

BioSec: A Biometric Based Approach for Securing Communication in Wireless Networks of Biosensors Implanted in the Human Body

Sriram Cherukuri, Krishna K Venkatasubramanian and Sandeep K S Gupta
Department of Computer Science and Engineering
Arizona State University, Tempe, AZ.
sandeep.gupta@asu.edu

Abstract

Advances in microelectronics, material science and wireless technology have led to the development of sensors that can be used for accurate monitoring of inaccessible environments. Health monitoring, telemedicine, military and environmental monitoring are some of the applications where sensors can be used. The sensors implanted inside the human body to monitor parts of the body are called biosensors. These biosensors form a network and collectively monitor the health condition of their carrier or host. Health monitoring involves collection of data about vital body parameters from different parts of the body and making decisions based on it. This information is of personal nature and is required to be secured. Insecurity may also lead to dangerous consequences. Due to the extreme constraints of energy, memory and computation securing the communication among the biosensors is not a trivial problem. Key distribution is central to any security mechanism. In this paper we propose an approach wherein, biometrics derived from the body are used for securing the keying material. This method obviates the need for expensive computation and avoids unnecessary communication making our approach novel compared to existing approaches.

Index Terms—security, key management, Pervasive Computing, biometrics, randomness.

I. INTRODUCTION

Pervasive computing is an environment where people interact with various companion, embedded, and invisible computers. The objective of pervasive computing technologies is to enable a seamless integration of computing devices with the environment. This enables the environment to react to the user's computing needs without the user actually expending his time and energy. Use of embedded micro-sensors is an integral part of pervasive computing. Health care is a very important aspect of everyday life. Hence it is imperative that pervasive computing be extended to health care applications. Pervasive computing has the potential to provide low cost, high performance, and people centric solutions for health care, and monitoring. The rapid improvements in microprocessor and sensing material technology has lead to a development

of miniature sensors that can be implanted in the human body. The biosensor based approach to health care makes it much more effective by reducing the response time, and decreasing the granularity of the application. By this technology continuous health monitoring of a human body and real time data collection would be possible.

The biosensors are implanted in the human body. These sensor form a wireless network between themselves and some entities which are external to the human body. A wireless network is the most suitable option because a wired network would require laying wires within the human body, which is not desirable. Such a network can be used for a multitude of applications. These include both data aggregation and data dissemination applications. Biosensors may be used for monitoring the physiological parameters like blood pressure, glucose levels and collecting the data for further analysis. This enables real time health monitoring. Biosensors placed in the subcutaneous layer of skin, nasal area, tongue may be used to detect the presence of harmful toxins in say the food ingested and air inhaled. As soon as a toxin is detected corrective actions may be taken or at least the host may be informed about it. Such applications would prove to be extremely useful in situations like biological and chemical attacks where small response time is very crucial to avoid extensive damage. Data dissemination applications include biosensors for visually impaired, wherein biosensors are implanted in the retina of persons with visual disabilities. These biosensors collect the light signals from outside and stimulate optical cells of the eye, thereby enabling at least partial vision. Other applications include those which stimulate actions like drug administration inside the body based on certain inputs.

In any information system it is essential to build a security mechanism in order to protect the information, as it is susceptible to breaches either when it is stored or when it is being transmitted. The required degree of security depends upon the particular application. A trade-off always exists between the provided level of security and the performance of the system. The information to be transmitted in the present application is crucial medical information. It is required by law that this information must be secure [1], in the sense that it must possess the characteristics of secure data: authenticity, integrity, and confidentiality. Wireless

communication among biosensors requires authenticity, because the physician or patient (receiving the feedback) must be confident that the received signals did indeed originate from biosensors of that patient. This communication should be confidential, because this medical information must be inaccessible to outsiders. Integrity of the communication must be enforced, so that it is not possible for an adversary to modify the signals and go undetected, giving rise to false results such as erroneous images and false feedback. If any of the above conditions is not satisfied, serious harm to the health of an individual could occur, depending upon the location and the intended application. Therefore the biosensors have to make use of cryptographic algorithms to encrypt the data they send to the control node. Cryptographic algorithms can be classified into symmetric and asymmetric schemes. The biosensors, like any other miniature sensors has constraints of energy and bandwidth. In addition they are also subject to constraints arising out of their unique location of placement (i.e.), the human body. These constraints must be addressed while making security architecture choices for the biosensor network security. The selection of a symmetric or asymmetric encryption system, key generation method, the key distribution protocol are some of the significant factors. Asymmetric key cryptography needs more resources compared to symmetric key cryptography in terms of both communication and computation. In symmetric key cryptography a key is obtained by the entities in a secure manner. Once the key is exchanged the entities can use this for further communication.

