PHYSIOLOGICAL VALUE BASED PRIVACY PRESERVATION OF PATIENT'S DATA USING ELLIPTIC CURVE CRYPTOGRAPHY

Basant Tiwari[1] and Dr. Abhay Kumar[2]

[1]School of Electronics, DAVV, Indore, India
basanttiw@gmail.com
[2]School of Electronics, DAVV, Indore, India
dr.abhaykumar@rediffmail.com

ABSTRACT

Body Sensor Network (BSN) is a suitable combination of wearable tiny devices attached to patient's body. Their purpose is to monitor patient's physiological data (or BSN data) values. Sensors continuously monitor and collect patient's data and send it to a remote server through a network. This server can be called Database Server (DBS). DBS collect and stores the received patient's medical data which can be later used for any medical emergency by the Healthcare provider. Further, patient's data may be used to educate medical students, to provide data for medical research and analysis. Since the patient's physiological data are highly sensitive and BSN is very susceptible to attacks, therefore, it must be ensured that patient identity should not be exposed and altered as well as patient's data should not fall into hands of unauthorized users. Hence, maintaining privacy of patient's data over the network is an important aspect. So communication between BSN and DBS has to be secure. A strong security mechanism should be applied to maintain patient's privacy and confidentiality.

This paper proposes information security of physiological data which flow through network which is highly susceptible to attack and unauthorized access. Paper proposing physiological value based Encryption and Mutual Authentication (PVEMA) mechanism to enable mutual authentication and data encryption for a patient's physiological data. The work used practical approach of Elliptic Curve Cryptography (ECC), Message Authentication Code and Symmetric Encryption Scheme for maintaining the confidentiality, authenticity and integrity of patient's data through previous stored physiological value.

KEYWORDS

Body Sensor Network, Physiological Values, ECDSA, Mutual Authentication, Encryption, MAC Protocols,

1. INTRODUCTION

Recent research and advancement in wireless communication and physiological sensing have resulted in manufacture of tiny wearable devices, which are lightweight, run on low battery power. These devices can be integrated into the BSN for health monitoring in healthcare system. These tiny wearable devices are installed on human body for continuously sensing and collecting patient's data. This data has to be sent to database server (DBS) through PDA, also attached with patient's body (see figure 1). For such a system data security is a very important factor as well as the right information at the right time is the most important need for getting best possible care to the patient.

Patient's information has to be free from unauthorized access, so that patient's privacy is maintained. This information generation and flow is obtained by means of Body Sensor

Networks (BSN). This information includes physiological values and environmental parameters which flows between BSN, storage site called Database Server (DBS) and health care provider.

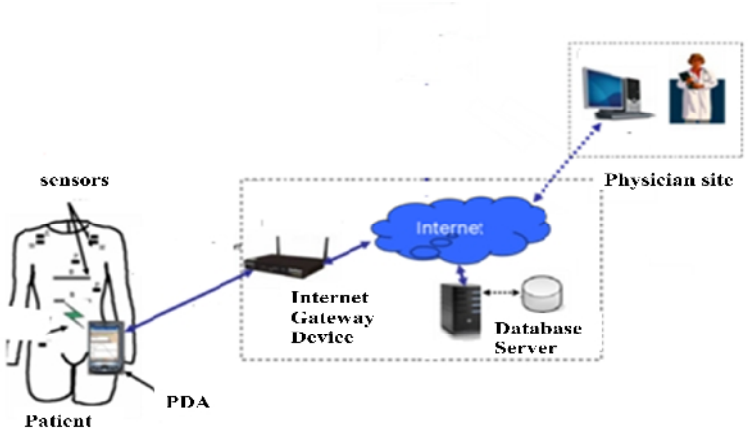

Figure 1: Architecture for a Body Sensor Network with major components

This communication of data between PDA and DBS must be continuous for storage of medical data for future use at time of medical emergency to any patient as well as can be used for any medical research and analysis. This communication over the Internet is strictly private and confidential [1] and should be encrypted to protect the patient's privacy. Any discloser in security may not only lead to loss of patient privacy, but may also cause physical harm to the patient by allowing adversaries to introduce false data or modification or suppression of correct data. This can also result in wrong diagnosis and actuation and treatment.

For that matter, inspiration comes from the HIPAA (Health Insurance Portability and Accountability Act) and PSQIA (The Patient Safety and Quality Improvement Act of 2005) of the US legislations could serve as a beaconing light for efficient regulation to protect the interest of patients. In Indian perspective, there is no such act but there are guidelines given by The Medical Council of India, Code of Ethics Regulations that set the professional standards for medical practice directs under chapter 2, section 2.2 as "Confidences concerning individual or domestic life entrusted by patients to a physician and defects in the disposition or character of patients observed during medical attendance should never` be revealed unless their revelation is required by the laws of the State" [2]. Also, Information Technology Act, 2000 (India) u/s 3 dealing with the 'authentication of electronic records' would provide the legal sanction and thus improvise security of the data. Also various other sections of this act *inter alia* other amendments and notifications (viz. Section 24, 25 of Part- II section 3(i) , Gazette of India, Oct 27th,2009) for data protection.[3]

Key management is a factor for secure transaction. For this intention the asymmetric keys (public/private key pair) or symmetric key whichever used must be secure. So whenever such key is transmitted between BSN and DBS or whenever new keys are generated the communication must be encrypted.

