIS for Misbehavior Detection in Wireless Sensor Network: Performance and Design Principles," *In Proceeding of IEEE Congress on Evolutionary Computation*, 2007, pp. 3719-3726.