In this research we examine the constraints and issues of the biosensor security. The set constraints experienced by the biosensors make existing solutions to sensor network security unsuitable for biosensor security. Hence biosensors security requires novel solutions to these problems. In our application, biosensor are placed inside the human body. Hence it is an attractive proposition to derive the required inputs for security mechanism from the body. This input is in the form of biometrics from the body. In this research we examined the utility of various biometric for security purpose. The main criterion for a biometric to be suitable for security purpose is whether it is random enough for it to be used to build security system. In this paper we propose a security mechanism which makes use of biometric derived from the human body to secure the keying material which in turn is used to secure the data communication.

The remainder of the paper is organized as follows. In Section II we present the related work. In Section III we discuss the constraints of the biosensor networks and discuss as to why biosensor security is not a trivial problem. In Section IV we present our system model. Then we describe the problem of securing the biosensor communication in detail. This is followed by a brief description of the role of random numbers in security. Then our solution is presented in Section VII. Finally we present the conclusions

and future work in Section VIII.

II. RELATED WORK

Relatively very little work has been done in the area of security for sensor networks. Whatever little work that has been done is for generic sensors and have not considered operation in environments with extremely stringent constraints as in case of biosensors.

Perrig et al. [3] have presented a set of protocols for achieving requirements of security like confidentiality and authenticity. Their architecture consists of two building blocks namely SNEP and μ Tesla. In SNEP they use symmetric keys to encrypt the data. Symmetric keys are also used to compute the Message Authentication Code(MAC). Both these set of keys are derived from a master key which is shared by the nodes with the base station and are placed in them before being deployed. μ Tesla is used to achieve authenticated broadcast by delayed key disclosure. The keys are computed from the master pre-deployed key and the counter which is incremented after each block. Effectively this system uses pre-deployed keying. The communicating entities are synchronized with each other by means of the counter, which is incremented after each block that is communicated.

Joshi et al. [7] present a scheme that is similar to the one that is presented above. In addition they address issue of multi-hop communication among the nodes. They ensure end-to-end security by encrypting different parts of the packet like header and payload with different keys. A routing table is maintained at the base station in order to establish optimal routes to nodes either by single hop or multi-hop. They also address the issue of malignant nodes by keeping track of number of corrupted packets which they send. If this exceeds the limits, then the base station deprives the node of its energy by flooding it with packets to prevent further corruption of data packets. This will not lead to disconnection due to the existence of multiple routes. In this scheme also, pre-deployed keys are shared between base station and nodes. Rekeying requires physical access to nodes. Adding new nodes requires synchronization with the existing nodes.

We conclude with this section with work on Fuzzy commitment scheme proposed by Juels and Wattenberg [15]. This work is not specific to sensor networks, but it serves as a significant support to our approach to solving the problem of securing biosensor networks. In this work the authors present a commitment scheme which tolerates errors in the encryption key within a specified range. Such an encryption scheme proves to be very useful in scenarios where biometrics are used. This is due to the fact that any two readings of a biometric are rarely identical as it depends heavily on the way the human body provides them, which is not the same every time.

III. CONSTRAINTS OF THE BIOSENSOR NETWORK

In this section we describe the constraints of the biosensors. Some of the constraints of the biosensors are also experienced by generic sensors. But the constraints are far more stringent for the biosensors. In view of this nature of the constraints the security solutions proposed for the other wireless systems like generic sensors would not be suitable for biosensor networks. Hence they need solutions specific to them. The constraints are as follows:

A. Low Power

Sensors in general are subject to power scarcity. But this scarcity becomes acute in the case of biosensors. The power source of the biosensors could either be a battery or a rechargeable source of energy. Out of the two the later is a better option since battery has very limited energy. The rechargeable source of energy recharges the biosensor by means of an infra-red beam. The biosensors use the power to perform all their functions like sensing, computation, and communication. In the process they dissipate heat. This heat is absorbed by the tissue surrounding the biosensor and causing an increase in temperature. The tissue surrounding the biosensor will also get heated during the recharging. But the human tissue can tolerate only a certain degree of rise in temperature with out damage. Also there is a possibility of certain bacteria to thrive at higher temperature, which would not have been possible at the normal temperature. This places a cap on both the energy that can be expended and the degree to which the biosensor can be recharged.

