PSKA: Usable and Secure Key Agreement Scheme for Body Area Networks

Krishna K. Venkatasubramanian, *Member, IEEE*, Ayan Banerjee, and Sandeep Kumar S. Gupta, *Senior Member, IEEE*

***Abstract*—A body area network (BAN) is a wireless network of health monitoring sensors designed to deliver personalized healthcare. Securing intersensor communications within BANs is essential for preserving not only the privacy of health data, but also for ensuring safety of healthcare delivery. This paper presents *physiological-signal-based key agreement (PSKA), a scheme for enabling secure intersensor communication within a BAN in a* usable *(plug-n-play, transparent) manner*. PSKA allows neighboring nodes in a BAN to agree to a symmetric (shared) cryptographic key, in an authenticated manner, using physiological signals obtained from the subject. No initialization or predeployment is required; simply deploying sensors in a BAN is enough to make them communicate securely. Our analysis, prototyping, and comparison with the frequently used Diffie–Hellman key agreement protocol shows that PSKA is a viable intersensor key agreement protocol for BANs.**

***Index Terms*—Body area networks (BANs), physiological-signals-based key agreement (PSKA), secure communication, usable security.**

I. Introduction

Body area networks (BANs) are networks of wireless medical sensors, deployed on a person, for enabling *pervasive, individualized, and real-time* health management. As BANs deal with personal health data, securing them, especially their communication over the wireless link, is very critical (one of the main research challenges in BAN design [1]). Lack of adequate security features may not only lead to a breach of patient privacy, but also potentially allow adversaries to compromise patient safety by modifying actual data resulting in wrong diagnosis and treatment [2]. Indeed, protecting health data is a legal requirement as per the Health Insurance Portability and Accountability Act (HIPAA) (http://www.hhs.gov/ocr/hipaa/), which mandates that all personally identifiable information in electronic form be protected.

Sensors rely on cryptographic keys to secure their communication. Keys are usually made available to sensors through explicit key distribution protocols. Well-known classes of symmetric key distribution techniques [3], [4], require some form of predeployment. However, given the progressively increasing size of BANs, these approaches may potentially involve considerable latency during network setup or any subsequent adjustments, due to their need for predeployment. We believe that for BANs to be useful, they should provide usable security—one that is plug-n-play and largely transparent. For example, one should be able to add, remove, and adjust the sensors on their BAN, as and when required, without reconfiguring parts of the network (a very important requirement in mission critical environments) and still have secure communication. In some instances, asymmetric cryptosystems such as Diffie–Hellman (DH) and its variants have been used to avoid predeployment. However, they are prone to man-in-the-middle attacks and need additional authentication mechanisms to be useful.

In this paper, we present a novel scheme called *physiological-signal based key Agreement (PSKA)*, which utilizes physiological signals for enabling sensors to agree upon a pairwise symmetric cryptographic key, in an authenticated manner. It requires no *a priori* distribution of keying material, simply deploying the sensors on a subject is enough, thus facilitating secure BAN communication that is *usable*. PSKA is being designed as a part of securing the Ayushman health monitoring system [5] being developed at the IMPACT Laboratory, Arizona State University. The use of physiological signals has the potential to eliminate the need for explicit key distribution allowing constituent sensors to agree upon keys, as needed [6]. The idea of using physiological-signals-based features for key agreement comes from the observation that the human body is dynamic and complex, and the physiological state of a subject is quite unique at a given time [7]. Generally speaking, PSKA works by using physiological signal features and *fuzzy-vault* cryptographic primitive [8] to hide the key at one end, transporting it to the other, and unhiding it using the physiological signal features measured at the other end. PSKA meets the *design goals* suggested in [9] when physiological signals are used as a basis for key agreement, which are as follows.

1) *Length and randomness*: The keys agreed upon are long and random to prevent brute-forcing.
2) *Low latency*: The duration of physiological signal capture required is minimal.
3) *Distinctiveness*: Knowing the feature derived from the current value of the physiological signal of one subject will

Manuscript received October 16, 2008; revised March 20, 2009 and August 13, 2009. First published December 11, 2009; current version published January 15, 2010. This work was supported in part by National Science Foundation under Grant CNS-0617671 and Grant CT-0831544. This paper was presented in part at the IEEE Military Communications Conference 2008, San Diego.

K. K. Venkatasubramanian was with the IMPACT Laboratory, Arizona State University, Tempe, AZ 85287 USA. He is now with the Department of Computer and Information Science, University of Pennsylvania, Philadelphia, PA 19104 USA (e-mail: vkris@cis.upenn.edu).

A. Banerjee and S. K. S. Gupta are with the IMPACT Laboratory, Arizona State University, Tempe, AZ 85287 USA (e-mail: abanerj3@asu.edu; sandeep.gupta@asu.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TITB.2009.2037617

not provide significant advantage in guessing the keys being agreed by sensors on another subject. An important characteristic of distinctiveness is that it authenticates the communicating sensors by ensuring that only sensors on the same BAN can agree on a shared key.

4) *Temporal variance*: Knowing the physiological signals at any time will not provide significant advantage in knowing the keys agreed upon in future executions of the scheme. This is an important property which *differentiates the proposed technique from traditional biometrics-based techniques*, where once a template is created it is never changed [10].

The contributions of this paper are threefold: 1) a scheme for authenticated pair-wise key agreement between two nodes in BANs, i.e., PSKA (see Section IV); 2) analysis of PSKA security properties (see Section V); and 3) validation of PSKA, using actual data from two of the most commonly collected physiological signals—photoplethysmogram (PPG) and electrocardiogram (EKG), based on the aforementioned design goals (see Section VI).

