Hindawi Publishing Corporation
Journal of Sensors
Volume 2016, Article ID 1210938, 6 pages
http://dx.doi.org/10.1155/2016/1210938

Research Article

An Access Control Protocol for Wireless Sensor Network Using Double Trapdoor Chameleon Hash Function

Tejeshwari Thakur

School of Studies in Mathematics, Pt. Ravishankar Shukla University, Raipur, India

Correspondence should be addressed to Tejeshwari Thakur; tejeshwari31@gmail.com

Received 3 June 2016; Revised 6 September 2016; Accepted 20 September 2016

Academic Editor: Hana Vaisocherova

Wireless sensor network (WSN), a type of communication system, is normally deployed into the unattended environment where the intended user can get access to the network. The sensor nodes collect data from this environment. If the data are valuable and confidential, then security measures are needed to protect them from the unauthorized access. This situation requires an access control protocol (ACP) in the design of sensor network because of sensor nodes which are vulnerable to various malicious attacks during the authentication and key establishment and the new node addition phase. In this paper, we propose a secured ACP for such WSN. This protocol is based on Elliptic Curve Discrete Log Problem (ECDLP) and double trapdoor chameleon hash function which secures the WSN from malicious attacks such as node masquerading attack, replay attack, man-in-the-middle attack, and forgery attacks. Proposed ACP has a special feature known as session key security. Also, the proposed ACP is more efficient as it requires only one modular multiplication during the initialization phase.

1. Introduction

A wireless sensor network (WSN) is a system of a network consisting of spatially distributed autonomous devices which uses sensors to cooperatively monitor physical or environmental conditions such as temperature, sound, vibration, pressure, motion, or pollutants at different locations. The purpose of a WSN is to collect and process data from a target domain and transmit the information back to specific sites. WSN technology is an emerging technology that can be utilized in a wide range of potential applications in the real world. Such a network usually consists of a number of wireless sensor nodes that arrange themselves into a multihop network. Each node consists of one or more sensors. In many WSN, it is sufficient to secure the data transfer between the sensor nodes and the base station, especially, when the base station is needed to ensure that the received message sent by the specific sensor node is unaltered during transfer. However, in any WSN, providing security during authentication, key establishment and new node deployment is important and for that purpose, an ACP is needed. In the health-care monitoring systems, military domains, and in many other applications, WSN requires a hard and fast authentication scheme to secure the data from the attackers because the authenticity and integrity of such data received at the base station highly influence the final results in many WSN applications, as shown by Abduvaliev et al. [1], Akyildiz et al. [2], and Akyildiz and Kasimoglu [3]. In a paper, Zhou et al. [4] developed an ACP based on the elliptic curve cryptosystem (ECC) for securing the new node deployment process. For details on the elliptic curve (EC) one can refer to Miller and Koblitz [5, 6] and so forth. Next, Huang [7] proposed an efficient ACP based on the EC and hash chains. In this scheme, new nodes can be easily added. The authors claimed that it is resistant to various attacks. Later, Kim and Lee [8] pointed out that the ACP given by Huang [7] is insecure and it lacks hash chain renewability which is an important aspect needed in any resource constrained sensor network. Consequently, Kim and Lee [8] further proposed an enhanced ACP by adding a hash chain renewal phase supporting the mutual authentication. Also, they claimed that their enhanced access control protocol is resistant to various known attacks.

Further, Shen et al. [9] and Zeng et al. [10] demonstrated that the scheme given by Kim and Lee was still vulnerable to masquerade attack executed by new as well as legal nodes because it lacks hash chain renewability soon after the authentication and key established phase. Finally, Lee et al. [11] proposed a practical ACP based on EC and the hash chain. However, it was later observed that a large number of key distributions in Lee et al. [11] and Zhou [4] are also vulnerable to various adversary attacks and had hung storage overhead at the sensor node.

The concept of chameleon hash function was first given by Krawczyk and Rabin [12]. Chameleon hash function is used to calculate the message digest. A chameleon hash function is a basically trapdoor collision-resistant hash function. It is found to be a very useful tool in cryptography. In order to take such advantage of this function, Chen et al. [13] involved it in the access control protocol. However, the Chen et al. [13] protocol required the precomputed secret value of x^{-1} during the transection even without verifying the authentic value and thus invites attacks.