B. Limited Memory

The amount of memory available to biosensor is severely limited due to size and energy consumption restrictions. It is of the order of few kilobytes. The amount of memory is limited by the small size of the biosensor. The implementation of the cryptographic routines may not consume much memory, but it the actual storage of the keying material, which takes up most part of the memory.

C. Low Computation Capability

The biosensors have low computation power. Their computation power is limited by both lack of power as well as memory. Due to lack of enough memory they cannot perform large bit computations. Also the most significant function performed by a sensor is communication of the information which has been sensed. Hence there is very less amount of energy which can be expended on computations.

D. Low Communication Rate

The most expensive operation in terms of energy is the communication operation. In comparison to communication the cost of computation is so small that it is almost

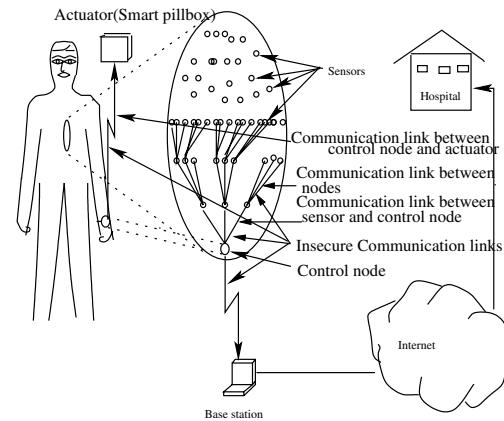


Fig. 1. System Model

negligible. Hence it very important to keep the amount of communications to the minimum. It is necessary that those communications which occur for purposes other than the actual data communication should be minimized if it is not possible to eliminate them.

IV. SYSTEM MODEL

In this section we describe our system model. The system model is as shown in the Figure 1.

The biosensor network consists of a group of biosensors implanted inside the human body, external device (control node) placed on the human body, and a base station (refer Figure 1). A network is formed by the biosensors between themselves and the control node. The control node is connected to an external base station. A *biosensor* consists of a processor, memory, transceiver, sensors/actuators and a power unit. These biosensors will perform the tasks like sensing information about the human body, processing it, and transmitting it to the control node, receiving external signals to trigger action inside the body. The control node acts as both the data aggregation and dissemination point. The control node also sends the data collected periodically to the *base station*, where it is stored for further processing.

The control node and the base station have significantly higher transmission and processing capabilities as compared to the biosensors. It is assumed about the biosensors that they have limited amount of power and cannot afford heavy computation and communication. Between computation, communication, and sensing, communication is very expensive. Hence, it is desirable if communications can be avoided at the expense of more computation and sensing. This include actions like compressing data at every stage before transmitting and deriving inputs for computation like keys from the body instead of relying on communication between nodes. Here it is assumed that the biosensors can perform multiple sensing functions simultaneously. In addition to these primary assumptions we also assume the existence of a propagation model [16] that

can be used in computing the transmission power based on the distance of communication and that not all sensors in the network will be able to communicate with the control node in a single hop (i.e.) they do so using a multi-hop link through other biosensors. This involves the issues such as routing of data between nodes and medium access. Both these are addressed by a scheme based on a combination of CDMA/TDMA [4].

There are three types of wireless communication links in the biosensor network based health care system. They are the communication links between the biosensors, the communication links between the biosensor and control node, and the link between control node and the base station. All these wireless links are considered to be insecure due to the fact that the data is available on the channel, which unlike a wired channel is accessible to anyone who cares to listen. Therefore, data exchange using any of these communication links has to be secured (with respect to authenticity, integrity, and confidentiality). As stated earlier the base station and the control node have higher computing and communication capability, hence the link can be secured by means of asymmetric cryptography. This problem is well studied and hence is not object of our focus. The problem of securing the communication between the biosensors is addressed in this paper in great detail. The link between the biosensor to control node is one which starts from node inside the human body and terminates outside the human body or vice versa. Hence securing this link has to be addressed separately. The data is secured by encrypting it by means of an encryption algorithm using a suitable key. The algorithm used is a light weight encryption algorithm such as RC5 [9].

V. PROBLEM STATEMENT

In this section, we formally define the problem of securing the wireless communication in a biosensor network. We describe the security requirements of biosensor networks and factors that determine the approach to be adopted.