II. Background

The use of physiological signals for securing intersensor communication was presented in [11]. Building upon this initial idea, Poon *et al.* [12] proposed the use of interpulse interval (IPI) to generate cryptographic keys. The advantage of using IPI is that it can be derived from multiple sources namely PPG and EKG time series by measuring the time difference between the peaks in the EKG/PPG signal. The IPI-based key generation process works as follows: 1) the sensors first measure EKG and PPG signals in a synchronized manner; 2) they then generate a series of IPI values from their respective data; and 3) they take 67 contiguous IPI values (which takes about 30 s to measure, as an EKG/PPG peak generate every 300–500 ms) from a particular start point and encode them into 128 bit to form the key_{ekg} and key_{ppg} at each of the sensor, which can then be used for secure communication between them. However, through our own experimentation with the scheme, we found that even though the keys are long and random (entropy of the generated keys are above 0.9), the average Hamming distance between key_{ekg} and key_{ppg} for the same subject is $\sim$60 and $\sim$65 bits for two different subjects. We believe the primary reason for this difference is the *topographic specificity* of the human body—physiological signals measured from different areas of the body appear to have similar trends (high correlation), but not the exact same values. As a result, the information symbols in the keys get reordered, thus leading to translational and rotational errors [8] that produce drastically different values.

We therefore take a different approach: instead of trying to generate keys from physiological signal measurements, we use them to facilitate key agreement. This is because given the dynamic nature of the human body, the chances of physiological signal features being exactly identical is low. Further, we do this by processing physiological signals in frequency domain, instead of time domain. Frequency-domain processing has many advantages: 1) frequency components of physiological signals, at any given time, have many more common values, compared to time-domain values of physiological signals, irrespective of where they are measured on the body; 2) the sample of physiological signal required for key agreement is much smaller; and 3) the level of synchronization required for measuring the physiological signals at the sensors is not very strict[1]. The aim of this paper is to show that physiological signals can be used for establishing symmetric keys between sensors in the BAN, and validate the results.

III. System Model

A BAN is a network of physiological and environmental monitoring *sensing nodes* that are worn and/or implanted on a *subject* or *individual* [3]. The sensing nodes collect health and contextual data at regular intervals and forward it over a multihop network to a highly capable *sink* node for further processing. A typical sensing node (called a *sensor*), consists of a sensing element, analog-to-digital converter, wireless communication stack, processor, and memory. We assume that the sensors *communicate wirelessly*, as wires running between sensors in a BAN will make it obtrusive. The wireless medium is, however, not trusted. All sensors are assumed to be able to measure the appropriate physiological signals. Any entity not in contact with the subject cannot measure physiological signals from the subject. We assume that only legitimate sensors are in contact with the body. Further, we assume that malicious entities cannot introduce nor compromise sensors within the BAN without being detected, as anything worn is mostly under supervision of the host or the caretaker. Therefore, the threats faced by a BAN are primarily from adversaries, who can eavesdrop on all the traffic within the BAN, inject messages, replay old messages, and spoof sensor identities. Adversaries may also try to use the physiological signal data obtained from other people to break the key distribution process. Note that, in this paper, we focus solely on securing intersensor communication within the BAN. Communication from the sink onwards can utilize conventional security schemes such as Secure Socket Layer (SSL), given the considerable capabilities of the entities involved. Finally, we do not consider denial of service (DoS) attacks such as jamming, electromagnetic interference, or battery depletion attacks in this paper.

IV. Physiological-Signal-Based Key Agreement

The purpose of *PSKA* is to *facilitate secure intersensor communication*s between two sensors by enabling them to agree upon a pair-wise symmetric key, using physiological-signal-based features. The key agreement process works as follows (see Fig. 1): one of the two sensors (sender) generates a random symmetric key that it then hides using a feature vector obtained from the physiological signal. This hidden key is sent over to

[1] Our experiments show successful key agreement even with a 1 s difference in the start times of the measurement of physiological signals using frequency domain features (see Section VI, for more on this). Prominent solutions proposed for time synchronization for sensor networks [13] achieve microsecond-level synchronization, which is more than sufficient for PSKA.

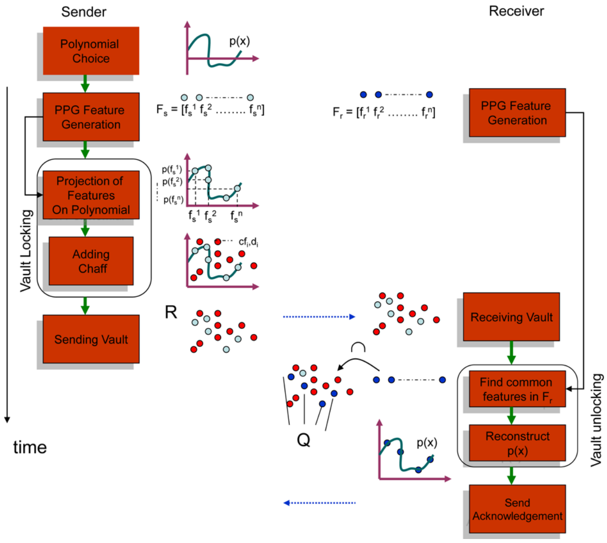

Fig. 1. PSKA protocol.

the other sensor (receiver) that uses its own version of the feature vector and obtains the random key after compensating for the differences between its feature vector and the one used by the sender. In our previous work [11], we proposed the use of simple error-correction scheme, such as majority decoding [14], as the compensatory mechanism, to arrive at a common key at the receiver. The inspiration behind the idea was the observation that each measurement of a physiological signal was independent of others; any difference in their measured values could be modeled as communication error. An inherent problem with using this approach is although it can correct the presence of a few differences in feature vectors, it cannot handle reordering of or presence of additional features (in one of the sensors) in the feature vector [8]. We address this drawback by proposing the use of a cryptographic construct called *fuzzy vault* [8].