Motivated by the use of the double trapdoor chameleon hash function by Chen et al. [14], in this paper, we propose a secure and efficient ACP based on ECDLP. In our opinion, the proposed protocol which does not require the precomputed value of x^{-1} dynamically provides the security against different attacks, even when new nodes are added to the WSN. Looking to the other advantages, our proposed scheme is better as compared to the scheme given by Chen et al. [13].

The rest of the paper is organized as follows. In Section 2, we give preliminaries required for the proposed access control protocol. In Section 3, the proposed scheme is explained. The security and efficiency analysis of our proposed scheme is given in Section 4. Finally, the conclusion is made in Section 5.

2. Preliminaries

As we have said earlier, in this section, we first explain the requirements for the ACP of a wireless sensor network using the ECDLP and trapdoor chameleon hash function. Before doing so, we need to explain the notion of a trapdoor chameleon hash function as given by Chen et al. [15] scheme. Let us first recall the EC as given below.

2.1. Elliptic Curve. We consider the parameters of any EC such that the EC domain parameters can be verified to meet the requirements as given by Law et al. [16]. In order to avoid the Pollard-rho [17] and Pohlig-Hellman algorithms for the discrete logarithm problem defined on EC, it is necessary that the number of F_p-rational points on E, denoted by $\#E(F_p)$, be divisible by a sufficiently large prime n. Also, in order to avoid the reduction algorithms of Menezes et al. [18] and Frey and Rück [19], our EC should be nonsuper singular (i.e., p should not divide $(p+1-\#E(F_p))$). Further, in order to avoid the attack of Semaev [20] on F_p-anomalous curves, our EC should not be F_p-inconsistent (i.e., $\#E(F_p) \neq p$).

2.2. Elliptic Curve Discrete Logarithm Problem. Let E be an elliptic curve defined over a finite field F_p and let $P \in E(F_p)$ be a point of order n. Given Q, where $Q \in E(F_q)$, the ECDLP is used to find the integer l, $0 \leq l \leq n-1$, such that $Q = l \cdot P$.

2.3. Trapdoor Chameleon Hash Function. Following the ACP of Chen et al. [15], we define double trapdoor chameleon hash function as below.

Let G be a subgroup generated by P and define a cryptographic secure keyed-hash function $f : Z_q \times G \to Z_q$. Choose random elements (two trapdoor keys) $k, x \in_R Z_q$ and compute $K = kP$, $Y = xP$. The public hash key is $\mathrm{HK} = (K, Y)$, and the private trapdoor key is $\mathrm{TK} = (k, x)$. For the given hash family, we define the hash key HK and the proposed chameleon hash function $H_{\mathrm{HK}} : Z_q \times Z_q \to G$ as follows:

$$H_{\mathrm{HK}}(m, r) = f(m, K) \cdot (K + Y) + rP. \quad (1)$$

A double trapdoor chameleon hash function carries the following properties.

(1) *Efficiency.* Given a hash key pair HK and a pair $(m, r) \in Z_q \times Z_q$, $H_{\mathrm{HK}}(m, r) = f(m, K) \cdot (K + Y) + rP$ is computable in the polynomial time.

(2) *Collision Resistance.* Without the trapdoor key TK, it is computationally infeasible to find two pairs $(m_1, r_1), (m_2, r_2) \in Z_q \times Z_q$ which satisfy $m_1 \neq m_2$ and $H_{\mathrm{HK}}(m_1, r_1) = H_{\mathrm{HK}}(m_2, r_2)$.

(3) *Trapdoor Collision.* Assume that we have given the hash and the trapdoor key pair (HK, TK), a pair $(m_1, r_1) \in Z_q \times Z_q$, and an additional message $m_2 \in Z_q$, and we want to find $r_2 \in Z_q$ such that

$$\begin{aligned} & f(m_1, K) \cdot (K + Y) + r_2 Y \\ & \quad = f(m_2, K) \cdot (K + Y) + r_2 Y. \end{aligned} \quad (2)$$

The value of r_2 can be computed in polynomial time as follows: $r_2 = r_1 + (k + x)(f(m_1, K) - f(m_2, K)) \bmod q$.

