Public Key Cryptography - based Security Scheme for Wireless Sensor Networks in Healthcare

Xuan Hung Le[1], Ravi Sankar[1], Murad Khalid[1], Sungyoung Lee[2,*]
[1]University of South Florida, 4202 E Fowler Ave, Tampa, FL 33620, USA
[2]Kyung Hee University, Seocheon-dong, Giheung-gu, Yongin-si, Gyeonggi-do, 449-701, Korea
Email: xhle@eng.usf.edu, sankar@eng.usf.edu, mkhalid@mail.usf.edu, sylee@oslab.khu.ac.kr

ABSTRACT
The application of wireless sensor networks (WSNs) in healthcare is one of the most important and rapidly growing areas. One of the most critical security concerns is patients' privacy. Since patients are monitored all the time, authentication of who can access the information, and what information one is authorized to access are indispensable to maintain privacy. In healthcare environments, authentication and access control face a big challenge due to dynamic network topology, mobility, and stringent resource constraints. In this paper, we propose a secure, scalable, and energy-efficient security scheme called Mutual Authentication and Access Control scheme based on Elliptic Curve Cryptography (MAACE). MAACE provides mutual authentication where a healthcare professional can authenticate to an accessed node (a PDA or medical sensor) and vice versa. This is to ensure that medical data is not exposed to an unauthorized person. On the other hand, it ensures that medical data sent to healthcare professionals did not originate from a malicious node. By applying elliptic curve cryptography (ECC), MAACE provides a public key approach which is more scalable and requires less memory compared to symmetric key-based schemes. Furthermore, it is practically feasible to implement it on sensor platforms. Security analysis and performance evaluation results are presented and compared to existing schemes to show advantages of the proposed scheme.

Categories and Subject Descriptors
C.2.1 [Network Architecture and Design]: Wireless Communication; C.2.0 [General]: Security and Protection

General Terms
Algorithms, Security

This research was supported by the MKE (Ministry of Knowledge Economy), Korea, under the ITRC (Information Technology Research Center) support program supervised by the NIPA (National IT Industry Promotion Agency) (NIPA-2009-(C1090-0902-0002)). Also, it was supported by the IT R&D program of MKE/KEIT, [10032105, Development of Realistic Multiverse Game Engine Technology].

*Dr. Sungyoung Lee is the corresponding author.

ICUIMC'10, January 14–15, 2010, Suwon, Korea

Keywords
Security, Elliptic Curve Cryptography, Authentication, Access Control, Sensor Networks, Healthcare

1. INTRODUCTION
Over the past few years, extensive research efforts have focused on developing wireless sensor networks (WSNs) for healthcare applications. Most of the research work has mainly focused on how to seamlessly collect and wirelessly transmit health data (e.g. vital signs) in the presence of extreme resource-limitations in terms of power, computation, and bandwidth [2]-[10]. Security is an important factor for WSN's success and acceptance in medical applications. One of the most critical security concerns is how to maintain patients' privacy which requires secure authentication and access control. In healthcare environments, authentication and access control face a big challenge due to dynamic network topology, mobility of nodes, and resource constraints. Public key cryptography-based schemes are ideal to overcome these challenges due to their high scalability, low memory requirements, easy key-addition/revocation for a new node, and no requirement of complicated key pre-distribution [18][20]. However, it is computationally expensive to apply public key cryptography to such resource-limited devices like sensors [27].

Authentication is to allow legitimate healthcare professionals to conveniently access monitored information while declining malicious persons or attackers. After authentication, access control takes charge to restrict authenticated healthcare professionals to access only data that they have privilege for proper healthcare services. In this paper, we propose a new method, Mutual Authentication and Access Control based on Elliptic Curve Cryptography (MAACE) that provides mutual authentication (a healthcare professional can authenticate to an sensor node and vice versa) and ensures a healthcare professional can only access data that he/she has privilege. By applying elliptic curve cryptography, MAACE provides a public key approach based on Elliptic Curve Cryptography (ECC) which is more scalable and requires less memory compared to symmetric key – based schemes. Furthermore, although MAACE is public key – based scheme, it is practically feasible to implement it on sensor platforms.

The remainder of the paper is organized as follows. Section 2 briefly reviews related work. Background of Elliptic Curve Cryptography, which forms the foundation for this proposed method, is described in Section 3. The proposed security scheme is presented in Section 4. In Section 5, security analysis and performance evaluation are given and compared with the existing approaches. Finally, Section 6 concludes the paper and outlines investigation for future work.

2. RELATED WORK

Over the last decade, there have been a number of security schemes proposed for WSNs [18]-[25]. These schemes have solved the problem of how to pre-distribute pair-wise shared keys (symmetric keys) to a large number of nodes which are scattered over a large field. Most of them have not taken into account challenges in healthcare domain. For example, [21]-[23] are based on node deployment knowledge (i.e. node location information) to efficiently and securely pre-distribute key rings to a number of group. In healthcare environments, node locations are not fixed. Furthermore, node location retrieval and frequent location updates increase network overhead and energy consumption significantly.