A. Fuzzy Vault

The fuzzy-vault scheme first proposed in [8] is designed to lock (hide) a secret (S) in a construct called a *vault* using a set of values A. Once the vault has been locked, it can be unlocked only with another set of values B that has a *significant* number of values in common with set A. As illustrated in Fig. 1, the construction and locking of the vault is accomplished by: 1) generating a vth-order polynomial p over the variable x that encodes the secret S; 2) computing the value of the polynomial at different values of x from set A and creating a set $R = \{a_i, p(a_i)\}$, where $1 \leq i \leq |A|$; and 3) adding randomly generated set of points called *chaff* to R, which do not lie on the polynomial. Once the vault is constructed, unlocking it based on the set B is done by constructing a set $Q = \{(u, v) | (u, v) \in R, u \in B\}$. The unlock process is possible only if Q has a significant number of legitimate (non-chaff) points that are on the polynomial [8]. We can map this scheme onto PSKA by setting the features obtained at the sender to set A, those obtained at the receiver to set B, and generating a polynomial, whose coefficients form the secret key to be agreed upon.

Consider the following example that illustrates the operation of the fuzzy vault. Let the polynomial be $p(x) = x + 1$, set $A = \{1, 2, 3\}$, and $B = \{1, 3, 4\}$, then the vault R created by computing the polynomial's value at each point in A is $R = \{(1, 2)(2, 3)(3, 4)(4, 7)(6, 9)(7, 12)(8, 5)\}$. The last four points are the chaff points that do not fall on the polynomial. To unlock the vault, the set Q is constructed, where $Q = \{(1, 2)(3, 4)(4, 7)\}$. As the set Q has two points on the polynomial, we can use it to easily reconstruct the first-order polynomial, and thus, unlock the secret.

B. Vault Locking and Unlocking in PSKA

The use of polynomials ensures that the sets A and B need not have any order to them, as long as they have a significant number of common values. The presence of the chaff points adds security to the vault and hides legitimate points and the actual polynomial. Unless the adversary knows a large number of points on the polynomial, it cannot reconstruct the polynomial. In this section, we show how fuzzy-vault scheme for key agreement with PSKA. We use the term *sender* for the sensor that creates the vault and locks it, and the *receiver* for the sensor that unlocks the vault to access the secret key. The key agreement occurs as follows.

1) Feature Generation: First, both the sender and the receiver obtain physiological-signal-based features. This is a four step process: a) Both the sensors sample the physiological signal in a loosely synchronized manner, at a specific sampling rate for a fixed duration; b) The samples are divided into windows and a fast Fourier transform (FFT) is then performed on each of these parts; c) The FFT coefficients of each of the overlapping windows (a predefined number of contiguous signal time-series points) are then passed through a peak-detection function (a simple local maxima detector) that returns a tuple of the form $\langle k_x^i, k_y^i \rangle$, where k_x^i is the FFT point at which peak is observed (its the peak location on x-axis, also called *peak index*), k_y^i is its corresponding FFT coefficient values (magnitude of the peak, or *peak value*), and i is the index of the peaks. The number of peaks observed by a sensor vary upon situation; d) Each of these peak index (k_x) and peak value (k_y) pairs are quantized and converted into a binary string and concatenated ($[k_x|k_y]$) to form a *feature*. Individual features obtained from a single measurement are grouped together to form a feature vector $F_D = \{f_D^1, f_D^2, \dots f_D^N\}$, where $f_D^l = [k_x^l|k_y^l]$, D is either the sender (s) or receiver (r) node, and N is the size of the feature vector, i.e., number of indexes, where peaks were observed. The values of the different parameters used for feature generation are dependent upon physiological signal used and they need to be tuned during deployment. We chose FFT peaks as features for two reasons: 1) they are simple to detect and 2) they characterize a subject's physiology very well. They are ideal for distinguishing between sensors that are in the same BAN or different BANs, thus providing an efficient authentication mechanism and a basis for key agreement (see Section VI, for more details). At the end of the feature-generation process, the sender and receiver possess feature vectors of the form $F_s = f_s^1, f_s^2, \dots f_s^N$ and $F_r = f_r^1, f_r^2, \dots f_r^N$, respectively.

2) Polynomial Choice: Once the features are generated, the sender generates a vth-order polynomial of the form $p(x) = c_v x^v + c_{v-1} x^{v-1} + \cdots + c_0$, where the value of the coefficients (c_is) are selected randomly (using a pseudorandom number generator, for example). The order of the polynomial (v) used within the BAN is not a secret and known to all sensors in the network. The coefficients, concatenated together, form the secret key that the sender wants to communicate to the receiver ($\text{Key} = c_v|c_{v-1}|\ldots|c_0$). We have set the length of this Key to be 128 bits (longer keys can easily be used), and depending upon the order of the polynomial used, the coefficients are obtained by dividing the Key accordingly.

3) Vault Creation: With the polynomial and feature vector available, the sender now creates the fuzzy vault, by computing the set $P = \{f_s^i, p(f_s^i)\}$, where $f_s^i \in F_s$, and $1 \leq i \leq N$. It also computes a much larger set of M random chaff points of the form $C = \{cf_j, d_j\}$, where $cf_j \notin F_s$, $d_j \neq p(cf_j)$, and $1 \leq j \leq M$. Each chaff point cf_i is within the same range (0–2^{13}) as that of the features. Therefore, 2^{13} is the bound for the total number of points in the vault ($|R|$), which is equal to $|M| + |N|$. We refer to $|R|$ as the *vault size*.

4) Vault Locking: The sender then randomly permutes the values in the vault $R = \text{RandPermute}(P \cup C)$, to ensure that the chaff points and the legitimate points are indistinguishable. The cardinality of the set C can vary with respect to the level of security needed. The larger the set C, the more difficult it is to break the vault. Section V discusses the relationship between the vault's size and its security in more detail.

5) Vault Exchange: The sender communicates the vault R to the receiver using the following message: Sender → Receiver: IDs, IDr, R, No, MAC(Key, $R|No|$IDs). Here, IDs and IDr are the ids of the sender and receiver, respectively, No is a nonce (unique random number) for transaction freshness, MAC is a message authentication code [e.g., Hash Message Authentication Code - Secure Hash Algorithm 1 (HMAC-SHA1)], and, the key (Key) used is the one that is locked in the vault.