A. Security Requirements of Biosensor Networks

The security requirements of biosensor networks are as follows,

- *Data Confidentiality*: The data that is communicated between biosensors is the health information, which is of personal nature. It is essential and in the interest of the individual, to keep this information from being accessed by unauthorized entities. This is referred to as confidentiality. The confidentiality of the data, especially during transmission when it is vulnerable, is achieved by encrypting the data by a key. The communicating entities, which are the only ones with the knowledge of the key can access the data.
- *Data Authenticity*: Authenticity is the property of the data by which the recipient of the data can verify and

trust that claimed sender is in reality the actual sender. This property is very important for the biosensor network because certain actions are initiated only if the legitimate nodes requested the action. Absence of this property may lead to situations where an illegitimate entity masquerades as legitimate one and reports false data to control node or gives wrong instructions to the other biosensors possibly causing considerable harm to the host.

- *Data Integrity*: It is possible that data can be modified by a hostile entity, while it is being transmitted. In this situation the data is authentic as it has originated from a legitimate source. But the consequences could be equally harmful as in the case of lack of authenticity. Data integrity is a property by which it is possible to defend against modification in data introduced by malicious intermediaries.

Key distribution is central to any security mechanism based on cryptographic techniques. All the security requirements described above can be fulfilled if a key is successfully and securely distributed. Data on encryption is unavailable to unauthorized entities thereby making it confidential. Since the key is distributed securely, it is possessed only by the legitimate parties and hence only they can encrypt and send data which would decrypt properly with the shared key. Data integrity can be obtained by sending the message digest or *MAC* of the data computed using the secret key along with data. The data is considered integral only if the *MAC* of the received data maps correctly with *MAC* accompanying it. The security of any such scheme rests on the secrecy of the key. Hence our problem is primarily that of secure key distribution. The problem of key distribution in ordinary networks has been heavily studied. In ordinary networks, with nodes possessing significant amounts of processing power and storage space, public key cryptography based schemes are used. Asymmetric cryptosystems involve heavy exponentiation making them orders of magnitude more expensive than symmetric crypto systems. Asymmetric cryptography based key exchange is not suitable for even the generic sensors. This is due to the heavy overhead associated with them.

Biosensor networks face severe constraints as described in Section III. Even the simplest of the asymmetric cryptography-based key exchange protocols available presently involve multiple exponentiations and message exchanges. The difference in the energy consumed for performing asymmetric operation and symmetric encryption is of orders of magnitude. For example using a MIPS R4400 processor, establishment of a key with a 128 bit operation of Diffie-Hellman costs 15.9mJ while symmetric encryption of same bit length on the same processor consumes 0.00115mJ of energy[13]. Hence asymmetric operations are very expensive in terms of resource consumption and are not suitable for the biosensor networks.

Pre-deploying or programming the keys into the sensors has been suggested as a solution for the sensors. In case of generic sensors this solution may be suitable because the sensors are accessible and they may be re-keyed with ease. Rekeying of the sensors is a reality because of two factors. Firstly it improves the security of the system over time. Secondly when we add more sensors to the system, re-keying is required to ensure all the sensors share the same key.

Predeployed keys may be a more promising solution for biosensors, but for the need for re-keying. Rekeying is a real possibility since biosensors for different applications may be added later on. Once implanted inside the human body the biosensors become almost inaccessible physically. In biosensors using wireless communication for re-keying is not an available option due to constraints mentioned earlier. Therefore, re-keying poses a major problem while using predeployment for biosensors.

So our *problem* is to ensure that each of the communicating parties (the biosensors in this case) possess the key with which they perform low cost symmetric key encryption by an inexpensive mechanism and not by using expensive operations such as asymmetric cryptography and fulfill the security requirements. For solving this problem we have been guided by the fact that the biosensors are placed in an unique environment.

VI. KEYS AND RANDOM NUMBERS

Keys for symmetric crypto-systems are generated using standard key generating functions [12]. These functions use pseudo random numbers as input parameters to generate unique keys. These functions are commonly known and hence the strength of the key generated depends upon the pseudo-random number.

Any pseudo-random number irrespective of the source from which it is generated, it should satisfy certain conditions for them to be used for security purposes. This characteristic is known as *cryptographic randomness*. A random number generated from a particular source is said to be cryptographically random, if it is not possible for an adversary with full knowledge of the working of the system, to determine the n^{th} number generated from the knowledge of $n - 1$ previous numbers generated from the same source with a probability greater than half. This property ensures that the random number and hence the keys generated from it cannot be guessed by an adversary.