Want *et al.* [20] (HBQ scheme) and Le *et al.* [18] (ENABLE scheme) apply public key cryptography based on ECC to solve the problem of symmetric key approaches in terms of scalability, key storage, and key pre-distribution. However, the performance evaluation in [20] has shown that HBQ is still burdensome for sensors leading to impracticability of implementation. Although ENABLE [18] has solved security limitations and performance issues in [20], it relies on a trusted third-party (e.g. Key Distribution Scheme) to handle significant ECC operations. This introduces significant cost increase in healthcare.

Recently, a few papers have focused on secure healthcare sensor networks. Ng *et al.* came up with new interesting security issues of wireless sensor networks in healthcare applications [1]. Authors discuss the unique challenges of security implementation in healthcare such as resource limitations of sensor nodes, uncontrollable environment, and dynamic network topology. In [10], the authors introduce a hierarchical network for in-home, in-hospital, nursing-house healthcare applications. The sensor network tier uses *BTnode* (Bluetooth-enable node) and relies on Bluetooth security. Since many current sensors are built on Zigbee standard (e.g. CodeBlue [2][3]), the proposed scheme lends itself to be impractical. Boukerche and Yonglin [11] propose a secure mobile healthcare system using trust-based multicast system. The authors presented a secure multicast strategy that employs trust in order to evaluate the behavior of each node so that only trustworthy nodes are allowed to participate in communications, while the misbehavior of malicious nodes is effectively prevented. Chakravorty [12] introduces a health-related service architecture (MobiCare) for mobile patient care. It satisfies the need of medical monitoring by deploying medical sensors to form a body sensor network, and also provides the necessary protection to clinical services by applying secure and reliable dynamic software. The author further discusses issues with MobiCare, which include confidentiality, integrity, and privacy of patient's information. Many techniques are suggested, such as authentication, access control, encryption, and so on.

Kim *et al.* [13] discuss some potential threats for ubiquitous healthcare systems and describe the security requirements for these u-healthcare systems. They propose a systematic architecture in order to design a security policy for such healthcare systems and to allow a patient to control access to any sensing data recorded by a personal healthcare device. Bao *et al.* [14] propose an interesting scheme that would solve the issue of entity authentication for BSN, in which the notion of biometrics is applied as an authentication approach that automatically verifies an individual's identity. In the established BSN, peer authentication can ensure secure connections between different entities. This method is however only designed for wearable biometric sensors. Jeong *et al.* [15] present a mobile collaboration framework based on distributed systems. It supports the necessary security services by checking access rights for corresponding users. It then divides the collected data into secure and public data, and subsequently applies the access control technique to specify that each security object needs the corresponding access privilege.

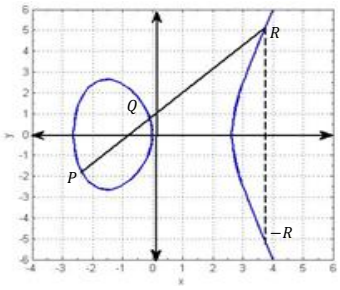
Marti *et al.* [16] present a specification of integrated network and security services for mobile e-health environments. It applies different security mechanisms to address threats such as eavesdropping or manipulating patient information, and thus guarantees the patient data confidentiality and integrity. Markovic *et al.* [17] consider the issues of mobile healthcare security and employ cryptographic techniques to address possible vulnerabilities. They make use of symmetrical cryptographic methods to protect data confidentiality, and asymmetrical cryptographic algorithms such as Public Key Infrastructure (PKI) and digital signature technique to achieve data integrity. PKI is the most preferable solution in healthcare, but their technique is applied to powerful computing systems.

To sum up, a new security method is needed for WSNs that will address the key challenges faced in healthcare applications specifically, secured authentication and access control for patient's privacy. Our approach applies PKI based on ECC to solve the performance problem of PKI in WSNs.

3. ECC BACKGROUND

3.1 Overview

Elliptic Curve Cryptography (ECC) is an approach of public-key cryptography based on the algebraic structure of elliptic curves over finite fields. The use of elliptic curves in cryptography was suggested independently by Miller [29] and Koblitz [30] in 1985. In recent years, ECC has attracted much attention as a security solution for wireless networks due to the small key size and low computational overhead.



$P(-2.35, -1.86)$
$Q(-0.1, 0.836)$
$-R(3.89, 5.62)$
$R(3.89, -5.62)$

$P + Q = R = (3.89, -5.62)$

$\boldsymbol{y^2 = x^3 - 7x}$

Figure 1 Elliptic curve and point addition

An elliptic curve is a plane curve which consists of the points satisfying the equation:

$$y^2 = x^3 + ax + b,$$

where x, y, a and b are elements in $GF(q)$ (a *Galois Field* of order q, where q is a prime).

