A Novel Secure Authenticated Key Exchange Protocol for Wireless Sensor Networks

Tamer Barakat[†]

[†]Faculty of Engineering, Fayoum University, Fayoum, Egypt

Summary
Recently, Eun-Jun *et al.*'s proposed an improvement of the authentication key exchange protocol. In their protocol, they proved that neither Hung et al.'s nor Tian et al.'s provide perfect forward secrecy, and present an improved protocol in order to address this problem. In this paper, we will demonstrate some security leaks inherent in Eun-Jun *et al.*'s protocol and show that this protocol is still suffers from perfect forward secrecy problem. Finally, we will propose a new Secure Authenticated Key Exchange (SAKE) protocol to eliminate the pointed out security leaks.

Keywords:
Cryptography, sensor networks, key exchange protocol, perfect forward secrecy, self-organizing.

1. Introduction

Self-organizing sensor networks [1] are emerging and being put into practice for many new applications. This type of network can usually be set up quickly and inexpensively using low-cost and ultra-low-power sensors. Target applications include battlefield services, rescue missions, nature research projects, etc. The self-organizing model can be further divided into two cases [1] [2]. In the nonuniform self-organizing model the network may contain two types of nodes: full functional devices (FFD) with high energy, power, and storage capabilities; and the restricted functional devices (RFD), which are typical low-capability sensor nodes. In the uniform self-organizing model, all nodes are assumed to be restricted functional devices. A RFD plays on the role of an end device, such as a low-power sensor, while a FFD takes the role of a coordinator, a router, or a sensor. The initial session key is then used to build a secure channel in which link keys are subsequently installed into the sensor by the security manager.

Various kinds of Authentication Key Exchange (AKE) protocols have been proposed, such as Huang et al.'s protocol [3] which provides key exchange and mutual authentication between a sensor node and a security manager. This protocol was suffered from a big problem which is that a security manager can learn the long-term private key of a sensor after having one normal run of the protocol with the sensor.

Tian et al. [4], proposed an improvement AKE protocol which solves the problem of Hug et al.'s protocol and makes all of their security claims hold again.

Recently, Eun-Jun et al. [5] showed that neither protocol is appropriate for uniform self-organizing sensor networks, since they assume the existence of full-functional devices and don't provide perfect forward secrecy [6][7].

In this paper, we will analyze the security of Eun-Jun et al.'s protocol and we will show that it doesn't provide perfect forward secrecy which make that; achieving the required security level is more difficult in the uniform level. To remedy this weakness, we propose a new improvement which provides perfect forward secrecy. The rest of this paper is sketched as follows: in Section 2, we will review Eun-Jun *et al.*'s protocol. In Section 3, we will point out the security leaks inherent in Eun-Jun *et al.*'s protocol. A novel improvement is proposed in Section 4. In Section 5 we analyze the security of our protocol. Finally, we give conclusions in Section 6.

2. Brief Review of Eun-Jun et al.'s Protocol

In this section, we briefly review Eun-Jun et al.'s protocol. The scheme consists of two phases: certificate generation and AKE protocol for initial session key phases.

In this protocol, the first phase (certificate generation) is the same as both Huag et al.'s and Tian et al.'s AKE protocol. Figure 1 shows the second phase (AKE protocol for initial session key) in which a sensor and a security Manager carries out an AKE protocol to establish an initial session key. The proposed initial session key generation phase requires only five rounds and works as follows:

Phase 2: The AKE Protocol for Initial Session Key. After generating certificates and public key pairs, sensor U and security manager V carries out the following protocol to establish an initial session key. The session key will be used to set up a secure channel for V to install link keys to U.

(1) $U \rightarrow V: IC_U = Q_{CA},\ ID_U, B_U, t_U$
$V \rightarrow U: IC_V = Q_{CA}, ID_V, B_V, t_V$
U and V exchange their implicit certificates. The content the certificate is verified on the other side

Manuscript received December 5, 2012
Manuscript revised December 20, 2012

that includes the certificate format, expected device identity, and the validity period. The public keys of the counter-party are also obtained from the certificates; V obtains $U's$ public key Q_U as $H(IC_U)B_U + Q_{CA}$. Similarly, U obtains $V's$ public key Q_V as $H(IC_V)B_V + Q_{CA}$. If anything goes wrong, the protocol is terminated. Note that $Q_U = q_U P$ and $Q_V = q_V P$.