6) Vault Unlocking: The receiver upon receiving the vault R, first computes the set Q, where $Q = \{(b, c) | (b, c) \in R, b \in F_r\}$. It then tries to reconstruct the polynomial p based on the points in Q using the Lagrangian interpolation (as suggested in [10]), according to which, the knowledge of $v + 1$ points $\{(x_0, y_0)(x_1, y_1), \ldots, (x_v, y_v)\}$ on a polynomial allows the reconstruction of a vth-order polynomial by performing the following linear combination: $p'(x) = \sum_{j=0}^{v} y_j d_j(x)$, where $d_j(x) = \prod_{i \neq j, i=0}^{i=v} (x - x_i)/(x_j - x_i)$. For the receiver to be successfully able to unlock the vault, the condition $|Q| > v$ should hold. It then takes $v + 1$ points (from Q) at a time and tries to unlock the vault. The coefficients of the resulting polynomial are then used to verify the MAC. This not only confirms the correctness of the unlocking process, but also authenticates the sender to the receiver (confirms that the sender is on the same BAN as the receiver). This is because of the distinctiveness and temporal variance property of the physiological signal features that ensures: 1) the features generated from physiological signals for PSKA are drastically different for two different people and 2) the old vaults cannot be replayed, as the features would have changed by that time, and cannot be unlocked (see Section VI, for more details).

7) Vault Acknowledgement: If unlocking was successful, the receiver sends a reply back to the sender to inform it of its correct unlocking of the vault using the following message: Receiver → Sender: MAC(Key, $No|$ID$s|$IDr). The symbols have the same meaning as described earlier. The successful verification of the acknowledgement authenticates the receiver to the sender. This is because only a node on the BAN (receiver), which measured the same physiological signal at the same time as itself, could have unlocked the vault, given the distinctiveness and temporal variance properties of the physiological-signal-based features.

Fig. 1 shows the feature-generation process. We refer to the execution of these seven steps as an *iteration* of PSKA. The random key (Key) generated in the first step is used to enable confidential, authenticated, and integrity protected communication between sensors in a *plug-n-play manner making BANs more usable*. None of the traditional key distribution schemes [4], [15] nor physiological-signal-based approaches [12] consider this property. Further, with PSKA, no random key or physiological features are ever reused. This ensures any knowledge of past keys or physiological features (due to their temporal variance property, as seen in Section VI) of a subject cannot be used for subverting the vault.

V. Security of PSKA

Security issues in PSKA primarily arise due to its vault exchange requirement and reply it to authenticate oneself and join the BAN. An eavesdropper can record the vault and try to construct the hidden polynomial (key) from it. In this section, we discuss the security implications of the two principal aspects of PSKA: the vault and its exchange.

A. Vault Security

The use of the fuzzy-vault construct in PSKA ensures that, even though the two sensors may not have all the features in common, they can still agree upon a common key in a secure manner. The security of the PSKA scheme is based on the difficulty of polynomial reconstruction. The hiding of the legitimate feature points among much larger number of the bogus chaff points, whose values are in the same range, makes the job of identifying the legitimate points very difficult. An adversary, who does not know any legitimate points (as it cannot measure the relevant physiological signals from the host's body), has to try out each of the $v + 1$ points in set R to be able to arrive at the correct polynomial. By the same account, the more the number of features an entity is aware of, the easier it is to reconstruct the hidden polynomial, a fact exploited by the receiver to open the vault. Fig. 2 shows the strength of the vault for different values of polynomial order used for different number of chaff points. The strength of the vault is determined by the number of combinations an adversary has to try in order to find $v + 1$ legitimate points in the vault. For ease of understanding, we represent this computation requirement in terms of its equivalence to brute-forcing a key of a particular length (bits). As expected, increasing the number of chaff points, increases the security

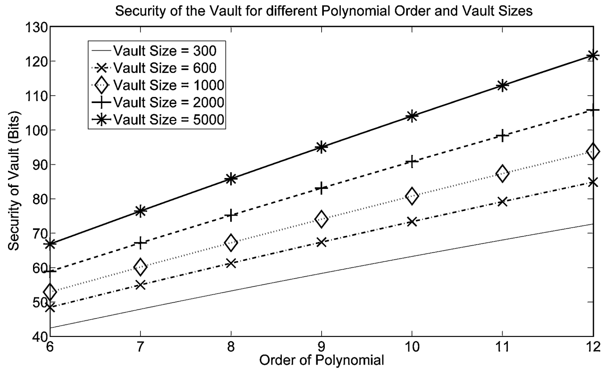

Fig. 2. Vault strength w.r.t. polynomial orders for different vault sizes.

provided by the vault. Higher the order of the polynomial, the more common features we need to find, and therefore, higher the security. Note that PSKA guarantees successful unlocking of the vault, as long as the number of common features in Q are greater than v. By choosing the order of the polynomial to a value $|F_s \cap F_r'| < v < |F_s \cap F_r|$, where $|F_s \cap F_r'|$ are the number of common features between feature vectors of two different individuals and $|F_s \cap F_r|$ are the number of common features between feature vectors for the same individual, we can ensure successful vault unlocking for the receiver, but not for the adversary.

B. Exchange Security

The vault exchange and acknowledgement phases make it very difficult for adversaries to know the key being agreed upon, because of the following reasons.