Each choice of (a, b) yields a different elliptic curve. For example, Figure 1 shows an elliptic curve of $y^2 = x^3 - 7x$.

The elliptic curve group operation is closed under addition so that addition of any two points is also a point in the group. Given two

points $P(x_1, y_1)$ and $Q(x_2, y_2)$, the addition results in a point $R(x_3, y_3)$ given by:

$$(x_1, y_1) + (x_2, y_2) = (x_3, y_3).$$

such that

$$x_3 = \Psi^2 + \Psi + x_1 + x_2 + a$$

$$y_3 = \Psi(x_1 + x_3) + x_3 + y_1$$

where $\Psi = (y_1 + y_2)/(x_1 + x_2)$

An example of $P(-2.35, -1.86)$ and $Q(-0.1, 0.836)$ is illustrated in Figure 1.

If $P = Q$, then $R = P + P = 2P$. Addition of multiple points P will give $R = kP$. ECC relies on the difficulty of the Elliptic Curve Discrete Logarithm Problem (ECDLP), that is, given points P and Q of the group, it is practically infeasible to find a number k such as $Q = kP$.

3.2 Elliptic Curve Diffie-Hellman Protocol

Based on ECDLP, a typical *Elliptic Curve Diffie-Hellman* (ECDH) key-exchange protocol is built as shown in Figure 2. Initially, Alice and Bob agree on system based point P and generate their own public key Q_A and Q_B. To share a secret, Alice and Bob exchange their public keys and then use their own private key to multiply the other's public key. Since $R = k_A \times Q_B = k_A \times (k_B \times P) = k_B \times (k_A \times P) = k_B \times Q_A)$, the resultant point R will be the secret of Alice and Bob.

The protocol is secure because nothing is disclosed (except for the public keys, which are not secret), and no party can derive the private key of the other unless it can solve the Elliptic Curve Discrete Logarithm Problem.

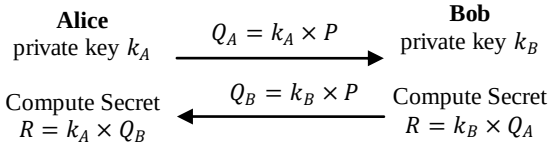


Figure 2 ECDH key exchange protocol.

4. PROPOSED SECURITY SCHEME

4.1 Network Model

A typical sensor network in healthcare is shown in Figure 3. We define it as a hierarchical network consisting of three layers: *Sensor Network* (SN) layer, *Coordination Network* (CN) layer, and *Back-end Network* (BN) layer. Figure 4 shows the abstract hierarchical structure.

- **Sensor Network (SN) Layer**: In SN layer, different types of medical sensors are wearable on a human body to monitor health status such as blood pressure, electrocardiogram (EKG), heart rate, blood oxygen saturation (SpO_2). Furthermore, embedded sensors are also deployed in indoor areas (e.g. patient's home, hospital ward) to monitor environmental conditions which is necessary for healthcare services. These sensors use either ZigBee (IEEE 802.15.4) or Bluetooth (IEEE 802.15.1) wireless technology. Since these sensors have a short communication range (10 - 100 *m*), they must be connected to more powerful devices in CN layer so as to deliver sensed data to healthcare professionals. They may communicate with each other to exchange and deliver sensed data.
- **Coordination Network (CN) Layer**: In CN layer, a number of mobile computing devices such as *Personal Digital Assistant* (PDA), laptop, cell phone, are organized regionally using an ad hoc network or an infrastructure-based network to connect to a fixed remote or local station. CN nodes collect and analyze data from SN layer because SN node does not have mass data storage capability over a long period of time (such as a few months or years). Further, CN nodes are tamper-resistant.
- **Back-end Network (BN) Layer**: The BN layer includes a number of fixed stations and servers which are structured on the Internet to provide application-level services. The server-side database stores physical records for long-term periods from the monitored individuals along with their residence environmental data. A third-party, a *Key Distribution Center* (KDC), set up on the Internet can be trusted to open access areas such as hospitals or nursing homes supporting the proposed healthcare monitoring service. The third party issues effective certificates and keys to valid SN and CN nodes.

4.2 Mutual Authentication and Access Control based on ECC (MAACE)

The first step is to establish key between nodes. To meet scalability requirements for a large number of sensor nodes, we propose a public key management scheme based on Elliptic Curve Cryptography (ECC). Compared to symmetric key cryptography, ECC is more scalable, requires lesser memory for storing keys, introduces low communication overhead, and is easy to deploy [18]. Furthermore, ECC requires much less computational cost and key length compared to conventional public key cryptosystems (e.g. *RSA* [26]). It has been proven that ECC with 160-bit key length has equivalent security level compared to RSA with 1024-bit key length [28]. On the other hand, ECC multiplication operation has been proven feasible on a sensor mote that takes only 0.81 *second* on 8-bit CPU Atmel ATmegal128 MHz [27]. The proposed key management for SN, CN, and BN layers is based on ECC [18].