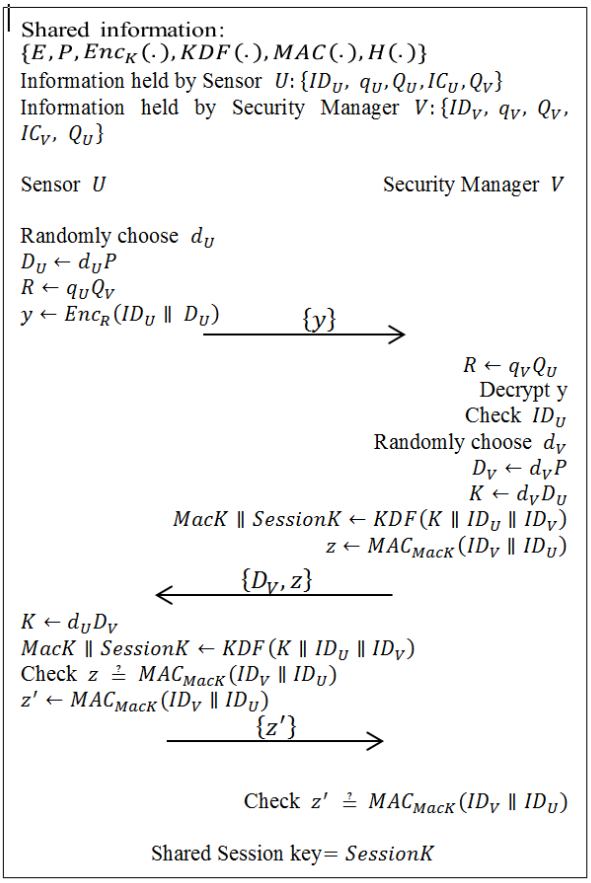

Fig. 1. Eun-Jun et al.'s AKE Protocol

(2) $U \rightarrow V: y = Enc_R(ID_U \| D_U)$
U randomly picks d_U, compute $D_U = d_U P$, $R = q_U Q_V = q_U q_V P$, and sends $y = Enc_R(ID_U \| D_U)$ to V, where Enc_R is some secure symmetric key encryption function under the key R.

(3) $V \rightarrow U: D_V, z = MAC_{MacK}(ID_U \| ID_V)$
V compute $R = q_V Q_U = q_V q_U P$, decrypt y, and check if the plaintext is ID_U followed by some number. If the check is fails, the protocol is terminated. Otherwise, V denotes the number which follows ID_U as D_U. V then randomly picks d_V, and computes $D_V = d_V P$, $K = d_V D_U = d_V d_U P$ and $MacK \| SessionK = KDF(K \| ID_U \| ID_V)$. V then sends $D_V, z = MAC_{MacK}(ID_U \| ID_V)$ to U. Then V destroys d_V, R, and K.

(4) $U \rightarrow V: z' = MAC_{MacK}(ID_U \| ID_V)$
U compute $K = d_V D_U$ and $MacK \| SessionK = KDF(K \| ID_U \| ID_V)$. U then checks if $z' \stackrel{?}{=} MAC_{MacK}(ID_U \| ID_V)$. If the check fails, the protocol is terminated. Otherwise, U sends $z' = MAC_{MacK}(ID_U \| ID_V)$ to V.Then U destroys d_u, R and K.

(5) V checks if $z' \stackrel{?}{=} MAC_{MacK}(ID_U \| ID_V)$ is valid. If not, the protocol is terminated. The initial session key established by U and V is $SessionK$.