A potential way for key generation and management is the use of biometrics. It is a technique used for automatic identification and verification of person by his/his physiological and/or behavioural characteristics [4]. An algorithm proposed in [5] ensures authenticity, confidentiality and integrity of data using biometric. In [6, 7] heartbeat is used to generate a key, while [8] used physiological value Inter-Pulse-Interval (IPI) for generation of keys.

Paper describing algorithm and protocol for (1) offline global parameter and certification generation through Certificate Authority; (2) online two way mutual authentication between PDA and the Database Server (DBS); and (3) online encryption of BSN data for message

confidentiality between PDA and DBS. In spite of these, paper proposes, dynamic generation of keys on the basis of physiological values retrieved from DBS storage. It also eliminates the need for any additional explicit key distribution after the network is setup (e.g. [9] and [10]).

Further paper uses ECC scheme for securing data transmission between BSN and DBS. ECC provide lightweight and stringent security mechanism for resource constraint devices like mobile phone and PDAs [12, 13].

2. RELATED WORK

Physiological values were used first time in [14, 15] for securing inter-sensor communication. Basic idea behind the use of physiological values is to hide the key shared between 2 sides and also to correct any difference in physiological value by some simple error correction method. The choice of physiological value to be used for secure communication as well as the encryption scheme to be used was left on the choice of 2 communication sides. Author in [16] extended the work of [14] and used inter-pulse-interval (IPI) to generate cryptographic keys. IPI was measured from KEG and PPG by measuring time difference of 2 successive peaks in EKG/PPG signals. Important difference between this scheme and our work is that we are using physiological values for mutual authentication as well as for key generation in encryption process.

There is lot of work done on BSN authentication and key agreement schemes. Pre-loaded symmetric shared keys are used in large scale sensor networks for geographical region observations [17,18]. In these techniques, a certain key is loaded in each node and used to derive a shared secret key. In [19] a secure-limited channel (e.g., infrared) was used to exchange public-keys between parties before to the authentication process. However, for such approach we need high memory and computational power which may not be always available in small device of BSN. In [20] self-certified keys (SCK) and Elliptic Curve Cryptography (ECC) was used to establish asymmetric keys for authentication. Here KDC was used for key generation. Protocol called SNAP [21] also makes uses of ECC to set up pair-wise keys between nodes and the gateway. For which every sensor has biometric device which can authenticate the patient and shared secret is used for communicating with the base station. But, it does not set up any group keys. Lot of work has been done about ECC-based public-key cryptography [22]. It is best suitable for resource constrained devices.

These techniques are not always sufficient since there may be overlapping between communication path and time of two pairs of communication sides. That is why regeneration of keys in case of node addition, node revocation and the change of key when previous key was used for long time, is not included in the above proposed work. Also the pair wise key generation and exchange cannot always provide security against node capturing attack. Also there is an issue of limited storage. So, proposed work provides dynamic re-keying from previously stored PVs between PDA and DBS.

3. BACKGROUND

3.1 Elliptic curve cryptography

Elliptic curves are geometric curves with algebraic equation. They have been deeply studied for about last 150 years resulting in detailed knowledge. Cryptosystems some time need the use of algebraic groups. For these groups we can use elliptic curve. Group consists of elements on which custom-made operation can be done. For elliptic curve groups, these custom-made operations can be done geometrically as well as algebraically. The number of points on the curve is made limited to create a field for elliptic curve group. Elliptic curves are first studied over real numbers to study their geometrical properties. After which, elliptic curves are studied for the prime field F_p and binary field F_2^m.

An elliptic curve over real numbers may be defined as the set of points (x,y) that satisfy the ECC equation: $y^2 = x^3 + ax + b$, where x, y, a and b are real numbers. Every pair of values of a & b generates a different elliptic curve. For example, in the curve shown below in figure 2, we have a = -4 and b = 0.67. The equation of curve now becomes $y^2 = x^3 - 4x + 0.67$. If the ECC equation has no repeated factors for which, $4a^3 + 27b^2$ is not equal to 0, then we can use the elliptic curve $y^2 = x^3 + ax + b$ to form a group. An elliptic curve group over real numbers has points on the same elliptic curve and a point *O* at infinity.

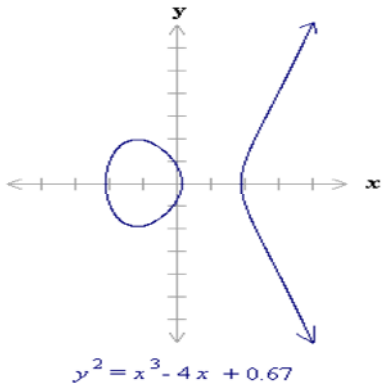

Figure 2: Elliptic curve graph for the equation $y^2 = x^3 - 4x + 0.67$

In this paper, we use prime field GF(p) over elliptic curve. The addition operation on an elliptic curve only involves a few arithmetic operations in GF(p), and hence is efficient. Taking an elliptic curve E on GF(p) with P >3 as an example, that addition follows the rules specified below:

(1) O + O = O.