In case of ordinary devices the pseudo-random number is generated from the hardware level and the key is generated at one node and is distributed to all the other nodes. This method is adopted because it is not possible to generate the same pseudo-random number at different nodes due to the differences in the hardware of the node.

VII. PROPOSED SOLUTION

In the earlier sections, the need for securing the biosensor network communication, and their specific security requirements were presented. We also explained how all the said requirements could be fulfilled by means of having a secret key shared between the nodes. Once all the communicating entities have the same key, it can be used to perform cryptographic functions like encryption and computing Message Authentication Code (*MAC*).

In conventional computing systems and generic sensor systems the key sharing is achieved by means of asymmetric cryptography. The symmetric key is encrypted and sent to the recipient, who decrypts it. This key is used for subsequent symmetrically encrypted communication. But this involves extensive use of exponentiation, which are mathematically intensive operations. Thus rendering this approach unsuitable for biosensor networks due to the extremely resource constrained environment in which they operate. The conventional and generic sensor networks do not take into consideration the environment in which they operate. Hence they do not attempt to make use of any resources which it may offer. If we could design architectures such that the sensors can make use of the surroundings in their computing tasks, then it would lead to significant advantages. In ubiquitous computing parlance, this is known as context aware computing.

At this point we would like emphasize on a fundamental aspect of the biosensor network. It is the fact that, biosensors are implanted in the human body which is a single entity. Thus in spite of being physically distributed, the sensors form part of centralized system constituted by the human body. Since the biosensors are part of the same body we propose a solution, wherein the same pseudo-random number is generated from the properties of the human body at different sites and is used to encrypt and decrypt the symmetric key to distribute it securely. This key can be used to achieve the requirements of security as mentioned earlier.

In the scheme described above the following two significant issues need to be addressed.

Biometric Measurement: An inherent problem with the use of biometrics is that their measurement is never perfect. A biometric when measured by the same sensor serially in time or when measured in parallel at the same time by multiple sensors, results in readings which differ from each other. These variation could be to the extent of a hamming distance of 10%. This would lead to faulty decryption on receiving end although the receiver is a legitimate one. This situation is called *truth rejection* which is undesirable. The different readings of biometrics are independent of each other. Hence this situation may be considered analogous to the one wherein error is introduced in data during transit leading to a non-zero hamming distance between data sent and received. Error correction would help in alleviating this problem of error. An (N, K, D) code, where N

is the length of the code, K is the length of actual biometric, and D is the minimum distance of the code, which can correct $T = (D - 1)/2$ errors is suitable. The number of errors can be reduced by taking multiple readings independently and using the code obtained by the *majority encoding* of those readings. The fuzzy commitment scheme mentioned in Section II incorporates error correction codes in order to protect or encrypt data. There are two phases in this scheme namely the *commit phase* and *decommit phase*. In the commit phase the entity to be protected (say) c is committed with x as proof using $F_{com}()$:

$$F_{com}(c, x) = (h(c) \parallel \delta)$$

where $\delta = x \oplus c$ (\oplus is the bitwise XOR operation) and $h()$ is a hash function. The receiver receives $h(c) \parallel \delta$ from the sender. Now the receiver decommits c using $F_{dec}(h(c) \parallel \delta, x')$ as follows. It computes $c' = f(x' \oplus \delta)$, where x' is variant version of proof x available to the receiver and f is an error correction function. Now the receiver checks if $h(c') = h(c)$. If they are equal then the receiver will go ahead and use c' in place of c . applies [15],[14].

Example: This is a simple example to explain the above scheme. Consider an error correcting code with code set $C = \{00000, 11111\}^2$, and f is a majority decoding function which decodes five bits at a time. Thus error to the extent of two bits can be corrected. Now choose $c = \{00000 \ 11111\}$ from C . Suppose the proof for committing c , $x = \{01010 \ 10101\}$. Therefore $\delta = \{01010 \ 01010\}$ and $F_{com}(c, x) = (\alpha, \delta) = (h(00000 \ 11111), 01010 \ 01010)$. Suppose the receiver has the commit proof corrupted in 2 bits i.e $x' = \{11010 \ 11101\}$. Now the decommit operation computes $f(x' \oplus \delta) = f(10000 \ 10111) = \{00000 \ 11111\} = \alpha$. Hence the decommit operation is successful [15].