1) The presence of IDr in the vault exchange message tells the sensors in the vicinity of the sender, who the intended receiver is. The nonce No is used to maintain the freshness of the protocol, i.e., to ensure that the acknowledgement received is in response to its latest transmission.
2) If a malicious entity sends a vault exchange message (by replaying previous exchanges or creating its own vault using old physiological features), it will be discarded by any receiver, as the MAC would not match due to difference in the Key used given the temporal variation of the physiological features.
3) The vault has a many orders of magnitude more number of chaff points compared to the legitimate points (e.g., 1000 chaff points to 30 legitimate feature points), which makes it difficult for adversaries to know which points are legitimate and which are not (as discussed earlier). A malicious entity that cannot unlock the vault cannot send a valid acknowledgement, as it would have to generate a valid MAC without the Key.
4) A malicious entity trying to mount a man-in-the-middle attack has to be aware of the physiological signal features being used. Without them, any modification of the vault during exchange would be caught, as none of the $\binom{|Q|}{v+1}$ keys unlocked by the receiver will verify the MAC.

TABLE I
PSKA Feature-Generation Parameters

Signal	Parameters	Values
PPG	Sampling Sampling Duration FFT	60 Hz 12.8 secs 256 points, 5 windows[a]
EKG	Sampling Sampling Duration FFT	125 Hz 4 secs 256 points, 2 windows[b]
PPG/EKG	Peak Value Quantization Peak Index Quantization Feature Length	5 bits 8 bits 13 bits

[a] First 32 points per window were concatenated together for peak-based feature generation.

[b] First 128 points per window were concatenated together for peak-based feature generation.

VI. Performance of PSKA

We validated the PSKA approach using two of the most common physiological signals that can be collected from a person-PPG and EKG. The former is a measure of the volumetric change in the distension of arteries, due to the perfusion of blood through them during a cardiac cycle, while the latter is the representation of a subject's cardiac cycle generated by the electrical activity of the heart. The basis for validation was the meeting of our design goals set forth in Section I. We begin by discussing the data collection procedure for our experiments, followed by the analysis of the performance the two physiological signals when they are used with PSKA.

A. Data Collection and Feature Extraction

The PPG data utilized for our analysis were collected from ten volunteer subjects in the IMPACT Laboratory. We used Smith Medical pulse oximeter boards (http://www.smithsoem.com/applications/oxiboards.htm) to collect the data from the volunteers. The volunteers were asked to sit upright with their hands firmly placed on a desk; an oximeter sensor was placed on the index finger of each hand. We assume, for the purposes of our experiment, that the two communicating sensors utilize signals measured from each finger. Data were collected for 5 min from each subject at a sampling rate of 60 Hz. The EKG data (for ten subjects, from two leads of each person), on the other hand, were obtained from the PhysioBank database (http://www.physionet.org/physiobank). We assume that the two communicating sensors utilize signals from each lead for key agreement. About 15 min of data were downloaded for our analysis, with the signals sampled at 125 Hz. The PSKA implementation and analysis was done using MATLAB. Table I shows the feature-generation parameters.

B. Results

In this section, we discuss the results obtained for PSKA when used with PPG and EKG signals as the physiological signal of choice. Our aim is to demonstrate that the results follow the design goals set forth earlier.

1) Long and Random Keys: The keys to be agreed upon are generated by the sender in the form of polynomial coefficients using a pseudorandom number generator. The length and randomness of the keys agreed can, therefore, be ensured.

TABLE II
PPG/EKG FEATURE STATISTICS

Signal	Parameters	Values
PPG	Number of Iterations (1.6 s apart)	113
	Avg. Feature Vector Length	∼30
	Avg. # Common Features (Same Subject)	12
	Mode # of Features (Same Subject)	14.8
	Avg. # Common Features (Different Subjects)	2
	Mode # of Features (Different Subjects)	0.8
EKG	Number of Iterations (4 s apart)	180
	Avg. Feature Vector Length	∼87
	Avg. # Common Features (Same Subject)	24.7
	Mode # of Features (Same Subject)	25.2
	Avg. # Common Features (Different Subjects)	9.5
	Mode # of Features (Different Subjects)	8.1

2) Low Latency: The duration of sampling needed for secure key agreement depends upon the physiological signal used. With PPG (sampled at 60 Hz), our best results were obtained with 12.8 s of data. While for EKG (sampled at 125 Hz), this dropped down to 4 s of data. In general, we observe that the more detailed the data available, the lower the latency. Both of these signals outperform IPI, which requires about 30 s of data.

3) Distinctiveness: An important requirement of PSKA is that the physiological signals can distinguish people. This ensures that the vault created by a sensor in one BAN cannot be unlocked by another sensor located on another subject (either accidentally or maliciously), based on features generated from its measurements. Therefore, the number of common features for sensors on the same subject must be "significantly" more than the number of common features for sensors on the different subject. Our definition of significant is dependent upon the polynomial order v used. Table II shows the statistics observed for the features when PSKA was executed based on PPG and EKG signals. The difference between the number of common features between two sensors on the same subject and two sensors on two different subjects, is significant. Therefore, given the statistic on differences in the number of common features, we can now decide the possible values for v. The polynomial order has to be such that we minimize both the *false positives*, i.e., the number of times the common features between two people exceeds it, and the *false negatives*, i.e., the number of times the common features for the same subject is below it. Fig. 3 shows the percentage of false positives and false negatives when PPG and EKG is used with PSKA for different orders of polynomials. For PPG, the false-positive and the false-negative rates are minimized when the order of polynomial used is 6, while for EKG, it is 14. These results show that using features derived from the PPG and EKG signals to generate a vault, does not give any significant advantage to an adversary, who uses features derived from another subject, provided an appropriate polynomial order is chosen.

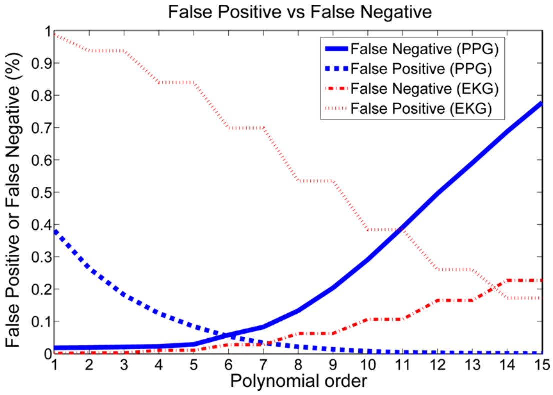

Fig. 3. False-positive versus False-negative rates for EKG and PPG.