Also, as r_1 is uniformly distributed in $\mathscr{R}$ then the distribution of r_2 is computationally indistinguishable from the uniformly distributed r_1 in $\mathscr{R}$.

2.4. Notations Used in the Proposed Scheme. The notations involved are listed as follows:

N_i: ith node.

N_j: jth node.

BS: base station.

l: integer number.

E: elliptic curve.

P: generator of subgroup G.

f: cryptography secure hash function.

r: random number.

H_{HK}: chameleon hash function.

A_u: authentication value.

3. Proposed Access Control Protocol Based on ECDLP

Now we propose our ACP based on ECDLP and double trapdoor chameleon hash function. This method consists of two phases: initialization phase and the node authentication with key establishment phase. The implementation of the proposed ACP is as follows.

3.1. Initialization Phase of the Proposed ACP. The initialization phase is described in the following steps.

Step 1. The base station (BS) chooses a random element $x \in_R Z_q$ and computes $Y = xP$. The public hash key is $\mathrm{HK} = Y = xP$ and the private trapdoor key is $\mathrm{TK} = x$.

Step 2. Choose a random number $k^* \in_R Z_q$, and compute the chameleon hash value $H_{\mathrm{HK}_{\mathrm{BS}}} = k^* P$.

Step 3. Given message m from pair $(k_i, k_i P)$, where $k_i \in Z_q$ as the secrete key and $i = 1, 2, 3, \ldots, n_1$, then compute a security key $r_i = k^* - f(m, k_i P)(k_i + x) \bmod q$, uploaded $(N_i, k_i P, r_i, H_{\mathrm{HK}})$ to node N_i.

Note. $r_i = k^* - f(m, k_i P)(k_i + x) \bmod q$. It requires only 1 modular multiplication of Z_q in this phase.

3.2. Authentication with Key Establishment Phase of ACP. In this section, we give different steps of authentication of the proposed ACP.

In all the sensor nodes when deployed, if node N_i wants to communicate with another node N_j, they must implement the following steps to authenticate each other. Subsequently, they must establish a shared session key for securing their communication.

Step 1. Two nodes are N_i and N_j, where $i = 1, 2, 3, \ldots, n_1$ and $j = 1, 2, 3, \ldots, n_2$, for $n_1, n_2 \in n$, and node N_i chooses random number $c_1 \in Z_q$ to compute the public key $c_1 P$ and $(K + Y) r_i$ and then sends $(N_i, c_1 P, (K + Y) r_i, k_i P)$ to node N_j.

Step 2. Node N_j computes the chameleon hash value H_{HK} of node N_i based on the received message $(N_i, c_1 P, (K + Y) r_i, k_i P)$. If $H_{\mathrm{HK}_{\mathrm{BS}}}$ and $H'_{\mathrm{HK}_{\mathrm{BS}}}$ are equal, then node N_j chooses random number $c_2 \in Z_q$ to compute $c_2 P$ and session key $(c_1 c_2)^x P = (c_2 c_1)^x P$ between nodes N_i and N_j. Then node N_j uses different security key r_j to compute authentication value $A_{u_j} = H_{\mathrm{HK}}((c_1 c_2)^x P \parallel r_j (K + Y))$. It then delivers the message $(N_j, c_2 P, r_j (K + Y), k_j P, A_{u_j})$ to node N_i.

Step 3. Node N_i receives the message from N_j and computes chameleon hash value of node N_j and according to the message $(N_j, c_2 P, r_j (K + Y), k_j P, A_{u_j})$ from N_j, it then computes $H'_{\mathrm{HK}_{\mathrm{BS}}} = f(m, K)(K + Y) + r_i P$ with the chameleon hash of base station $H_{\mathrm{HK}_{\mathrm{BS}}}$. If $H_{\mathrm{HK}_{\mathrm{BS}}} = H'_{\mathrm{HK}_{\mathrm{BS}}}$, node N_i then computes the share session key and $(c_1 c_2)^x P = (c_2 c_1)^x P$ the authentication value A'_{u_j}, where $A'_{u_j} = H_{\mathrm{HK}}((c_1 c_2)^x P \parallel r_j (K + Y))$.