Randomness of Biometric: A major concern with using biometrics for cryptographic purposes is their degree of randomness. Unless the biometric is random enough, an attacker would be able to guess and be able compromise the security of the system. This situation is referred to as false acceptance. The level of randomness is any quantity is determined by the amount its entropy [12]. Our studies of some biometrics like heart rate have indicated that the level of entropy is not satisfactory. The required entropy is obtained by deriving the sequence from multiple biometrics simultaneously. The entropy may further be increased by using a combination of readings at more than one instant of time because the search space is further increased. Some of the candidate biometrics and their ranges are as shown in Figure 2 [2]. The ranges are mentioned for normal ranges as well as conditions which are not normal.

While dealing with the above issues there is a trade-off. When we use biometrics the variation in readings leads to *truth rejection*. This tends to increase when multiple biometrics are used. But when multiple biometrics are used

Biometric	Range
Blood Glucose	64–140mg/dL(varies with activity)
Blood Pressure	120–160mmHg (systolic)(Range is from hypotension to hypertension)
Temperature	97.0–105.0 F (Range across ages and normal and abnormal conditions)
Hemoglobin	12.1–17.2g/dL(Varies between male female and age and altitudr)
Blood Flow	Greater than 0.9 ABI(normal), Less than 0.5 ABI (abnormal)

Fig. 2. Ranges of some Biometrics

the condition of false acceptance reduces. Hence a balance is to achieved between the desired *flexibility* and *security*.

It is desirable that the properties of the human body or biometrics, used for the purpose of deriving random numbers for key encryption possess the following characteristics,

- Degree of variation value throughout the human body at any particular time should be within limits tolerable by fuzzy commitment.
- It should be possible to measure them easily, accurately and precisely.
- There should be some degree of variation in value with time to ensure that the encryption key derived from it also changes.

A. Description of the Proposed Scheme

We end this section with the description of the proposed scheme. The description includes the data structures used, message formats, primitives required by the system, and finally the pseudo-code for the algorithms followed.

1) *Data Structures:* The following data structures are required and maintained at each one of the biosensor nodes,

- m_s : Stores the random number generated by from a combination of biometrics. Its length is 128 bits.
- r_u : Stores a number which is unique to the individual. Its length is 128 bits.
- K_{commit} : Stores the number used to commit the session key. It is derived by the combination of m_s and r_u . Its length is 128 bits.
- $K_{session}$: Stores the session key used for performing cryptographic routines. It is 128 bits in length.
- $Data$: Stores the data to be encrypted and sent.
- $eData$: This data structure stores the result of encrypting data with the session key $K_{session}$.
- m : Stores the result of the MAC computed upon the encrypted data $eData$.
- S_{com} : Stores the result of committing the session key $K_{session}$ with the commit key K_{commit} .

- *DataToSend* : This is a flag which is set to true when the biosensor has data to send.
- *DataToReceive* : This is a flag which is set to true when the biosensor has data to receive.

2) *Messages*: There is only a single message which is transmitted across between nodes communicating. Its format is $(eData \parallel m \parallel S_{com})$, where $eData$, m , S_{com} are as defined above. Clearly the format of a message received must also be the same. The format is as shown in the Figure 3.

3) *Keys Used*: The following keys are used in the schemes presented.

- $K_{session}$: This key is used to perform all the required cryptographic functions like encryption, decryption, computation of MAC , verification of MAC .
- K_{commit} : This key is used to commit the session key $K_{session}$ in order to hide it while sending it across.

4) *Primitives Required*: The following primitives are required for the execution of the algorithm.

- *getMetric()* : This function returns a random number generated from a combination of biometrics.
- *startNewSession()* : This function obtains the length of the next session and starts the timer for it.
- F_{com}, F_{dec} : These are the fuzzy commitment and decommitment schemes respectively explained earlier.
- $E_{K_{session}}, D_{K_{session}}$: These are the encryption and decryption routines of the RC5 algorithm [9]. The RC5 algorithm uses variable rotation for each rounds. Software implementation is cheaper since full 32 bit rotation is very expensive in hardware [17].
- MAC : This function computes the MD5 [10] hash of the input to generate a 128 bit output. The hardware implementation of MD5 which is four times faster than the software implementation is achievable. But this is achieved with significantly higher hardware cost for performing the required optimizations [11].