3. Security Analysis of Eun-Jun et al.'s Protocol

In this protocol, suppose an attacker E obtains the long-term private key q_V from the compromised security manager. It is easily to compute $R = q_V Q_U$ where Q_U is the public-key for sensor U.
Once the attacker E knows the value of R, he is easily to decrypt y and get the value of D_U as he previously knows the value of ID_U. Then, the attacker E will play as a general-purpose attack known on Elliptic Curve Discrete Logarithm Problem (ECDLP) which is the combination of the Pohlig-Hellman algorithm [8] and Pollad's Rho algorithm [9] to find the value of d_U as follow:
The principle concept of the Pollard's Rho algorithm is to find distinct pairs (c', d') and $(c", d")$ of integers $mod\ n$ such that:

$$c'P + d'P = c"P + d"D_U \quad (1)$$

$$(c' - c")P = (d" - d')D_U = (d" - d')d_U P \quad (2)$$

And so,

$$(c' - c") = (d" - d') \quad in\ \langle P \rangle \quad (3)$$

Hence, $d_U = dlog_P D_U$ can be obtained by computing:

$$d_U = (d' - d")^{-1}(c' - c") \quad mod\ n \quad (4)$$

If the attacker E is successfully computed d_U, he is very easy to compute $K = d_U D_V$ since he intercepts transmitted value D_V from an open network.
Finally, the attacker E can compute the shared session key $MacK \| SessionK \leftarrow KDF(K \| ID_U \| ID_V)$by using K. Then, we conclude that Eun-Jun et al.'s doesn't provide perfect forward secrecy.

4. The Proposed (SAKE) Protocol

To resist the attack pointed out in the previous section, we propose an improved protocol.
First, we will review some Preliminaries that will be used in this paper. Second, we will describe the proposed protocol.

4. 1 Preliminaries

In this section, we review some definitions, notations and widely accepted complexity [10] [11] [12].

Notation 4.1 (**some of the notations used in the proposed SAKE protocol are defined as follows)**

- C: an elliptic curve defined over GF(P)
- P: a base point of large order $n$on **C**. It is assumed that the elliptic curve discrete logarithm problem $(ECDLP)$[6] [7] in this group is intractable.
- CA: a system-wide trusted party called Certificate Authority.
- U, V : a sensor and a security manager, respectively.
- q_{CA}, Q_{CA}: the private/public key pair of CA, where q_{CA} is a random integer and $Q_{CA} = q_{CA}P$.
- $KDF(.)$: a secure key derivation function.
- $MAC(.)$: a message authentication code function.
- ||: the conventional binary string concatenation operator.
- K ∈ N: a system-wide security parameter.

Definition 4.1 **(Elliptic Curve Discrete Logarithm Problem (ECDLP))**

Given an elliptic curve E over a finite field F_q, a point $P \in E(F_q)$ of order n, and a point $Q \in \langle P \rangle$, find the integer $l \in [0,\ n-1]$ such that $Q = lP$. The integer l is called the discrete logarithm of Q to the base P, denoted $l = dlog_P Q$.
For security, the $ECDLP$ should be intractable and so the elliptic curve parameters (q, the equation of $E, P, n = ordP$) should be carefully chosen in order to resist all known attacks on the $ECDLP$.

Definition 4.2 **(Strong Elliptic Curve)**

A cryptographically strong elliptic curve is an elliptic curve such that the DLP in the group of points is expected to be difficult.
The most naive way to solve the $ECDLP$ is an exhaustive search where one computes the sequence $P, 2P, 3P, ...$ until the result is equal to Q. In the worst case one need to compute n steps and $n/2$ steps on average.

Assumption 4.1 **(security constraints)**

- In order to avoid Pohlig-Hellman attack and Pollard's rho attack in the $ECDLP$, it is necessary that $\#E(F_q)$ is divisible by a sufficiently large prime n $(e.g.\ n > 2^{160})$.
- Maximum resistance to these two attacks can be attained by selecting E such that $\#E(F_q)$ is prime or almost prime, i.e., $\#E(F_q) = hn$ where n is prime and h is small $(e.g.\ h = 1, 2, 3)$.