(2) P + O = P for all P = (x, y) ▫ E. Namely, E has O as its identity element.

(3) P + Q = O for all P = (x, y) ▫ E and Q = (x, −y)
Namely, the inverse of (x, y) is simply (x, −y).

(4) Adding two distinct points for all P = (x1, y1) ▫ E and
Q = (x2, y2) ▫ E with x1 = x2, P + Q = (x3, y3) is defined by

x3 = λ2 − x1 − x2,

y3 = λ(x1 − x3) − y1,

where λ = (y2 − y1)/(x2 − x1).

(5) Doubling a point—for any P = (x, y) ▫ E with y = 0, 2P = (x*, y*) is defined by

x* = λ2 − 2x,

y* = λ(x – x*) − y,

where λ = (3x2 + a)/(2y).

3.2 Elliptic curve discrete logarithms

The discrete log problem involves finding logarithm of a number inside a finite field arithmetic system. Prime fields are fields in which sets are primes that is numbers of member in set is a prime numbers. Prime field have special importance in asymmetric cryptography. Since in prime field exponentiation is easy operation but inverse i.e. find the log is difficult. To generate a key pair in the discrete logarithm (DL) system, we have to calculate:

$$y=(gx) \bmod p$$

Here p is field size which is a large prime number, x and g are smaller than p. y is the public key. x is the private key. In discrete logarithm problem, we need to find x when we are given y, g and p. For example if g is multiplied by itself x times and result is reduced to fit in to the field with use of modulo operation to keep the result less than p. If y, g and p are known then it is very difficult to find the value of x.

3.3 Elliptic Curve Digital Signature Algorithm (ECDSA)

ECDSA is used to create a digital signature of data or file in order to verify its authenticity without compromising its security. ECDSA is used with a SHA1 cryptographic hash of the message to sign the file. A hash is a mathematical equation used on every byte of data which gives a number unique to the data. For example, the sum of values of all bytes can be used as hash function. So if anything changes in the message or in the file then the hash will also change. ECDSA signs the hash so if the data changes, the hash changes, and the signature isn't valid now.

For ECDSA, firstly need to know our elliptic curve parameters that are (P, a, b, G, n). We know that a and b are the elements of the curve equation $y^2 = x^3 + ax + b$, P is the prime modulus, that n is the number of points of the curve, and G is a 'Base point' or a point of origin. These curve parameters are important and without knowing them, we obviously can't sign or verify a signature. Verifying a signature isn't just about knowing the public key, we also need to know the curve parameters for which this public key is derived from.

So firstly, there must be a private key and a public key. The private key is a random number that is chosen from 1 to n-1, and the public key is multiplication of G with the private key that becomes a point on the curve. We set d_a as the private key and Q_a as the public key, then $Q_a = d_a * G$ (where G base point from the curve parameters).

ECDSA generates a pair (R, S) together which is the ECDSA signature. We use these two values used for signature. In this process firstly we generate a random value 'k', and use point multiplication to find the point $P_{(x,y)}=k*G$. This point's x coordinate is called 'R'. Since the point on the curve P is represented by its (x, y) coordinates, we only need the 'x'value for the signature, and that value will become 'R'. Now all we have to calculate 'S' value.

For which, we make a SHA1 hash of the message, that will be huge integer number and we call it 'z'. Now we can calculate S using the equation:

$$S = k^{-1} (z + d_a * R) \bmod p$$

Note here the k^{-1} which is the 'modular multiplicative inverse of k. Which is inverse of k, but it is not possible since we are dealing with integers, so it's a number for which $(k^{-1} * k) \bmod p$ is equal to 1. Now we have signature (R,S), which we want to verify. For which we need the public key and curve parameters. We use following equation to calculate a point P:

$$P_{(x,y)} = S^{-1}*z*G + S^{-1} * R * Q_a$$

If the Px (x coordinate of the point P) is found equal to R, then signature is valid, otherwise not valid. We are using ECDSA in mutual authentication process described in section 4.2.

3.4 Elliptic Curve Integrated Encryption Scheme (ECIES)

Elliptic curve Integrated Encryption Scheme (ECIES) is based on ECC, is public-key cryptographic mechanisms. It provide capabilities for encryption, digital signature and key exchange. ECIES integrated in the ANSI X9.63, ISO/IEC 18033-2, IEEE 1363a, and SECG SEC 1 standards. In the present work, we use ECIES for detailing the encryption and decryption procedures and the list of functions and special characteristics included in aforementioned standards.

ECIES, is an enhancement of ElGamal encryption specifically designed for EC groups. As it name indicates, ECIES is an integrated encryption scheme that uses the following functions:

Key Agreement (KA): It is a function used by two parties for the creation of a shared secret.

Key Derivation (KDF): It is a mechanism that produces a set of keys from domain parameters of EC and some optional parameters.

Hash (HASH): is a Digest function .

Encryption (ENC): Symmetric encryption algorithm.

Message Authentication Code (MAC): Information used to authenticate a message.