5) *Pseudo-code for Procedures executed*: Before deployment a timing sequence is programmed into the sensor. This sequence specifies after what interval of time the key has to be changed. For every such interval secret key is computed. This key is used by the sensor for committing the session key. This committed session key is sent along with the encrypted data. Each sensor executes two procedures, which we call *COMMIT_KEY()* and *DATA_PROCESS()*.

- *COMMIT_KEY()* is executed in response to the event of a session coming to an end. It computes the commit key K_{commit} for that session and starts the new session.

```

COMMIT_KEY()
   $m_s \leftarrow \text{getMetric}()$ 
   $K_{commit} \leftarrow m_s \oplus r_u$ 
  startNewSession()

```

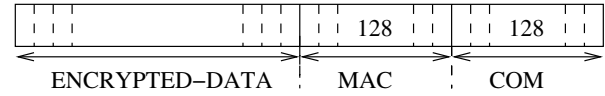


Fig. 3. Payload Structure

- *DATA_PROCESS()* is executed when a node has to send data to some other node or it has to receive data from some other node. In this procedure, if the node has data to send it encrypts it with the session key $K_{session}$. Then $K_{session}$ is committed by means of fuzzy commitment described earlier with K_{commit} which is computed in the *COMMIT_KEY()* procedure. Then the MAC of the encrypted data with the session key is computed. Then the encrypted data, its MAC and the commitment of the session key $K_{session}$ with K_{commit} is sent across. When a node receives the message, it first attempts to decommit the committed session key $K_{session}$ by using its K_{commit} . If the decommit operation is successful the MAC of the encrypted data is computed and is verified with MAC from the the encrypted data. If it does match then it is decrypted with the $K_{session}$, otherwise it is discarded.

```

DATA_PROCESS()
  if (DataToSend) then
    eData  $\leftarrow E_{K_{session}}(\text{Data})$ 
     $S_{com} \leftarrow F_{com}(K_{session}, K_{commit})$ 
     $m \leftarrow MAC(eData, K_{session})$ 
    send(eData  $\parallel$  m  $\parallel$   $S_{com}$ )
  else if (DataToReceive) then
    if ( $F_{dec}(rData.S_{com}, K_{commit})$  succeeds) then
       $K_{session} \leftarrow F_{dec}(rData.S_{com}, K_{commit})$ 
      if ( $rData.m == MAC(rData.eData, K_{session})$ )
        then
          Data =  $D_{K_{session}}(rData.eData)$ 
        else
          reject rData
      end if
    end if
  end if

```

In the above algorithms the encryption and decryption functions $E_{K_{session}}$ and $D_{K_{session}}$ both use the $K_{session}$ which is generated by the sender. This session key is committed and decommitted with K_{commit} which is computed from the biometric measured from the body. A typical communication scenario is as shown in Figure 4.

The scheme described above is for securing the communication between the biosensors. It remains to secure the communication between the biosensor and the control node. The above solution cannot be extended to this because the control node may not be able to measure the relevant biometrics. Hence to solve this problem we may have one dedicated biosensor which computes the necessary bio-

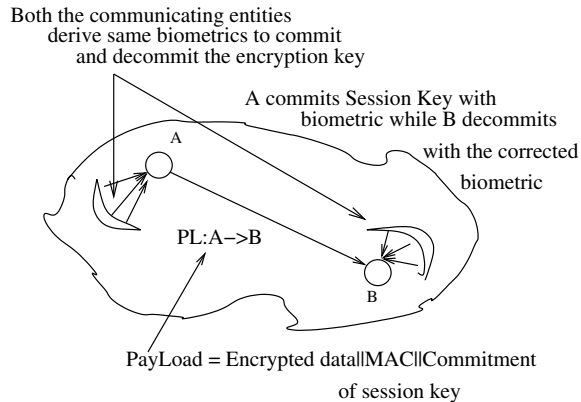


Fig. 4. Sample Biosensor Scenario

metric and sends it to the control node after encrypting it. For this encryption at this node alone a predeployed key is used. Since this node is dedicated for just this purpose the load on it is comparatively less and it can perform the required encryption and communication of just the biometric. In the above scheme we perform secure key distribution without making use of expensive computations like exponentiation and multiple rounds of communication. This would lead to a significant conservation of energy and bandwidth.