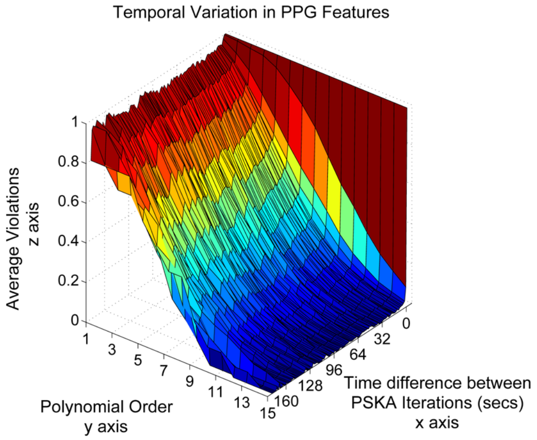

Fig. 4. Temporal variance in PPG features.

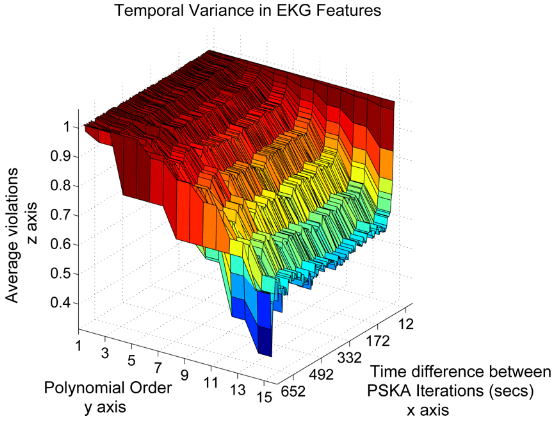

Fig. 5. Temporal variance in EKG features.

4) Temporal Variance: Figs. 4 and 5 show the temporal variance of the PPG and the EKG signal features, respectively. The x-axis of the graph is the time difference between the PPG and EKG measurement start times of two iterations of PSKA, the y-axis is the polynomial order used, while the z-axis shows the *average violations*, which is the percentage of times the number of common features between the first and second iterations of PSKA are greater than the order of the polynomial used. (We considered over 100 random start times for PPG and EKG, over all the ten subjects to compute the average violation.) As expected, when the time difference between the two iterations of PSKA for both PPG and EKG is very close, violations are very high; since the feature values in both the iterations are very similar. However, as the time difference increases, the violations fall drastically for PPG, reaching almost zero within the first few seconds for polynomials of order 9 and above. For EKG with its higher number of common peaks between two different people, the fall in violations is more gradual and does

TABLE III
PSKA COMPUTATIONAL COST (IN AVGERAGE CLOCK CYCLES), AND MEMORY FOOTPRINT

Signal	Entity	Key Gen	FFT	Feature Gen	Key Hide	Key Un-Hide	Total Cycles	Memory Footprint
PPG	Sender	9	1280	5112.2	31.95	-	6,433.15	47.35KB
	Recvr.	-	1280	5045.42	-	5454.38	11,779.80	45.3KB
EKG	Sender	12	512	22424.46	87.59	-	23,036.05	48.41KB
	Recvr.	-	512	22221.64	-	9789.28	32,522.92	46.31KB

not fall significantly before 14th-order polynomials and about 600 s time difference between two PSKA iterations. Finally, as expected for both EKG and PPG, the higher the order of polynomial, the more the number of common features needed, and therefore, lower the chance of getting violations. We can thus see that PSKA meets all our design goals. The PPG-based PSKA requires a smaller polynomial order and shows more time variance compared to EKG-based PSKA. Interestingly, the temporal variance plots for EKG and PPG also illustrate the level of synchronization required between sensors when either is used as the physiological signal of choice. We see that for both EKG and PPG, the violations are highest when the time difference between the iterations is around 1 s, irrespective of the polynomial order used. This means that the features measured 1 s apart have not changed considerably thereby still allowing successful unlocking of the vault. We can therefore say that even if the communicating sensors measure their physiological signals for PSKA a second apart (even longer for EKG), they will succeed in agreeing on a common key.

VII. Prototype Implementation

In order to estimate the cost and performance of PSKA in hardware, we prototyped it using very-high-speed integrated circuit hardware description language (VHDL). The Altera Quartus software tool was used emulating a Stratix II platform (http://www.altera.com). The details of the implementation have not be presented here for space reasons and can be found in our paper [16]. The metric used for the evaluation are: 1) CPU clock cycles and 2) memory footprint. Table III shows the average computational cost associated with our implementation of PSKA using PPG and EKG. The cost is expressed in terms of clock cycles required to perform the various tasks of PSKA: 1) polynomial coefficient generation; 2) FFT computation; 3) feature generation; 4) key hiding (polynomial evaluation and chaff point generation for the sender); and 5) key unhiding (Lagrangian interpolation). The results have been averaged by executing PSKA over 100 iterations at random start times for each of the ten subjects, in the cases of both EKG and PPG. The clock cycles required for feature generation are slightly different for senders and receivers, due to the difference in the number of features observed at each end. The receiver requires more clock cycles than the sender because of its need to perform key unhiding with Lagrangian interpolation using $v + 1$ (v is the polynomial order) points at a time from the total number of features observed (Q). Further, the key hiding stage is cheaper even though it requires chaff point generation and polynomial projection of features because it is executed in parallel with the feature generation. If done sequentially, it would require about 3000 extra cycles. Another consequence of these results is that the execution of PSKA does not affect the latency requirements of the design goal. If we assume a 8 MHz clock [as the Mote platform developed by Crossbow, Inc., (http://www.xbow.com)], the time required for executing one iteration of PSKA would be only a few milliseconds. Table III also shows the memory footprint of PPG- and EKG-based PSKA implementation. The primary component of the these footprint values are the chaff points (3000, 13-bit x-values, and 23-bit y-values), physiological features (13-bit values, about 30 for PPG, and about 85 for EKG), and their polynomial projections (23-bit values). The memory footprint for EKG-based PSKA is greater than that for PPG-based PSKA because of its larger number of features.