4.2.1 ECC Key Management

4.2.1.1 ECC Key Distribution

There is one or more trusted third-parties on the network called *Key Distribution Center* (KDC) to generate all security materials (e.g. keys, certifiates), issue and revoke users's access privileges. Note that this KDC is not required to be online all the time like in ENABLE scheme [18]. Initially, KDC selects a particular elliptic curve over a finite field $GF(p)$ (where p is a prime) and publishes a base point P with a large order q (where q is also a prime). It picks a random number $x \in GF(p)$ as a private key, and publishes its corresponding public key $Q = x \times P$. It also generate a random number $x_i \in GF(p)$ as a private key for a sensor s_i and generate a corresponding public key $Q_i = x \times P$. The key-pair $\{x_i, Q_i\}$ is then loaded to s_i. For each node in CN and BN layers, it generates this key-pair based on the base P by itself since it is more powerful than a sensor node. After this step, every node in the network has an ECC key-pair which will be used to establish secret (symmetric) key for secure communication.

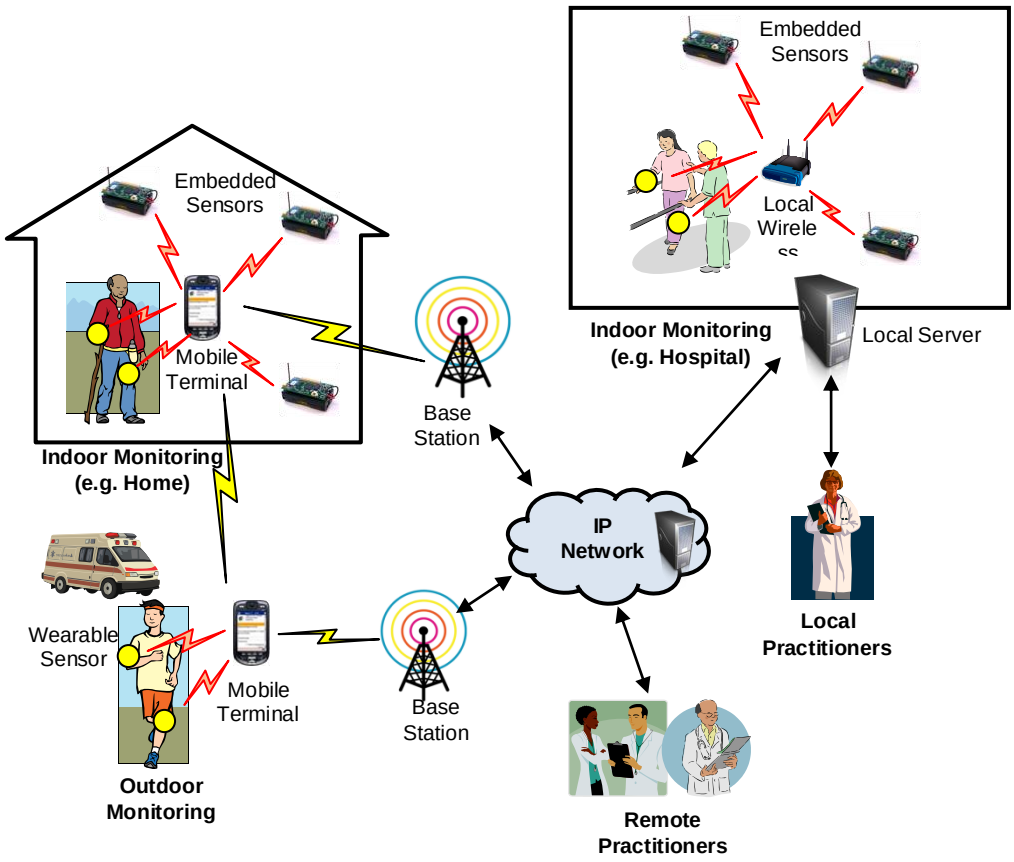


Figure 3 Typical network topology of wireless sensor networks in Healthcare application