VIII. CONCLUSIONS AND FUTURE WORK

In this paper have proposed and described a scheme for secure communication wireless biosensor network. There is need for novel approaches for securing the biosensor network communication because the existing paradigm will not solve the problem due to the extremely resource constrained environment under they operate. It is necessary that the nature of the operating environment be considered in its entirety in such situations. This includes making use of the environment in computing tasks. Hence we were motivated to propose a scheme based on biometrics derived from the human body itself for securing the keying material used for achieving security objectives. We have discussed significant issues related with the use of biometric for such a purpose. These include randomness required of the biometric and the error on measurement of the biometric. We have also proposed solutions for problems posed by these two. They are the use of error correcting codes and the use multiple biometrics for securing the key for the problems of measurement errors and randomness problems. Since we eliminate the computation required and reduce the communication involved drastically as compared to traditional asymmetric key establishment techniques. The future work involves collection of relevant biometric data such as the one presented in this paper and examine their variation with time for individuals and come up with a combination of biometrics leading to sufficient randomness. The practical im-

plementation of the scheme is final step in the realization the secure biosensor network system.

ACKNOWLEDGEMENTS

This research is supported in part by National Science Foundation Grants ANI-0086020 and ANI-0196156.

REFERENCES

- [1] Health Insurance Portability Accountability Act (HIPAA)
- [2] Medline Plus Medical Encyclopedia, U.S National Library of Medicine. <http://www.nlm.nih.gov/medlineplus/encyclopedia.html>
- [3] A. Perrig, R. Szewczyk, V. Wen, D. Culler, and D. Tygar SPINS: Security Protocols for Sensor Networks In Proceedings of Seventh Annual International Conference on Mobile Computing and Networks MOBICOM 2001, July 2001.
- [4] V. Annamalai and S.K.S. Gupta and L. Schwiebert On Tree-Based Convergecasting in Wireless Sensor Networks In *IEEE Wireless Communications and Networking Conference, New Orleans, 2003*.
- [5] D. Carman, B. Matt, D. Balenson and P. Kruus "A Communications Security Architecture and Cryptographic Mechanisms for Distributed Sensor Networks" In *DARPA SensIT Workshop, NAI Labs, The Security Research Division Network Associates, Inc., 1999*.
- [6] S.K.S. Gupta and S. Cherukuri "An Adaptive Protocol for Efficient and Secure Multicasting in IEEE 802.11 based Wireless LANs". In *IEEE Wireless Communications and Networking Conference, New Orleans, 2003*.
- [7] J. Undercoffer, S. Avancha, A. Joshi and J. Pinkston. Security for Sensor Networks. In Proc. of CADIP Research Symposium, 2002.
- [8] US Secure Hash Algorithm 1 (SHA1) Internet Request for Comments RFC 3174
- [9] R. L. Rivest. The RC5 Encryption Algorithm. In *Fast Software Encryption - Second International Workshop, Leuven, Belgium, LNCS 1008*, pages 86-96, Springer-Verlag, 1995.
- [10] R. L. Rivest. The MD5 message-digest algorithm. Internet Request for Comments, April 1992. RFC 1321.
- [11] J. Touch Performance Analysis of MD5 (1995). In ACM Special Interest Group in Communication (SIGCOMM), 1995.
- [12] Cryptographic Random Numbers Standard P1363: Appendix E, November, 1995
- [13] D. W. Carman, Peter S. Kruus, Brian J. Matt Constraints and Approaches for Distributed Sensor Network Security. NAI Labs Technical Report #00-010.
- [14] G. I. Davida, Y. Frankel and B. J. Matt "On Enabling Secure Applications Through Off-line Biometric Identification" in *IEEE Symposium on Security and Privacy*
- [15] A. Juels and M. Wattenberg "A fuzzy Commitment Scheme" in *Proceedings of 6th ACM conference on Computer and communication security*
- [16] S. K. S. Gupta, S. Lalvani, Y. Prakash, E. Elsharawy, and L. Schwiebert, Towards a Propagation Model for Wireless Communication in Biomedical Applications IEEE International Conference on Communications 2003, Alaska.
- [17] B. Schneier and D. Whiting Designing Encryption Algorithms for Optimal Software Speed on the Intel Pentium Processor In Eli Biham, editor, *Fast Software Encryption '97*, volume 1267 of Lecture Notes in Computer Science, pages 242-259. Springer-Verlag, 1997
- [18] V. Shankar, A. Natarajan, S.K.S. Gupta, L. Schwiebert "Energy-efficient Protocols for Wireless Communication in Biosensor Networks". In *IEEE Personal, Indoor and Mobile Radio Communications Conference, San Diego, 2001*.