Again, node N_i checks the authentication value A_{u_j}; if $A_{u_j} = A'_{u_j}$ then node N_j is valid and goes back to a authentication value for given r_i and $(c_1 c_2)^x P$, where $A_{u_i} = H_{\mathrm{HK}}((c_1 c_2)^x P \parallel r_i (K + Y))$.

Step 4. Node N_j receives A_{u_i}; it also computes the value $A'_{u_i} = H_{\mathrm{HK}}((c_1 c_2)^x P \parallel r_i (K + Y))$. If $A_{u_i} = A'_{u_i}$ then node N_i is authenticated; otherwise, the value A_{u_i} is discarded. Same method applies for node N_j, if $A_{u_j} = A'_{u_j}$ is authenticated; otherwise value A_{u_j} is discarded.

New Node Addition Phase. During the network communication phase, if some sensor nodes are lost, new sensor nodes are needed to deploy. When a new node with N_{i+1} is added, the base station also generates a secret key k_{i+1} and then the base station computes the chameleon hash value $H_{\mathrm{HK}_{\mathrm{BS}}} = f(m, k_{i+1} P)(k_{i+1} P + Y) + r_{i+1} P$ at node N_{i+1} and update as broadcasting chameleon hash value $H_{\mathrm{HK}_{\mathrm{BS}}}$ in the base station. The authentication and key establishment for any old node with the new node N_{i+1} is the same as authentication steps.

3.3. Correctness of the Proposed ACP. In order to show the correctness of our proposed ACP, we assert that, during the authentication with key establishment phase, node N_j authenticates node N_i based on the chameleon hash value of node N_i; that is, it computes the value of $H_{\mathrm{HK}_{\mathrm{BS}}} = f(m, K)(K + Y) + r_i P$ based on the received message $(N_i, c_1 P, r_i (K + Y), k_i P)$ from node N_i and publishes the message of the base station which is written as $H_{\mathrm{HK}_{\mathrm{BS}}} = f(m, k_i P)(k_i P + Y) + k^* P - f(m, k_i P)(k_i P + xP) = k^* P$ the chameleon hash value (see Box 1).

4. Security Analysis

For the purpose of analyzing the security aspect of our proposed ACP, we claim that attacker can not find the authentication value for communication node between N_i and N_j. These nodes require authentic value of the message to be communicated from N_i to N_j. First we ascertain that node N_i has been authenticated by node N_j using the chameleon hash value

$$H_{\mathrm{HK}_{\mathrm{BS}}} = f(m, k_{i+1} P)(k_{i+1} P + Y) + r_{i+1} P \tag{3}$$

and then computes the authentication value A_{u_j} corresponding to A_{u_i}. The authentication value A_{u_i} is obtained by the shared session key and the security key r_i. However, only the communication nodes accept the session key $c_1 c_2 P$, and the only node N_i and the base station can have the security key r_i.

Second, node N_j is preloaded with the chameleon hash value by the base station $H_{\mathrm{HK}_{\mathrm{BS}}}$ along with node N_i and obtained $H'_{\mathrm{HK}_{\mathrm{BS}}}$. However, the computed value of $H'_{\mathrm{HK}_{\mathrm{BS}}}$ needs some value of identity ID, secure hash key $k_i P$, and security key r_i of node N_i. This way, the process can authenticate ID and the hash key because computing $H'_{\mathrm{HK}_{\mathrm{BS}}}$ is an elliptic curve discrete logarithm problem and attacker can not find