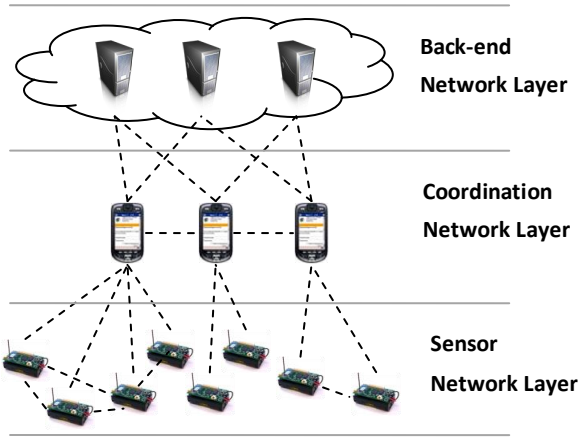


Figure 4 Hierarchical structure of a WSN in Healthcare

4.2.1.2 ECC Key Agreement

The proposed scheme is based on Elliptic Curve Diffie-Hellman (ECDH) [31] to establish a shared secret key between two nodes. ECDH is a key agreement protocol allowing two parties to establish a shared secret key that can be used for symmetric key cryptography. Suppose two nodes, say A and B, want to establish a secret shared key. They first exchange their public key Q_A and Q_B via an unsecured channel. Then A will compute $R_A = (x_A, y_A) = k_A \times Q_B$. B will compute $R_B = (x_A, y_B) = k_B \times Q_A$. Since $k_A \times Q_B = k_A \times k_B \times P = k_B \times Q_A$, therefore $R_A = R_B$, hence $x_A = x_B$. As the result, x_A is used as a shared secret key between node A and B.

4.2.2 Authentication and Access Control Protocol

MAACE is based on our previous work (ENABLE [18]) with significant modification to adapt to healthcare environment. ENABLE has been shown to provide a significant improvement over existing approaches (e.g. it is 184 times less energy consumption than HBQ scheme [19]). However, it requires that authenticating node has to communicate with a KDC to verify an access request from a user. In healthcare scenarios, communicating with a KDC introduces a significant amount of delay, network congestion, and energy-inefficiency. This is the impetus for developing MAACE to improve ENABLE that can adapt to healthcare applications while still retaining all its advantages. Notations are explained in Table 1.

We consider a situation that a medical practitioner or a healthcare server (generally called *Alice*, or A) wants to access data from a particular sensor, a group of sensors, or data on the coordination node. Prior to accessing data, *Alice* obtains the base P from a KDC and generates her private key (k_A) and public key $Q_A = k_A \times P$. She also requests an access permission list from KDC. Based on her background check, KDC issues a proper access control list ac_A. The list has similar structure as ENABLE scheme

[18] (see Figure 5). It is typically composed of *uid*, *gid*, and *user access privileges mask*. *uid* is a unique number to identify the user. *gid* is a group identification. *user access privilege mask* is a set of binary bits. Each bit represents a specific information or service. KDC generates a certificate of the user's access list and public key by signing with its private key ($cert_A = sign_{KDC}(ac_A \,||\, Q_A)$. The certificate is then sent to *Alice*.

Table 1 Notation

Symbol	Description
ID_A	Identifier of entity *A*
x_{AB}	Shared secret key between *A* and *B*
ac_A	Access control list issued to entity *A*
$sign_A(m)$	Message *m* is signed by entity *A*
$A \rightarrow B : m$	Entity *A* sends entity *B* a message *m*
$(m)K$	Symmetric encryption of message *m* with key *K*
$MAC\ (K, m)$	A message authentication code of message *m* with key *K*
$h(m)$	Hashing value of message *m*
$\|\|$	Concatenation

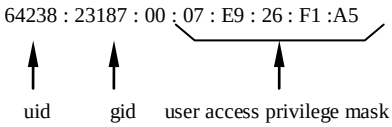


Figure 5 An example of user access control list

MAACE scheme is described in Figure 6 which includes the following steps.

- **Step 1**: $Alice \rightarrow C: (r)L, T_A, S_A$

Alice selects a random number $r \in GF(p)$ which will be used as a session key with C and S, creates a secret key $L = h(x_{AC} \oplus T_A)$ (where T_A is the current timestamp generated by *Alice*), and encrypts r with the key L (i.e. $(r)L$). *Alice* then signs this encrypted value along with its certificate (i.e. $S_A = sign_A((r)L \,||\, cert_A)$) and sends a combination $(r)L, T_A, S_A$ to the sensor *S*.

- **Step 2**: $C \rightarrow S: (r)M, T_C, ID_A, MAC_1$

Upon receiving the message from *Alice*, node C first checks if the timestamp T_A is valid (i.e. by verifying if $T_A < T_{now}$, where T_{now} is current timestamp). Then it verifies *Alice'* signature S_A. If valid, then *Alice* is authentic to C. *Alice's* certificate $cert_A$ is also verified to check the validity of the access list ac_A which was assigned to her. *Alice* is authorized if $cert_A$ is valid. Node C now constructs a secret key $L = h(x_{AC} \oplus T_A)$, and decrypts $(r)L$ to get r. It then generates a secret key $M = h(x_{CS} \oplus T_C)$ (where T_C is the timestamp created by C), encrypts *r*, and builds a MAC value (i.e. $MAC_1 = MAC(x_{CS}, (r)M \,||\, ID_A)$). Finally, C sends $(r)M, T_C, ID_A, MAC_1$ to S.