3.5 Key Derivation Functions

The key derivation functions are used by the Elliptic Curve Integrated Encryption Scheme and the key agreement schemes. The key derivation functions will be used to derive keying data from a shared secret octet string. The used key derivation function in this paper is ANSI-X9.63 KDF.ANSI-X9.63-KDF is the simple hash function construct described in ANSI X9.63 [23].

3.6 MAC schemes

The MAC schemes will be used by the Elliptic Curve Integrated Encryption Scheme specified used in this paper. MAC schemes are designed to be used by two entities — a sender PDA and a recipient DBS when sender PDA want to send a message M to DBS in authentic manner and PDA want to verify the authenticity of M.

Here the MAC schemes are described in terms of a tagging operation, a tag checking operation, and associated setup and key deployment procedures. PDA and DBS should use the schemes as follows when they want to communicate.

First PDA and DBS use a symmetric key KM to control the tagging and tag checking operations. Then each time PDA wants to send a message M to DBS, PDA apply the tagging operation to M under the symmetric key KM to compute the tag D on M, and convey M and D to DBS. Finally when DBS receives M and D, DBS apply the tag checking operation to M and D under the same symmetric key KM to verify the authenticity of M. If the tag checking operation outputs ‘valid’, DBS concludes that M is indeed authentic.

MAC schemes are designed so that it is hard for an adversary to forge valid message and tag pairs so that the schemes provide data origin authentication and data integrity. The list of supported MAC schemes is:

HMAC(SHA-160)with 160 bit key,

HMAC(SHA-1–80) with 160 bit keys,

HMAC-MD5 With 128 bit key.

Both these MAC schemes are specified in IETF RFC 2104 [24] and ANSI X9.71 [25] based on the hash function SHA-1 specified in FIPS 180-1 [26].

3.7 Symmetric Encryption Scheme

The symmetric encryption schemes will be used by the Elliptic Curve Integrated Encryption Scheme. Symmetric encryption scheme is used by the sender PDA and recipient DBS, when PDA wants to send a message M to DBS confidentially and DBS wants to recover M.

Here symmetric encryption schemes are described in terms of an encryption operation, a decryption operation, and associated setup and key deployment procedures. PDA and DBS use the scheme as follows when they want to communicate. First PDA and DBS use symmetric key KS to control the encryption and decryption operations. Then each time PDA wants to send a

message M to DBS, PDA the encryption operation to M under the symmetric key KS to compute the encryption or cipher text C of M, and convey C to DBS. Finally when DBS receives C, DBS apply the decryption operation to C under the same symmetric key KS to recover the message M.

Symmetric encryption schemes are designed so that it is hard for an adversary to recover information about messages (other than their length) from their ciphertexts. Thus they provide data confidentiality. The list of supported symmetric encryption schemes is:

3-Key TDES in CBS Mode

X-OR Encryption scheme

3-key TDES in CBC mode is specified in ANSI X9.52 [27].

The XOR encryption scheme is the simple encryption scheme in which encryption consists of XORing the key and the message, and decryption consists of XORing the key and the cipher text to recover the message. The XOR scheme is commonly used either with truly random keys when it is known as the 'one-time pad', or with pseudorandom keys as a component in the construction of stream ciphers. TheXOR encryption scheme uses keys which are the same length as the message to be encrypted or the cipher text to be decrypted.

3-key TDES in CBC mode is designed to provide semantic security in the presence of adversaries launching chosen-plaintext and chosen-cipher text attacks. The XOR encryption scheme is designed to provide semantic security when used to encrypt a single message in the presence of adversaries capable of launching only passive attacks.

4. PROPOSED SOLUTION

This paper proposing algorithm to make the communication 2 ways between DBS and BSN without compromising the secure exchange of the new keys generated whenever necessary except the first time when BSN is installed and activated. At this time involvement of third party such as Certification Authority (CA) is necessary. For further key regeneration, exchange and Encryption scheme between BSN and DBS, algorithm uses one of the past physiological values stored at DBS. So in this paper we present a novel and efficient secure communication protocol called Physiological Value based Encryption and Mutual Authentication (PVEMA) for secured communication between BSN and DBS by dynamically generated keys.

This paper proposed the use of Physiological value as input for key generation, generated at patient's Body sensor and which is collected and transmitted by PDA at patient's body. At PDA algorithm uses Elliptic Curve Cryptography for further encryption. For ECC, we have to find values of domain parameters a, b, G and n then key generation process will start and then Encryption/Decryption will start. These steps are further described in detail.

4.1 System Setup for Encryption

The setup procedure for proposed solution is specified in Section 4.1.1, the key deployment procedure is specified in Section 4.1.2, the encryption operation is specified in Section 4.1.3, and the decryption operation specified in Section 4.1.4.

4.1.1 System Setup

DBS should perform the following setup procedure to prepare to use ECIES:

1. The operation of each of the public-key cryptographic schemes involves arithmetic operations on an elliptic curve over a finite field determined by some elliptic curve domain parameters. Elliptic curve domain parameters over Fp are:

$$T = (P, a, b, G, n)$$

consisting of an integer p specifying the finite field Fp, two element a, b specifying an elliptic curve E(Fp) defined by the equation:

$$E: \quad y^2 = x^3 + ax + b \pmod{p},$$

a base point G(Gx,Gy) on E(Fp), a prime n which is the order of G. Elliptic curve domain parameters over Fp generated.