4.2 The SAKE Protocol

This protocol consisting of two basic phases: the first phase (certificate generation) is the same as both Huge et al.'s and Tian et al.'s AKE protocols

Phase 2: The proposed Protocol for Initial Session Key.

After generating certificates and public key pairs, sensor *U* and security manager *V* carry out the following protocol to establish an initial session key. The session key is used to set up a secure channel for V to install link keys to *U*. Figure 2 shows the SAKE protocol.

(1) $U \rightarrow V: IC_U = Q_{CA},\ ID_U, B_U, t_U$
$V \rightarrow U: IC_V = Q_{CA}, ID_V, B_V, t_V$
U and V exchange their implicit certificates. The content the certificate is verified on the other side that includes the certificate format, expected device identity, and the validity period. The public keys of the counter-party are also obtained from the certificates; V obtains $U's$ public key Q_U as $H(IC_U)B_U + Q_{CA}$. Similary, U obtains $V's$ public key Q_V as $H(IC_V)B_V + Q_{CA}$. If anything goes wrong, the protocol is terminated. Note that $Q_U = q_U P$ and $Q_V = q_V P$.

(2) $U \rightarrow V: y = rP, (ID_U \parallel D_U) + rQ_V$
U randomly picks d_U and r, compute $D_u = d_U P$, and sends $y = \{rP, (ID_U \parallel D_U) + rQ_V\}$, where y represents the ciphertext of the component: $(ID_U \parallel D_U)$ using the Elliptic Curve Cryptosystem.

(3) $V \rightarrow U: m, z = MAC_{MacK}(ID_U \parallel ID_V)$
V decrypts y, and check if the plaintext is ID_U followed by some number. If the check is fails, the protocol is terminated. Otherwise, denotes the number which follows ID_U as D_U . V then randomly picks d_V and S, and compute $D_V = d_V P$, $K = d_V D_U$, $m = sP, (ID_V \parallel D_V) + sQ_U$, where m represents the ciphertext of the message: $(ID_V \parallel D_V)$ using the Elliptic Curve Cryptosystem and $MacK \parallel SessionK = KDF(K \parallel ID_U \parallel D_V)$. V then sends m and $z = MAC_{MacK}(ID_U \parallel ID_V)$ to U. Then V destroys $d_V, s, and\ K$.

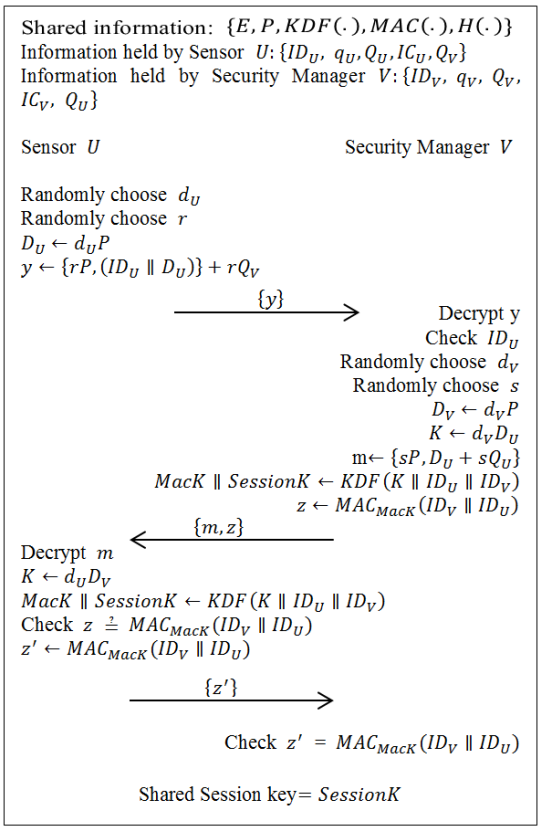

Fig. 2. The SAKE Protocol

(4) $U \rightarrow V: z' = MAC_{MacK}(ID_U \parallel D_V)$
U *decrypt* m and check if the plaintext is ID_V followed by some number. If the check is fails, the protocol is terminated. Otherwise, denotes the number which follows ID_V as D_V. U then computes $K = d_U D_V$ and $MacK \parallel SessionK = KDF(K \parallel ID_U \parallel D_V)$.
U then checks if $z = MAC_{MacK}(ID_U \parallel ID_V)$. If the check fails, the protocol is terminated. Otherwise, U sends $z' = MAC_{MacK}(ID_U \parallel D_V)$ to V. Then U destroys $d_U, r, and\ K$.