- **Step 3**: $S \rightarrow C: ID_S, MAC_2$

When S receives the message, it checks if $T_C > T_{now}$. Then, it verifies MAC_1 value. If valid, it indicates that *Alice* is authentic to S. After that, S constructs the secret key $M = h(x_{CS} \oplus T_C)$ and decrypts $(r)M$ to get r. Using this secret key, S builds a *MAC* ($MAC_2 = MAC(r, ID_S)$ and sends to *Alice*. S sends ID_S, MAC_2 to C.

- **Step 4**: $C \rightarrow A: ID_C, ID_S, S_C$

Node C verifies MAC_2. If valid, it generates a signature $S_C = sign_C(ID_S || ID_C)$ and sends ID_C, ID_S, S_C to *Alice*.

Upon receiving the ID_C, ID_S, S_C from C, *Alice* verifies C's signature S_C. If valid, then S and C is authentic to *Alice*.

5. PROTOCOL ANALYSIS

5.1 Security Analysis

Note that security level of the proposed protocol depends on the security level of ECC signature, message authentication code (CBC-MAC), and encryption algorithm (RC5). Those have been proven secure in literature. So in the scope of this paper, we focus on possible vulnerabilities to the proposed protocol.

5.1.1 It provides mutual authentication

In *step 2* of the protocol, node C verifies the signature S_A. If S_A is valid, then the user is authentic to C because only *Alice* can generate the signature S_A by his private key. Consequently, the user is also authentic to sensor *S* because *S* trusts C (*step 3*). On the other hand, only S shares the secret key x_{CS} with C. It means that only S can decrypt (*r*)M (where $M = h(x_{CS} \oplus T_C)$). So if S can achieve r from $(r)M$ to build $MAC_2 = MAC(r, ID_S)$), then S is authentic to the user. The mutual authentication is provided through trust relations between $Alice - C$, and $S - C$.

5.1.2 It can defend against replay attacks

There are two possible ways for an adversary to launch replay attacks as follows:

- The adversary can intercept the message sent out from *Alice* (in *step 1*) or from the sensor *S* (*step 3*). However, both cases are not possible in MAACE because C can easily detect by verifying timestamp T_A (*step 3*). If T_A is older than a predefined threshold, it is invalid because it has been used for previous authentication. If T_A was changed, then S_A ($S_A = sign_A((r)L \,||\, cert_A)$, where $L = h(x_A \oplus T_A)$) is not valid.

- The adversary can intercept the message sent out from C (*step 2*). Node S can detect by checking timestamp T_C. If T_C is older than the predefined threshold, it is not valid. If T_C has been changed to T_C^*, then the MAC_1^* value ($MAC_1^* = MAC(x_{CS}, (r)M \,||ID_A)$, where $M = h(x_{CS} \oplus T_C^*)$ is not consistent to MAC_1 .

5.1.3 It can mitigate DoS attack

Upon receiving the message from C (*step 2*), sensor node S first checks the validity of timestamp T_C. If it is not valid, then S discards the message. Otherwise, it computes a MAC value to compare with MAC_1 received. MAC, e.g. CBC-MAC, is a very fast message authentication code algorithm [24]. A CBC-MAC operation on Mica2 mote takes 3.12 ms [24], which is very fast compared to ECC point multiplications used by HBQ (which in total takes 3.5 s, about 1121 times longer). Therefore, the proposed scheme significantly reduces *DoS* compared to HBQ.

5.2 Performance Analysis

This section presents performance analysis of the proposed scheme and compare with existing ECC-based approaches such as HBQ [20] and ENABLE [18]. Since *Alice* and coordination node *C* are powerful devices, the computational overhead is trivial compared to that of the sensors. Therefore, we only consider computational overhead for sensors. We use the computational overhead (the computation time required by sensors, denoted by *T*) to analyze the performance. According to practical implementations on Mica2 motes [12][24][27], the computational time of each security primitives is mentioned in Table 2.

Table 2 Execution times of security primitives on Mica2

Notation	Description	Time (ms)
T_H	Time to perform one-way hash function (e.g. SHA-1)	3.636
T_{MAC}	Time to generate MAC value (e.g. CBC-MAC)	3.12
T_{RC5}	Time to encrypt/decrypt by RC5	0.26
T_{MUL}	Time to perform ECC point multiplication	810

The total computational time of the proposed scheme, ENABLE, and HBQ are shown in Table 2. In MAACE, both user authentication and node authentication take $2T_{MAC}+T_H+T_{RC5}$. For user authentication, ENABLE requires $1T_{MAC}$ (approximately 3.12 ms), while HBQ scheme requires $2T_H$, $2T_{MAC}$, $2T_{RC5}$, and $3T_{MUL}$ (total cost is approximately 2,451.04 ms).. For node authentication, ENABLE requires $2T_{MAC}+1T_{RC5}+1T_H$, while HBQ scheme does not support it. Based on Table 2, MAACE takes only 10.136 ms which is less than ENABLE (13.256 ms) and HBQ (2,451.04 ms). We used the formula $E = U*I*t$ to estimate the energy consumption of security computations [19][24]. For Mica2 mote, when processor is in active mode, $I = 8$ *mA*. Typically, $U = 3.0$ *V* if two new AA batteries are used [24]. Total computation time and energy consumption are shown in Figure 6. Our approach consumes 0.24 *mJ*, which is more efficient than ENABLE (0.381 *mJ*) and HBQ (58.82 *mJ*).