2. DBS establish the key derivation functions. For *KDF* we use ANSI-X9.63-KDF with the option SHA-1.
3. DBS establish the MAC schemes. In MAC scheme chosen, k_M denote the key used by *MAC*, to produce tag. Proposed work includes HMAC-MD5 MAC function.
4. DBS establish symmetric encryption schemes. Let *ENC* denote the encryption scheme chosen, and k_s denote the keys used by *ENC* to produce cipher text. Proposed work includes X-OR Encryption Scheme.
5. PDA obtain in an authentic manner the selections made by DBS that are the elliptic curve domain parameters T, key derivation function *KDF*, the MAC scheme *MAC*, and the symmetric encryption scheme *ENC*.

4.1.2 Key Deployment

PDA and *DBS* perform the following key deployment procedure to prepare to use ECIES:

1. DBS establish an elliptic curve private and public key pair $d_{dbs,}$ and Q_{dbs} associated with the elliptic curve domain parameters *T* established during the setup procedure. The key pair generated as follows:

   Input: ECC Domain parameters T = (P, a, b, G, n) .

   Output: ECC key pair $d_{dbs,}$ and Q_{dbs} related with *T*.

   process: Generate an elliptic curve key pair as follows:

   1. Randomly or pseudo randomly select an integer d_{dbs} in the interval 1 to *n*-1.
   2. Calculate $Q_{dbs} = d_{dbs}.G$.
   3. Output $d_{dbs,}$, Q_{dbs} .

2. In the same way PDA generates in public and private key pair d_{pda} and Q_{pda}. PDA and DBS obtain in an authentic manner the elliptic curve public key of each other.

4.1.3 Encryption Operation

PDA encrypts messages using ECIES using the keys and parameters established during the setup procedure and the key deployment procedure as follows:

Input: Inputs are:

1. An *Message M* which is to be encrypted.
2. Shared Physiological values S1 and S2.

Output: triplet (R, C, D)

Process: Encrypt *M* as follows:

1. Select a random number k ▫ [1,n- 1] and calculate R = kG
2. Compute P(x, y)=k.Q_{dbs} . If P ≠ 0. IF P= 0 then it is invalid and go to step 1.
3. Use a Key Derivation Function (KDF) to derive a (S + M)-bit key, k_{KDF} , from x and S.

$$k_{KDF} = KDF(x|| S1)$$

4. Parse the S left most bits of k_{KDF} as the S-bit key k_S.
5. Use the encryption operation of the symmetric encryption scheme *ENC* established during the setup procedure to encrypt *M*

$$C = ENC(k_S, M)$$

6. Parse the M right most bits of k_{KDF} as the M-bit key k_M.
7. Use the tagging operation of the MAC scheme *MAC* established during the setup procedure to compute the tag *D*

$$D = MAC(k_M, C||S2)$$

8. Send triplet (R, C, D) to DBS

4.1.4 Decryption Operation

DBS should decrypt ciphertext using ECIES using the keys and parameters established during the setup procedure and the key deployment procedure as follows:

Input: Inputs are:

1. A triplet (R, C, D).
2. Shared Physiological values S1 and S2

Output: Message M.

Process: Decrypt *C* as follows:

1. Compute (X', Y') = d_{dbs}.R
2. Use a Key Derivation Function (KDF) to derive a (S + M)-bit key, k_{KDF} , from x' and shared value S1.

$$k_{KDF} = KDF(x'||S1)$$

3. Parse the M right most bits of k_{KDF} as the M-bit key k_M
4. Use the tag checking operation of the MAC scheme *MAC* established during the setup procedure to compute the tag *D'*.

$$D' = MAC(k_M, C||S2)$$

5. Check whether D'=D. If yes then go to step 7 else invalid D and stop the procedure.
6. Use the encryption operation of the symmetric encryption scheme *ENC* established during the setup procedure to decrypt *C*

$$M = ENC(C, k_S).$$

7. Accept message M.

4.2 System Setup for Mutual Authentication

After a patient is installed with a new BSN, it has to establish a connection with DBS. After a link is setup, both PDA and DBS contact to Certification Authority (CA) for getting certificate. For certificate generation PDA and DBS both sends their public keys Q_{pda} and Q_{dbs} to CA through secure and authenticated channel to CA. The CA uses it private key to generate the signature. CA uses ECDSA algorithm to generate certificates and gives certificates (r_{pda},s_{pda}) to PDA and (r_{dbs},s_{dbs}) to DBS with its on public key Q_{ca} to both of them. The certificate consists of a pair of integers which is denoted as (r_{pda}, s_{pda}) for the PDA and (r_{dbs}, s_{dbs}) for the database server (DBS). Here rpda and rdbs are the x coordinates of the distinct elliptic curve points R_{pda} and R_{dbs} respectively. These certificates and public key of CA is used at the time of mutual authentication process when they send its own certificate to each other. They used *Qca* (public key of CA) in verification phase.