(5) V checks if $z' = MAC_{MacK}(ID_U \parallel D_V)$ is valid. If not, the protocol is terminated. Otherwise, the initial session key established by U and V is $SessionK$.

5. Security Analysis for the SAKE Protocol

In this section, we show that our proposed protocol with-stands various strong attacks.

a) Replay Attack: If an adversary E intercepts the information transmitted between sensor and security manager. He can reuse the information to spoof the sensor to be successfully authenticated by the security manager. In the proposed protocol, Attacker E can intercept all transmission values $\{y, m, z, z'\}$ and can use them to impersonate U (or V) when sending the next key agreement message. For a random challenge, $\{d_U, d_V, r, and\ s\}$, which are separately generated by U and V, are different every time. Since U and V always verify the integrity of the fresh session key $SessionK$ by checking z and z' in Steps (4) and (5), the replayed messages can be detected by U and V, respectively.

b) De-synchronization Attack: E sends spoofed messages to make the data stored in both sensor and security manager de-synchronized. It can cause the communication between the sensor and security manager to be invalid temporarily or permanently. Our proposed protocol overcomes the de-synchronization attack because the authentication data stored in both sensor and security manager (ID_U, ID_V) is secured via the encryption function $m = \{sP, (ID_U, ID_V) + sQ_U\}$. Therefore, if an adversary launches a de-synchronization attack on our scheme, he cannot succeed.

c) Impersonation Attack: E utilizes the messages eavesdropped before to impersonate a legitimate sensor (or security manager) to communicate with the security manager (or sensor) and pass the authentication successfully. Attacker E can't successfully execute an impersonation since V verifies the authentication data for U by decrypting the message y using its private key q_V and then returns m and z. U also verifies the authentication data for V by decrypting the message m using its private key q_U. Hence, U will have implicit assurance that it is talking to V. Therefore, the impersonation attack cannot work in our protocol.

d) Man-in-the-Middle Attack: An active adversary modifies the transmitted messages between the sensor and the security manager, making them believe that they are communicating to the intended party. In the (SAKE) protocol we encrypt the authenticated data as follow:
Sensor U selects private key q_U and generates a public key $Q_U = q_U P$ to encrypt and send a message (ID_U, ID_V) to V, U chooses a random positive integer r and produces the cipher text y consisting to the pair of points $y = \{rP, (ID_U, ID_V) + rQ_V\}$.
Note that U has used $V's$ public key Q_V. To decrypt the ciphertext, V multiplies the first point in the pair by $V's$ secret key and subtracts the result from the second point:

$$(ID_U \parallel ID_V) + rQ_V - q_V(rP)$$
$$= (ID_U \parallel ID_V) + r(q_V P) - q_V(rP)$$
$$= (ID_U \parallel ID_V)$$

U has masked the message $(ID_U \parallel ID_V)$ by adding rQ_V to it. Nobody but U knows the value of r, so even though Q_V is a public key, nobody can remove the mask rQ_V. However, U also includes a "clue," which is enough to remove the mask if one knows the private key q_V. For an attacker E to recover the message, the attacker would have to compute r given P and rP, which is hard.

e) **The Proposed Protocol Provides Known-Key Security:** Known-key security means that each run of a key agreement protocol between two entities *U* and *V* should produce unique secret keys; such keys are called session keys. In the proposed scheme it is impossible to compute the other session keys $MacK \parallel SessionK = KDF(K \parallel ID_U \parallel D_V)$ with the next key agreement protocols because knowing the values of both d_U and d_V is very hard since these values are randomly selected for each session.

f) **The Proposed Protocol Provides Perfect Forward Secrecy:** Perfect forward secrecy means that if the long-term private keys of one or more entities are compromised, the secrecy of previous session keys, which was established by honest entities, is not affected. Since the proposed protocol is obeys for the security constraints that was described in the assumption 4.1 in this paper, then If the long-term private keys of two entities *U* and *V* are compromised, an attacker will not be able to determine the session key *K* for the past sessions or to decrypt them, since the attacker is still faced with the *ECDHP*.