In order to put the performance of PSKA in perspective, we compared it with the implementation of DH and elliptic curve DH (ECDH) key agreement protocols on the same platform using VHDL. For DH, we used 1024-bit modulus and 160-bit exponent, which is equivalent to 80 bits of security in the symmetric cryptography, while for ECDH, we used a 163-bit public key, as in [15]. The computational cost and memory footprint are identical for senders and receivers in both DH and ECDH. We find DH takes 327 680 cycles, about ten times the number of clock cycles than PSKA, primarily because it requires exponentiation of large numbers. The ECDH protocol takes only 135 456 clock cycles, which is much cheaper than DH, but it still utilizes more computational cycles than PSKA, due to its elliptic-curve multiplications and additions. The memory footprint for DH (7 KB) and ECDH (2.5 KB) protocols is much lower than that for PSKA because they do not require the storage of any chaff points or features. It should be noted that both the DH protocols do not provide any form of authentication [17]. A given sensor can thus potentially agree on a key with any entity (malicious or otherwise). Therefore, any execution of the DH protocols has to be preceded by an authentication protocol, which will increase its overhead further. With PSKA, authentication is inbuilt, due to the distinctiveness property of the scheme. Hence we contend that it is feasible to implement PSKA to enable usable security in BANs.

VIII. Related Work

In our preliminary work [2], we presented PPG-based PSKA for key agreement. However, the work was limited in scope and did not include execution of PSKA using EKG, comparative analysis between EKG- and PPG-based implementations of PSKA, and a study of implementation cost of PSKA. The use of EKG directly for key generation was studied by us in [9]. However, it was later found that the way the features were extracted during the process tended to distort the original signal considerably, and therefore, it was not a sound way for key generation. Bui and Hatzinakos [18] present an approach for secure communications in BANs, which uses IPI signals and error-correction codes to arrive at the common key. However, their choice of time-domain features makes them susceptible to synchronization and feature reordering/introduction issues.

The fuzzy-vault scheme has so far been primarily applied to biometric-based authentication, such as fingerprints [10] and iris images [19]. However, the nonvariant biometric template opens up the fuzzy vault to attacks involving template modification [20]. Similarly, Kholmatov and Yanikoglu [21] present an attack where the attacker intercepts two vaults generated from the same biometric (fingerprint) data with different chaff points and correlates them to reveal the hidden biometric features. Both the attacks succeed because the fingerprint features in vaults do not change. These attacks are not possible with PSKA because the feature values in the fuzzy vaults generated in two iteration of PSKA are drastically different due to the temporal variance property of the physiological signals used. Mihailescu [22] suggests a brute-force attack on the fuzzy vault, which bounds the number of operations required to guess the legitimate points in a fuzzy vault, with a high probability, using a polynomial of order v to $8v\log^2 v(|R|/|N|)^v$. The consequence of this result is that with a probability close to 1, the complexity of identifying the polynomial used by the vault decreases. Using our parameters for PPG, PSKA is now only as secure as brute-forcing a 75-bit key rather the original 95-bit key, which is stillfairly strong. The EKG signal similarly provides a security of about 80 bits. In both cases, if we increase the number of chaff points, we can again increase the complexity of breaking the vault.

IX. Conclusion

In this paper, we presented a usable and secure key agreement scheme for BANs called PPSKA. It allows two sensors to agree on a shared key, in an authenticated manner, without any form of initialization or setup. Simple deployment of the sensors is enough to allow them to agree upon a common key, in a transparent manner. The security analysis of the PSKA protocol showed that physiological signals meet the design goals for key agreement namely, *length and randomness*, *low latency*, and *distinctiveness*. We analyzed the performance and cost of using the PSKA protocol by prototyping it in VHDL and concluded that PSKA is a viable approach to secure key agreement in BANs. A recent finding on the sustainability of PSKA via energy scavenging techniques from the human body [23] also supports its viability. Future work includes an expanded in-field study of PSKA to better understand the distinctiveness and temporal variance properties of the scheme. A detailed discussion of many of the issues of this paper can found in the extended version of this paper at http://impact.asu.edu/pub.html.

Acknowledgment

The authors would like to thank the area editor Dr. W. Wu and the anonymous reviewers for their helpful suggestions.

References

[1] L. Schwiebert, S. K. Gupta, and J. Weinmann, “Research challenges in wireless networks of biomedical sensors,,” in *Proc. 7th Int. Conf. Mobile Comput. Netw. (MobiCom 2001)*, Rome, Italy, pp. 151–165.

[2] K. Venkatasubramanian, A. Banerjee, and S. K. S. Gupta, “Plethysmogram-based secure inter-sensor communication in body area networks,” in *Proc. IEEE Military Commun. Conf.*, Nov. 2008, pp. 1–7.

[3] F. Adelstein, S. K. S. Gupta, G. G. Richard, and L. Schwiebert, *Fundamentals of Mobile and Pervasive Computing*. New York: McGraw-Hill, 2005.

[4] S. Zhu, S. Setia, and S. Jajodia, “LEAP+: Efficient security mechanisms for large-scale distributed sensor networks,” *ACM Trans. Sens. Netw. (TOSN)*, vol. 2, no. 4, pp. 500–528, Nov. 2006.

[5] K. Venkatasubramanian, G. Deng, T. Mukherjee, J. Quintero, V. Annamalai, and S. K. S. Gupta, “Ayushman: A wireless sensor network based health monitoring infrastructure and testbed,” in *Proc. IEEE Int. Conf. Distrib. Comput. Sens. Syst.*, Jun. 2005, pp. 406–407.