Table 3 Comparison of computational time.

	MAACE	ENABLE	HBQ
User Authentication	$2T_{MAC}+T_H+T_{RC5}$	T_{MAC}	$2T_H+2T_{MAC}+T_{RC5}+3T_{MUL}$
Node Authentication		$2T_{MAC}+1T_{RC5}+1T_H$	None
Total	$2T_{MAC}+T_H+T_{RC5}$	$2T_{MAC}+1T_{RC5}+1T_H$	$2T_H+2T_{MAC}+2T_{RC5}+3T_{MUL}$
Total Time	10.136 *ms*	13.256 *ms*	2,415.04 *ms*

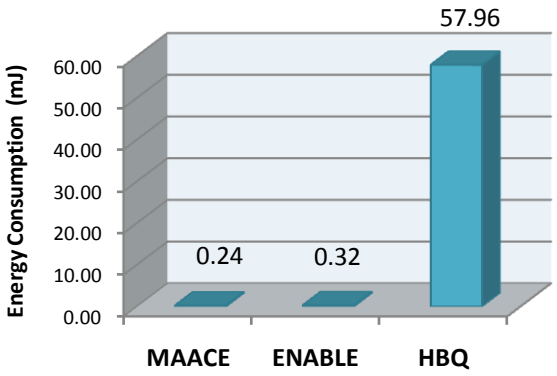


Figure 6 Comparison of energy consumption

6. CONCLUSION AND FUTURE WORK

One of the most critical security concerns before deploying a WSN in healthcare applications is patient privacy because their vital signs and activities are monitored all the time. To achieve this, authentication and access control must be enforced to ensure that only authenticated healthcare professionals can access, and further can access data that they have privilege for their healthcare services. This paper introduces a public key cryptography called Mutual Authentication and Access Control based on Elliptic Curve Cryptography (MAACE). MAACE provides mutual authentication (a healthcare professional can authenticate to an accessed node (a PDA or medical sensor) and vice versa) and ensures a healthcare professional can only access data that he/she has privilege. By applying elliptic curve cryptography, MAACE provides a public key approach which is more scalable and requires lesser memory compared to symmetric key-based schemes. Its performance makes it practically feasible to be implemented on sensor platforms. Security analysis and performance evaluation results have shown that MAACE is 238 times and 1.3 times faster than HBQ and ENABLE, respectively. Also, MAACE consumes 0.41 % and 75% energy compared to HBQ and ENABLE, respectively.

One of the main issues in ECC is that ECC multiplication operation takes significant time (and as a consequence, increases energy consumption). Reducing the operation cost will be our next goal to provide a more secure and energy-efficient scheme for WSNs.

7. REFERENCES

[1] H. S. Ng, M. L. Sim, C. M. Tan. Security issues of wireless sensor networks in healthcare applications. BT Tech. Journal, Vol. 24 No 2, 2006, pp. 138 – 144,

[2] K. Lorincz, D. Malan, T. F. Jones, A. Nawoj, A. Clavel, V. Shnayder, G. Mainland, S. Moulton, M. Welsh, *Sensor Networks for Emergency Response: Challenges and Opportunities*, , IEEE Pervasive Computing, Oct/Dec, 2004, pp. 16-23.

[3] http://fiji.eecs.harvard.edu/CodeBlue

[4] R. Jafari, R.Bajcsy, S. Glaser, B. Gnade, M. Sgroi, S. Sastry. *Platform Design for Health-care Monitoring Applications*, Joint Workshop on High Confidence Medical Devices, Software, and Systems (HCMDSS) and Medical

Device Plug-and-Play (MD PnP) Interoperability, June 2007, Boston, MA.

[5] http://bsn.citris.berkeley.edu/home/

[6] http://web.mit.edu/wockets/

[7] http://smart.csail.mit.edu/

[8] http://www.mobihealth.org/

[9] http://www.doc.ic.ac.uk/vip/ubimon/home/index.html

[10] Trossen, D.; Pavel, D.; *Sensor Networks, Wearable Computing, and Healthcare Applications*. IEEE Pervasive Computing. Vol. 6(2), April-June 2007, pp.58 – 61.

[11] Boukerche, A.; Yonglin R.. *A secure mobile healthcare system using trust-based multicast scheme*. IEEE J. Selected Areas Comm., vol 27(4), 2009 pp:387 – 399.