Node N_i | Node N_j

Generate $c_1 \in Z_n^*$

Compute $c_1P, (K+Y)r_i$

$$\xrightarrow{N_i, c_1P, (K+Y)r_i, k_iP}$$

Choose c_2

Compute $H_{\mathrm{HK}}, c_1P, (K+Y)r_i, k_iP$, and

$A_{u_j} = H_{\mathrm{HK}}((c_1c_2)^xP \parallel r_j(K+Y))$

Check $H_{\mathrm{HK}_{\mathrm{BS}}} ? = H'_{\mathrm{HK}_{\mathrm{BS}}}$

$$\xleftarrow{N_j, c_1P, (K+Y)r_j, k_jP, A_{u_j}}$$

Check $H_{\mathrm{HK}_{\mathrm{BS}}} ? = H'_{\mathrm{HK}_{\mathrm{BS}}}$

Compute $(c_1c_2)^xP$,

$A'_{u_j} ? = H_{\mathrm{HK}}((c_1c_2)^xP \parallel r_j(K+Y))$

$A_{u_i} = H_{\mathrm{HK}}((c_1c_2)^xP \parallel r_i(K+Y))$

$$\xrightarrow{A_{u_i}}$$

Check $A'_{u_i} ? = H_{\mathrm{HK}}((c_1c_2)^xP \parallel r_i(K+Y))$

Box 1

any information about ID and hash key. On the other hand, even if attacker successfully finds out the security key r_i then also he can not know the secret values x and k^* because of its trapdoor chameleon hash value. Only the authorized user can find out the secret key.

In addition, we claim that the proposed ACP is able to resist the attacks such as forgery attacks, legal node masquerading attacks, new node attack, replay attacks, man-in-the-middle attacks, and session key security attack as given below.

(1) *Forgery Attack*. Say, an attacker tries to obtain the commutation values by eavesdropping on the communication channel as

$$
\begin{aligned}
r_i' &= r_i - (k_i + x)\left(f\left(m', k_iP\right) - f\left(m, k_iP\right)\right) \mod q \\
r_i' &= k^* - f\left(m, k_iP\right)\left(k_i + x\right) - f\left(m', k_iP\right)\left(k_i + x\right) \\
&\quad + f\left(m, k_iP\right)\left(k_i + x\right) \mod q \\
r_i' &= k^* - f\left(m', k_iP\right)\left(k_i + x\right) \mod q.
\end{aligned}
\tag{4}
$$

But it is not possible for him because the value of r_i' cannot be computed without secret key $(k_i + x)$.

(2) *Legal Node Masquerading Attacks.* Under this attack, the attacker has to deploy a pseudonode by removing the legal one. For this purpose, attacker has to obtain the commutation values by eavesdropping on the communication between nodes N_i and N_j. However, even if the attacker obtains the values of c_1P and c_2P from the authentication and key establishment phase, then also, deriving the legalized session key c_1c_2P is extremely difficult to obtain because of the security tool employed as ECDLP. In other words, the legal node N_j is well equipped with the security key $r_j(K+Y)$ provided by the base station $H_{\mathrm{HK}_{\mathrm{BS}}}$ which attacker can not retrieve.

(3) *New Node Masquerading Attacks*. Under this attack, when some sensor node is lost, it needs to be replaced by new sensor node N_{i+1}. To take advantage of this situation, the attacker may try to know the secrete keys x and k^* from the new node. But, this is not possible because the secret keys are provided by the base station to the new node with chameleon hash values $H_{\mathrm{HK}_{\mathrm{BS}}}$ and r_{i+1} which attacker can not compute.

(4) *Replay Attack*. In this attack, the adversary first eavesdrops on the communication between two communicating entities and then tries to impersonate the legal authentic message by simply replacing the other messages to the dedicated entity. For example, when an attacker transfers the message $(N_i, c_1P, r_i(K+Y), k_iP)$ to another node N_j, the attacker provides r_i for establishing authentication value A_{u_i}. A_{u_i} is required for shared session key with the node to be connected. It is not possible for the attacker to obtain r_i without x and k^* which is the trapdoor secret value and available at the base station only. On the other hand, if the attacker sends the authenticated value A_u to node N_j, he can use the shared session key to authenticate, whether the connecting node is legitimate or not; if the node is legitimate then process is to proceed for the next step, otherwise discard, because the authenticated node uses up-to-date session keys c_1 and c_2 in order to apply the different strategies. Hence our proposed ACP successfully resists the replay attack.

(5) *The Man-in-the-Middle Attack*. This is one of the classical attacks that can be executed in any WSN environment. However, in any WSN equipped with our proposed ACP, the communication nodes can authenticate and establish the session keys between the users and the server. If attacker wants to mount the man-in-the-middle attack, he only knows the public keys c_1P and c_2P and wants to solve the ECDLP.

Table 1: Computational cost of the proposed protocol.