4.2.1 Mutual Authentication

When PDA and DBS are ready to communicate, first of all they authenticate to each other by sending signature to each other that are received by CA and they verify the signature of each other. This process is done as follows:

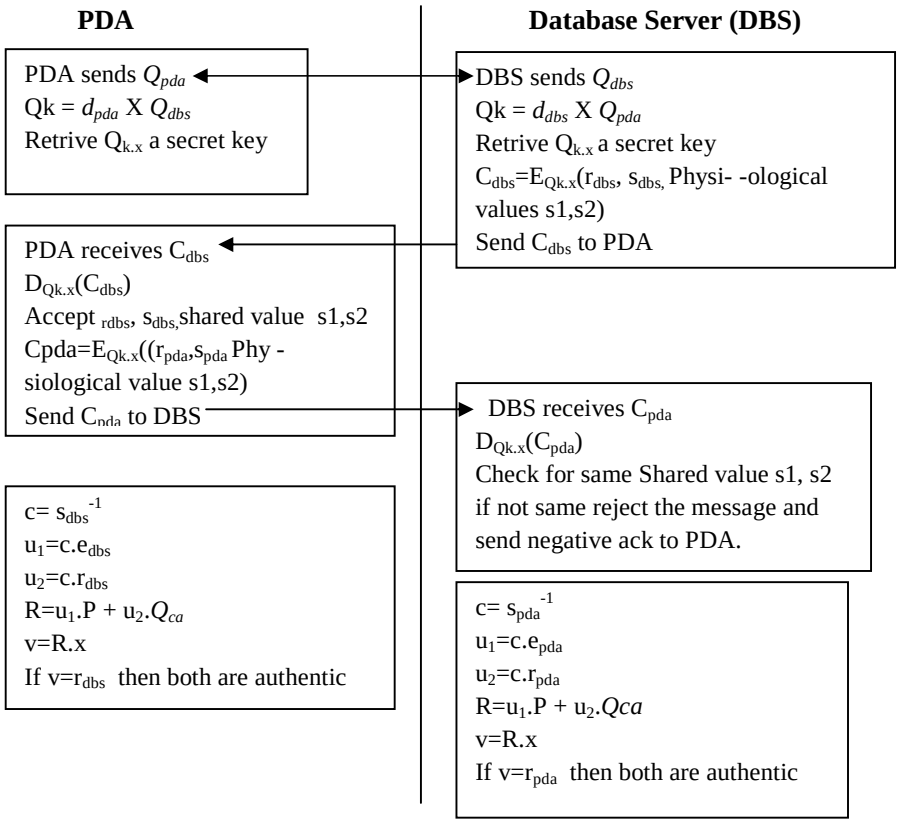

Figure 3: Mutual Authentication between PDA and DBS

In mutual authentication process, first of all PDA and DBS exchange its public keys Q_{pda} and Q_{dbs}. After that they generate the secret key by multiplying its own private key to other public key just received. Secret key is used to send the certificate in encrypted form. Now the DBS will send the certificate with 2 physiological values S1 and S2 to PDA in encrypted form that is Cdbs. Physiological values in initial mutual authentication is taken randomly from DBS on which both DBS and PDA are agreed. The PDA then decrypts Cdbs and obtains the certificate of DBS with physiological values. The PDA now encrypts its certificate and received physiological values and send Cpda to DBS. Now DBS will decrypt it and obtains the PDA's certificate with physiological values. Now DBS checks received physiological values with sent physiological values, if both are same then it will go for certificate verification otherwise it will reject the Cpda and send the negative acknowledgement to PDA and again restart the process of sending certificate. Now at this point, both PDA and DBS have the certificate of each other and they authenticate to each other by verifying the certificates. If they verify to each other and found authenticate, PDA and DBS are ready to send and receive the actual BSN data.

Physiological values are used in KDF function for key pair generation and MAC function for tag generation in Encryption and Decryption operation described in 4.1.3 and 4.1.4.

4.3 Initial Communication Process

4.4 Both DBS and PDA verify the signature and start communication with ECIES encryption and decryption operation described in subsection 4.1.3 and 4.1.4 respectively.

4.5 Mutual Authentication & Communication Protocol

When patient's successfully installed BSN and decide to communicate with Database Server through PDA, it will follow the following protocol:

4.5.1 DBS initialized the process by mutual authentication process described in 4.2.1.

4.5.2 PDA and DBS mutually authenticate to each other. After that PDA initiate the data communication. For that PDA Collects the respective Physiological data from sensor and make a message M that is sent to Encryption operation described in 4.1.3. PDA sends triplet (R,C,D) to DBS.

4.5.3 DBS decrypt the message using Decryption operation described in 4.1.4 and accept and store the BSN data. At the same time DBS send an acknowledgement of data stored to PDA.

4.5.4 Now PDA understand that mutual authentication has been done successfully and continue data transfer.

4.6 Mutual Re-Authentication or Regeneration of Keys

Mutual re-authentication or key Regeneration is initiated by DBS. After some time when DBS decide re-authenticate the PDA or to change the keys, it will proceed by selecting previously stored PVs. These PVs are used in KDF function for key pair generation and MAC function for tag generation in encryption and decryption operation. This decision of re-authentication or changing keys is taken preferably around mid night which will give normal values for extended period of time indicating normal health.