[12] R. Chakravorty, *A Programmable Service Architecture for Mobile Medical Care*. 4th IEEE International Conference on Pervasive Computing and Communications, 2006.

[13] J. Kim, A. R. Beresford, and F. Stajano, *Towards a Security Policy for Ubiquitous Healthcare Systems*, Proc. 1st International Conference on Ubiquitous Convergence Technology, pp. 263–272, 2006.

[14] S.-D. Bao, Y.-T. Zhang, and L.-F. Shen, *Physiological Signal Based Entity Authentication for Body Area Sensor Networks and Mobile Healthcare Systems*, Proc. 27th Annual International Conference of Engineering in Medicine and Biology Society, pp. 2455–2458, 2005.

[15] C.-W. Jeong, D.-H. Kim, and S.-C. Joo. *Mobile Collaboration Framework for u-Healthcare Agent Services and Its Application Using PDAs*, Proc. 1st KES International Symposium on Agent and Multi-Agent Systems: Technologies and Applications, pp. 747–756.

[16] M. Markovic, Z. Savic, and B. Kovacevic, Secure mobile health systems: principles and solutions, M-Health: Emerging Mobile Health Systems, Kluwer Academic Publishers, pp. 81–106, 2007.

[17] R. Marti, J. Delgado, and X. Perramon, Network and Application Security in Mobile e-Health Applications, Proc. International Conference on Networking Technologies for Broadband and Mobile Networks, pp.995–1004, 2004

[18] X.H. Le, S. Lee, I. Butun, M. Khalid, R. Sankar, M. Kim, M-H. Han, Y-K. Lee, H. Lee. *An Energy-Efficient Access Control Scheme for Wireless Sensor Networks based on Elliptic Curve Cryptography*. Journal of Communications and Networks, Special Issues on Secure Wireless Networking, December 2009 (in press).

[19] Huang, Y.M.; Hsieh, M.Y.; Chao, H.C.; Hung, S.H.; Park, J.H. *Pervasive, secure access to a hierarchical sensor-based healthcare monitoring architecture in wireless heterogeneous networks*. IEEE Journal on Selected Areas in Communications, Vol. 27, No 4, May 2009.

[20] Wang, H., Sheng, B., Li, Q.: *Elliptic curve cryptography-based access control in sensor networks*, Int. J. Security and Networks, Vol. 1, Nos. 3/4, pp.127–137 (2006).

[21] Du, W.; Deng, J.; Han, Y.; Varshney, P. *A Key Predistribution Scheme for Sensor Networks Using Deployment Knowledge*. IEEE Trans. Depend. Secure 2006, 3, 62–77.

[22] X-H. Le, S. Lee, Y-K. Lee, H. Lee. A Secure Coordination - based Data Dissemination for Mobile sinks in Sensor Networks. IEICE Transaction on Communication, 2009, Vol E92-B(01).

[23] X-H. Le, N. Canh, S. Lee, Y-K. Lee, H. Lee. An Energy-Efficient Secure Routing and Key Management Scheme for Mobile Sinks in Wireless Sensor Networks using Deploying Knowledge. Journal of Sensor, Special Issue "Wireless Sensor Technologies and Applications", 2008, Vol.8(12) pp. 7753-7782.

[24] Karlof, C. ; Sastry, N.; Dagner, D. *TinySec: A Link Layer Security Architecture for Wireless Sensor Networks*. In Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems (SenSys04), Baltimore, Maryland, November 2004; pp. 162-175.

[25] Zhu, S.; Setia, S.; Jajodia, S. LEAP+: Efficient Security Mechanisms for Large-Scale Distributed Sensor Networks. ACM Trans. Sens. Netw. 2006, 2, 500–528.

[26] Alex Biryukov, Christophe De Cannière, Gustaf Dellkrantz: Cryptanalysis of SAFER++. CRYPTO 2003: 195-211.

[27] Gura,N., Patel, A., Wander, A., Eberle, H., Shantz,S.C.: Comparing Elliptic Curve Cryptography and RSA on 8-it CPUs. In CHES 2004, Vol. 3156, LNCS , pp.119-132.

[28] Joppe W. Bos, Marcelo K., Thorsten Kleinjung, Arjen K. Lenstra and Peter L. Montgomery: On the Security of 1024-bit RSA and 160-bit Elliptic Curve Cryptography Cryptology. ePrint Archive: Report 2009/389.

[29] V. Miller, Use of elliptic curves in cryptography, CRYPTO 85, 1985.

[30] N. Koblitz, Elliptic curve cryptosystems, in Mathematics of Computation 48, 1987, pp. 203–209.

[31] ANSI X9.63, Elliptic Curve Key Agreement and Key Transport Protocols, American Bankers Association, 1999.

[32] Rivest, R. L. The RC5 Encryption Algorithm. Proceedings of the Second International Workshop on Fast Software Encryption (FSE) 1994e. pp. 86–96.