Phases	Base station	Node N_i
Initialization	$3T_{e_m} + 1T_h + 1T_{\text{mul}}$	Not applicable
Authentication and key establishment	Not applicable	$4T_{e_m} + 1T_h$
New node adding	$2T_{e_m} + 1T_{\text{mul}}$	Not applicable

Table 2: Comparison of computation cost with other protocols.

Protocol	Authentication and establishment phase
Zhou et al. [4]	$3T_{e_m} + T_{\text{mul}} + T_h$
Kim and Lee [8]	$4T_{e_m} + 8T_h$
Huang [7]	$2T_{e_m} + 5T_h$
Lee et al. [11]	$2T_{e_m} + 5T_h$
Our scheme	$4T_{e_m} + 1T_h$

Even if the attacker obtains the user's information $(N_i, k_iP, (K + Y)r_i)$, then also the attacker cannot pass the authentication and key establishment phase, because he cannot compute the session key A_u. Hence, our ACP can resist man-in-middle attack.

(6) *Session Key Security*. Our proposed ACP is well equipped with the session key security feature. Since only the communicating parties know the session key c_1c_2P and hence are aware of the security of the session key, consequently, they can only verify the user of the message. The session key c_1c_2P is not known to anyone because random values c_1P and c_2P are protected by the ECDLP. Therefore, the proposed ACP provides session key security as an additional feature.

4.1. Efficiency. The computational cost of proposed ACP is calculated in Table 1 at different phases and these are compared with other such schemes in Table 2. For this purpose, in Table 1, we have first given the computational cost of our ACP for three phases at base station and at node N_i considering the elliptic curve and hash chain components as below.

The notations we use in Tables 1 and 2 for the purpose of comparison are as follows:

T_{e_m}: one multiplication computation over an elliptic curve.

T_h: cryptographic secure hash function.

T_{mul}: modulus multiplication operation.

The total computational cost of proposed ACP is $4T_{e_m} + 1T_h$ during the authentication and key establishment phase at node N_i and its computational cost is $3T_{e_m}+1T_h+1T_{\text{mul}}$ during the base station and $2T_{e_m} + 1T_{\text{mul}}$ is the computational cost at the new node addition phase in Table 1.

Next, in Table 2, we have shown the comparison of the computational cost of our proposed ACP with Zhou et al. [4], Kim and Lee [8], Huang [7], and Lee et al. [11] scheme during authentication and key establishment phase.

Table 3: Comparison of time consumed with other protocols.

Protocol	Time consumed during authentication phase
Zhou et al. [4]	2.56 CPU time
Kim and Lee [8]	1.57 CPU time
Huang [7]	0.895 CPU time
Our scheme	0.256 CPU time

From Table 2, it is evident that the proposed ACP has the lowest computational cost $4T_{e_m} + 1T_h$ as compared to other schemes.

Finally, we compare the time consumed at authentication phase during data transmission in CPU device with other schemes using Mathematica 7.0, shown in Table 3.

From Table 3, it is evident that the proposed ACP takes 0.256 seconds in CPU time which is less as compared to other protocols.

5. Conclusion

From the aforesaid sections, we conclude to say that our proposed ACP using the double trapdoor function and whose security is based on ECDLP is best suited to any WSN environment. The reason for being more secured is that it can resist many known attacks such as masquerading, replay, man-in-the-middle, and forgery attacks and has a special feature known as session key security and as shown in Tables 1, 2, and 3 it is more efficient as compared to many other existing protocols.

Competing Interests

The author declares that there is no conflict of interests.

References

[1] A. Abduvaliev, S. Lee, and Y.-K. Lee, "Simple hash based message authentication scheme for wireless sensor networks," in *Proceedings of the 9th IEEE International Symposium on Communications and Information Technology (ISCIT '09)*, pp. 982–986, Incheon, South Korea, September 2009.

[2] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Communications Magazine*, vol. 40, no. 8, pp. 102–114, 2002.

[3] I. F. Akyildiz and I. H. Kasimoglu, "Wireless sensor and actor networks: research challenges," *Ad Hoc Networks*, vol. 2, no. 4, pp. 351–367, 2004.

[4] Y. Zhou, Y. Zhang, and Y. Fang, "Access control in wireless sensor networks," *Ad Hoc Networks*, vol. 5, no. 1, pp. 3–13, 2007.