6. Conclusion

In this paper, we have addressed the important issue securing a self-organizing wireless sensor networks against perfect forward security. First, security of Eun-Jun et al.'s protocol is analyzed and proved that this protocol does not provide perfect forward security. Second, an enhanced SAKE protocol is presented to address this problem. Third, the security of the proposed protocol is analyzed to proof that the proposed protocol is provide the perfect forward security as well as it is stand against several strong attacks such that: replay attack, De-Synchronization attack, impersonation attack and Man-in-the- Middle attack.

Acknowledgment

The author would like to thank the anonymous reviewers of the IJCSNS for their valuable comments.

References

[1] L. Eschenauer, V. D. Gligor, "A Key-Management Scheme for Distributed Sensor Networks," Proc. of 9th CCS ACM conference. (2002) 41-47

[2] IEEE Std. 802.15.4-2003.: IEEE Standard for Information Technology – Telecommunications and Information Exchange Between Systems - Local and Metropolitan Area Networks - Specific Requirements - Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low Rate Wireless Personal Area Networks (WPANS). (2003)

[3] Q. Huang, J. Cukier, H. Kobayashi, B. Liu, and J. Zhang, "Fast Authenticated Key Establishment Protocols for Self-Organizing Sensor Networks," In Proc. of the Second ACM International Conference on Wireless Sensor Networks and Applications. ACM Press. (2003) 141-150

[4] X. Tian, D.S. Wong, and R.W. Zhu, "Analysis and Improvement of an Authenticated Key Exchange Protocol for Sensor Networks," IEEE Communication Letters. Vol. 9. No. 11. (November 2005) 970-972.

[5] E.-J. Yoon and K.-Y. Yoo "An Optimizing Authinticated Key Exchange for Self-organizing Server Networks" Vol. 4239, pp 537-546, 2006.

[6] V.S. Miller, "Uses of Elliptic Curves in Cryptography," Proceedings of Crypto'85. Santa Barbara. USA. (1986) 417-426.

[7] N. Koblitz, "Elliptic Curve Cryptosystems. Mathematics of Computation," Vol. 48. (1987) 203-209.

[8] C. Pohlig and M. E. Hellman, "An improved algorithm for computing logarithms over GF(p) and its cryptographic significance," IEEE Trans. on Inf. Theory, 24:106–110, 1978.

[9] J.M. Pollard, "Monte Carlo methods for index computation $(modp)$," Mathematics of Computation, 32:918–924, 1978.

[10] A.J. Menezes, P.C. Oorschot and S.A. Vanstone, Handbook of Applied Cryptography," CRC Press. New York. (1997)

[11] W. Diffie and, M.Hellman, "New Directions in Cryptography," IEEE Transaction on Information Theory. Vol. IT-22. No. 6. (1976) 644-654

[12] B. Schneier, "Applied Cryptography-Protocols," Algorithms and Source Code in C.

Tamer Barakat received his BSc in communications and computers engineering from Helwan University, Cairo; Egypt in 2000. Received his MSc in Cryptography and Network security systems from Helwan University in 2004 and received his PhD in Cryptography and Network security systems from Cairo University in 2008. Currently, working as a lecturer, post doctor researcher and also joining several network security projects in Egypt. His main interest is Cryptography and network security. More specially, he is working on the design of efficient and secure cryptographic algorithms, in particular, security in the wireless sensor networks. Other things that interest him are number theory and the investigation of mathematics for designing secure and efficient cryptographic schemes.