4.6.1 Physiological value selection

The PVEMA scheme utilizes a specific physiological values for selection new key pair generation for encryption and origin authentication.

The transmission continues with present parameters and keys till a fixed minimum or maximum period is over. Between this minimum and maximum interval, key regeneration is decided by Database Server (DBS). This decision is based on 2 factors: (1) Minimum prescribed period is over and Maximum prescribed period is not over. For example it was decided that one set of keys will be used for minimum of 23 hours and maximum 24 hours. Then new keys pair will be regenerated between 23 and 24 hours from the previous key generation. (2) Key regeneration is done at a time23 to 24 hours from previous generation when continuously normal values are received for some period of time, suppose for 10 minutes. Continuous normal values for these 10 minutes indicates normal health and no emergency and this also predicts that patient is in relaxed state since it is time of his/her sleep.

5. SECURITY ANALYSIS

Reply Attack: A replay attack is a form of network attack in which a valid data transmission is maliciously or fraudulently repeated or delayed. Here an attacker copies a stream of messages between two parties and replays the stream to other party as a authenticate user. To prevent replay attacks, we use random physiological value that is sent in encrypted form from DBS to PDA and PDA to DBS at the time of mutual authentication. Replay attacks will be defeated because the replayer cannot know in advance the physiological value the sensor will generate.

Man in the middle attack: An attacker observes a session opening on a network. Once authentication process is complete between two parties, he/she can attack the client computer to immobilize it, and use IP spoofing to claim to be the legitimate client just authenticated and begin operating the session. This attack is prevented here since the PDA does not send the message containing his identity directly but in encrypted form of the data. So the man in middle cannot decipher the text since he does not have private key of receiver nor the physiological values by which key pairs could be generated.

Security against Chosen Cipher text Attacks: Proposed scheme is secure against chosen cipher text attack. If the PDA wants to encrypt any message 'M' then he uses DBS public key Q_{dbs} and physiological value for the encryption. Now the pair (R,C,D) is chosen and sent to the

DBS. Even if attacker get chosen cipher text he still need receiver's private key physiological values to generate key pairs for diciphering.

Confidentiality: Since our scheme is based on ECIES in which symmetric encryption is incorporated so that it become difficult for adversary to recover any information from cipher text. Thus our scheme provides data confidentiality.

Unforgeability: For forging the message, the private key of DBS is required, which is kept secured with DBS. Thus the property of unforgeability is maintained with the secrecy of the shared secret key. As well as It is computationally infeasible to forge a valid cipher text C sent by PDA and claim that it is came from PDA unless private key of PDA is known.

Non-repudiation: Non-repudiation is the assurance that someone cannot deny something. That is PDA cannot deny that encrypted text is not sent by it. Any trust party or receiver himself can check that it is sent by the PDA by running the verification procedure.

Integrity: Ensuring that information is not altered by unauthorized persons. If the Cipher text has been altered by unauthorized user from C to C' and then in decryption process the tag value is calculated other than D'. This alteration can be found at the time of verification process (step no 5 of Decryption process) and the cipher text has been denied by DBS to accept and hence integrity is ensured.

6. CONCLUSION

This paper has presented the working of a protocol for mutual authentication and key re-generation using physiological values from previously stored patient's data stored at Database Server (DBS) for secure communication. In this work, use of Key Distribution Centre (KDC) is eliminated. Benefit of this new scheme is that two parties can generate keys between themselves by physiological values. Also mutual authentication is done between two communicating parties only without assistance from CA, except first time when system is setup. This work ensures data confidentiality, integrity and unforgeability. It also prevents various attacks viz. reply attack, Man in the Middle attack, chosen cipher attack. The whole work is based upon the Elliptic Curve Cryptography, which provides good enough security with small enough keys and calculations. In summary, the goal of this paper is to look into the method for providing secure communication between BSN and DBS with less overhead for communication with third party. This paper also ensured that guideline and act laid down for patients rights in Indian legislature are satisfied.

REFERENCES

[1] Bhargava, A. & Zoltowski, M. (2003). Sensors and wireless communication for medical care. In Proceedings of 14th international workshop on database and expert systems applications, Sep 2003, pp. 956–960.

[2] http://www.mciindia.org/RulesandRegulatins/CodeofMedicalEthicsRegulations2002.aspx

[3] http://cca.gov.in/cca/sites/default/files/files/gsr780.pdf.s

[4] Guennoun, M., Zandi, M., & El-Khatib, K. (2008). On the use of biometrics to secure wireless biosensor networks. In 3rd International conference on information and communication technologies: From theory to applications. ICTTA 2008, Damascus, Apr 2008, pp. 1–5.

[5] Poon, C. C. Y., Zhang, Y.-T., & Bao, S.-D. (2006). A novel biometrics method to secure wireless body area sensor networks for telemedicine and m-health. IEEE Communications Magazine, 44(4), 73–81.

[6] Bao, S.-D., Poon, C. C. Y., Zhang, Y.-T., & Shen, L.-F. (2008). Using the timing information of heartbeats as an entity identifier to secure body sensor network. IEEE Transactions on Information Technology in Biomedicine, 12(6), 772–779.