[5] N. Koblitz, "Elliptic curve cryptosystems," *Mathematics of Computation*, vol. 48, no. 177, pp. 203–209, 1987.

[6] V. Miller, "Uses of elliptic curves in cryptography," in *Proceedings of the Workshop on the Theory and Application of Cryptographic Techniques (EUROCRYPT '85)*, vol. 218 of *Lecture Notes in Computer Science*, pp. 417–426, Linz, Austria, 1986.

[7] H.-F. Huang, "A novel access control protocol for secure sensor networks," *Computer Standards and Interfaces*, vol. 31, no. 2, pp. 272–276, 2009.

[8] H.-S. Kim and S.-W. Lee, "Enhanced novel access control protocol over wireless sensor networks," *IEEE Transactions on Consumer Electronics*, vol. 55, no. 2, pp. 492–498, 2009.

[9] J. Shen, M. Sangman, and C. Ilyong, "Comment: enhanced novel ACP over wireless sensor networks," *IEEE Transactions on Consumer Electronics*, vol. 56, no. 3, pp. 2019–2021, 2010.

[10] P. Zeng, K.-K. R. Choo, and D.-Z. Sun, "On the security of an enhanced novel access control protocol for wireless sensor networks," *IEEE Transactions on Consumer Electronics*, vol. 56, no. 2, pp. 566–569, 2010.

[11] H. Lee, K. Shin, and D. H. Lee, "PACPs: practical access control protocols for wireless sensor networks," *IEEE Transactions on Consumer Electronics*, vol. 58, no. 2, pp. 491–499, 2012.

[12] H. Krawczyk and T. Rabin, "Chameleon hashing and signatures," in *Proceedings of the Network and Distributed Systems Symposium (NDSS '00)*, pp. 143–154, San Diego, Calif, USA, February 2000.

[13] C.-Y. Chen, A. D. Yein, T.-C. Hsu, J. Y. Chiang, and W.-S. Hsieh, "Secure access control method for wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2015, Article ID 261906, 6 pages, 2015.

[14] X. Chen, F. Zhang, W. Susilo, and Y. Mu, "Efficient generic on-line/off-line signatures without key exposure," in *Applied Cryptography and Network Security*, vol. 4521 of *Lecture Notes in Computer Science*, pp. 18–30, Springer, Berlin, Germany, 2007.

[15] X. Chen, F. Zhang, H. Tian et al., "Efficient generic on-line/off-line (threshold) signatures without key exposure," *Information Sciences*, vol. 178, no. 21, pp. 4192–4203, 2008.

[16] M. Q. J. S. L. Law, A. Menezes, and S. Vanstane, "An efficient protocol for authenticated key agreement," *Codes and Cryptography*, vol. 28, no. 2, pp. 119–134, 2003.

[17] J. M. Pollard, "Monte carlo methods for index computation mod p," *Mathematics of Computation*, vol. 32, no. 143, pp. 918–924, 1978.

[18] A. J. Menezes, T. Okamoto, and S. A. Vanstone, "Reducing elliptic curve logarithms to logarithms in a finite field," *IEEE Transactions on Information Theory*, vol. 39, no. 5, pp. 1639–1646, 1993.

[19] G. Frey and H.-G. Rück, "A remark concerning m-divisibility and the discrete logarithm in the divisor class group of curves," *Mathematics of Computation*, vol. 62, no. 206, pp. 865–874, 1994.

[20] I. A. Semaev, "Evaluation of discrete logarithms in a group of p-torsion points of an elliptic curve in characteristic p," *Mathematics of Computation*, vol. 67, no. 221, pp. 353–356, 1998.

Journal of Engineering
The Scientific World Journal
International Journal of Rotating Machinery
Journal of Sensors
International Journal of Distributed Sensor Networks
Advances in Civil Engineering
Journal of Control Science and Engineering
Journal of Robotics
Journal of Electrical and Computer Engineering
Advances in OptoElectronics
VLSI Design
International Journal of Navigation and Observation
Modelling & Simulation in Engineering
International Journal of Aerospace Engineering
International Journal of Chemical Engineering
International Journal of Antennas and Propagation
Active and Passive Electronic Components
Shock and Vibration
Advances in Acoustics and Vibration