[6] K. Venkatasubramanian and S. K. S. Gupta, “Physiological value based efficient usable security solutions for body sensor networks,” *ACM Trans. Sens. Netw. (TOSN)*, to be published.

[7] B. J. West, “Studies of nonlinear phenomena in life sciences,” in *Where Medicine Went Wrong: Rediscovering the Path to Complexity 11*. Singapore: World Scientific, 2006.

[8] A. Juels and M. Sudan, “A fuzzy vault scheme,” in *Proc. IEEE Int. Symp. Inf. Theory*, 2002, p. 408.

[9] K. Venkatasubramanian, A. Banerjee, and S. K. S. Gupta, “ EKG-based key agreement in body sensor networks,” in *Proc. 2nd Workshop Mission Crit. Netw.*, Apr. 2008, pp. 1–6.

[10] U. Uludag, S. Pankanti, and A. K. Jain, “Fuzzy vault for fingerprints,” in *Proc. Audio- Video-based Biometric Person Authentication*, Jul. 2005, pp. 310–319.

[11] 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,” in *Proc. Wireless Security Privacy Workshop*, Oct. 2003, pp. 432–439.

[12] 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 Commun. Mag.*, vol. 44, no. 4, pp. 73–81, Apr. 2006.

[13] J. Elson, L. Girod, and D. Estrin, “Fine-grained network time synchronization using reference broadcasts,” in *Proc. 5th Symp. Oper. Syst. Des. Implementation*, 2002, pp. 147–163.

[14] A. Juels and M. Wattenberg, “A fuzzy commitment scheme,” in *Proc. ACM 9th Conf. Comput. Commun. Security*, Nov. 1999, pp. 28–36.

[15] D. J. Malan, M. Welsh, and M. D. Smith, “A public-key infrastructure for key distribution in TinyOS based on elliptic curve cryptography,” in *Proc. IEEE 2nd Int. Conf. Sens. Ad Hoc Commun. Netw.*, Oct. 2004, pp. 71–80.

[16] A. Banerjee, K. Venkatasubramanian, and S. K. S. Gupta, “Challenges of implementing cyber-physical security solutions in body area networks,” presented at the 4th Int. Conf. Body Area Netw., Los Angeles, CA, Apr. 2009.

[17] A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone, *Handbook of Applied Cryptography*. Boca Raton, FL: CRC Press, Oct. 1996.

[18] F. M. Bui and D. Hatzinakos, “Biometric methods for secure communications in body sensor networks: Resource-efficient key management and signal-level data scrambling,” in *Proc. EURASIP J. Adv. Signal Process.*, 2008, pp. 1–16.

[19] E. S. Reddy and I. R. Babu, “Authentication using fuzzy vault based on iris textures,” in *Proc. 2nd Asia Int. Conf. Model. Simul.*, 2008, pp. 361–368.

[20] W. J. Scheirer and T. E. Boult, “Cracking fuzzy vaults and biometric encryption,” in *Proc. Biometrics Symp.*, Sep. 2007, pp. 1–6.

[21] A. Kholmatov and B. Yanikoglu, “Realization of correlation attack against the fuzzy vault scheme,” *SPIE Security, Forensics, Steganography, and Watermarking of Multimedia Contents X*, vol. 6819, pp. 1–7, Jan. 2008.

[22] P. Mihailescu. (2007, Aug.). The fuzzy vault for fingerprints is vulnerable to brute force attack. The Computing Research Repository. [Online]. Available: http://www.citebase.org/abstract?id=oai:arXiv.org:0708.2974

[23] K. Venkatasubramanian, A. Banerjee, and S. K. S. Gupta, “Green and sustainable cyber-physical security solutions for body area networks,” in *Proc. 6th Int. Workshop Wearable Implantable Body Sens. Netw. (BSN 2009)*, Washington, DC, pp. 240–245.

Krishna K. Venkatasubramanian (S'06–M'10) received the B.S. degree in computer science from Webster University, St. Louis, MO, and the M.S. and Ph.D. degrees in computer science from Arizona State University, Tempe.

He is currently a Postdoctoral Researcher with the Department of Computer and Information Science, University of Pennsylvania, Philadelphia. His research interests include secure cyber-physical systems, body area networks, trust management, and medical device security. His publication list is available at http://www.seas.upenn.edu/∼vkris/

Dr. Venkatasubramanian is a member of the Association for Computing Machinery.

Ayan Banerjee received the B.E. degree in electronics and telecommunication engineering from Jadavpur University, Kolkata, India. He is currently working toward the Ph.D. degree with the School of Computing, Informatics, and Decision Systems Engineering, Arizona State University, Tempe.

Since Fall 2007, he has been with the Intelligent Mobile and Pervasive Applications and Computing Technologies Laboratory, Arizona State University. His current research interests include safety and security of cyber-physical systems.

Sandeep Kumar S. Gupta (S'93–M'95–SM'00) received the B.Tech degree in computer science and engineering (CSE) from the Institute of Technology, Banaras Hindu University, Varanasi, India, the M.Tech. degree in CSE from Indian Institute of Technology, Kanpur, India, and the M.S. and Ph.D. degrees in computer and information science from Ohio State University, Columbus.

He is currently a Professor with the School of Computing, Informatics, and Decision Systems Engineering, Arizona State University, Tempe, where he heads the IMPACT Laboratory. His current research interests include dependable, criticality-aware, adaptive distributed systems with emphasis on wireless sensor networks, thermal and power-aware computing and communication, and pervasive healthcare. He has coauthored the book *Fundamentals of Mobile and Pervasive Computing* (McGraw Hill). He is a member on the editorial board of the IEEE Communication Letters, the IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS and *Springer Wireless Networks*.

Dr. Gupta is a member of the Association for Computing Machinery.