[7] Bui, F. M., & Hatzinakos, D. (2008). Biometric methods for secure communications in Body Sensor Networks: Resourceefficient key management and signal-level data scrambling. In EURASIP Journal on Advances in Signal Processing, Vol. 2008, article ID 529879, 16 p.

[8] Krishna K. Venkatasubramanian; Sandeep K. S. Gupta "Physiological value-based efficient usable security solutions for Body Sensor Networks ACM Transactions on Sensor Networks 2010; Volume 6, Number 4, July 2010"

[9] Z Hu,S.,Setia,S.,and Jajodia, S. 2006. Leap+: Efficient security mechanisms for large-scale distributed sensor networks. ACM Trans. Sens. Netw. 2, 4, 500–528.

[10] E Schenauer,L. and Gligor, V. D. 2002. A key-management scheme for distributed sensor networks. In Proceedings of the 9th Conference on Computer and Communications Security. ACM, 41–47.

[12] Wendy Chou, Dr. Lawrence Washington, Elliptic Curve Cryptography and Its Applications to Mobile Devices, Proc. IEEE INFOCOM '04, Mar.2004.

[13] Kathryn Garson, Carlisle Adams "Security and privacy system architecture for an e-hospital environment" IDtrust '08 Proceedings of the 7th symposium on Identity and trust on the Internet Pages 122-130

[14] S. Cherukuri, K. Venkatasubramanian, and S. K. S. Gupta. BioSec: A Biometric Based Approach for Securing Communication in Wireless Networks of Biosensors Implanted in the Human Body. pages 432439, Oct 2003. In Proc. of Wireless Security & Privacy Workshop 2003.

[15] C. C. Y. Poon, Y.-T. Zhang, and S.-D. Bao. A Novel Biometrics Method To Secure Wireless Body Area Sensor Networks for Telemedicine And M-Health. IEEE Communications Magazine, 44(4):7381, 2006.

[16] K. Venkatasubramanian and S. K. S. Gupta. Security for Pervasive Health Monitoring Sensor Applications. pages 197202, Dec 2006. In Proc. of the 4th Intl. Conf. on Intelligent Sensing & Information Processing.

[17] Liu D, Ning P, Li R. Establishing pair-wise keys in distributed sensor networks. ACM Trans. Inf. Sys.Secur. 2005;8:41–77.

[18] Eschenauer L, Gligor VD. A key-management scheme for distributed sensor networks. Proceedings of the 9th ACM Conf on Computer and communications security; Washington, DC, USA. 18–22 November 2002.

[19] Balfanz D, Smetters D, Stewart P, Wong H. Talking to Strangers: Authentication in Ad-hoc Wireless Networks. Proceedings of Network and Distributed System Security Symp; San Diego, CA, USA. 6–8 February 2002.

[20] Jiang C, Li B, Xu H. An efficient scheme for user authentication in wireless sensor networks. Proceedings of the 21st International Conference on Advanced Information Networking and Applications Workshops; Niagara Falls, Canada. 21–23 May 2007.

[21] Malasri K, Wang L. Addressing security in medical sensor networks. Proceedings of the 1st ACM International Workshop on Systems and Networking Support for Healthcare and Assisted Living Environments; San Juan, Puerto Rico. 17 June 2007.

[22] Wang H, Sheng B, Li Q. TelosB Implementation of Elliptic Curve Cryptography over Primary Field.Dept. of Computer Science, College of William and Mary; Williamsburg, VA, USA: Oct, 2005. Technical Report WM-CS-2005-12.

[23] American Bankers Association, Public Key Cryptography For The Financial Services Industry : Key Agreement and Key Transport Using Elliptic Curve Cryptography, ANSI X9.63-2001, November 20, 2001.

[24] H. Krawcyzk, M. Bellare, and R. Canetti, HMAC: Keyed Hashing for Message Authentication. Internet Engineering Task Force, Internet RFC 2104, 1997. Available from: http://www.ietf.org/

[25] ANSI X9.71-199x: Keyed Hash Message Authentication Code. March, 1998. Working Draft.

[26] FIPS 180-1. Secure Hash Standard, Federal Information Processing Standards Publication 180-1, 1995. Available from: http://csrc.nist.gov/

[27] ANSI X9.52-1998: Triple Data Encryption Algorithm Modes of Operation. American Bankers Association, 1998.

Author

Basant Tiwari, Ph.D. Research Scholar from DAVV, Indore. He did his M. Tech. (CSE) from Rajiv Gandhi Technical University, Bhopal. He is working on patient monitoring and remote medical care. He has 9 years of teaching experience and has published 10 papers in National and International conferences & attended many National & International Conferences/ Workshops/ Seminars/ Symposiums etc. He is a Member of IEEE, ACM, CSI and IACSIT. Recently he is honored as "Affinity Gold Chair" by IEEE M.P. Sub-Section.

Prof. Dr. Abhay Kumar, is a Ph. D. n Electronics Engineering. He has 20 years experience in Teaching. He has published around 35 papers in reputed journals and conferences. Three Sponsored Research were carried out by himself as Principal Investigator. Served as referee for IEEE publications and other journals, Served as Programme Committee Member to many conferences.