Lightweight Mutual Authentication among Sensors in Body Area Networks through Physical Unclonable Functions

Liping Xie
School of Computer, Electronic, and Information
Guangxi University,
Nanning, Guangxi, China
Email: xie5012@mail.gxu.cn

Weichao Wang
Dept. of Software and Information Systems
UNC Charlotte
Charlotte, NC 28223
Email: wwang22@uncc.edu

Xinghua Shi
Dept. of Bioinformatics and Genomics
UNC Charlotte
Charlotte, NC 28223
Email: x.shi@uncc.edu

Tuanfa Qin
Key Laboratory of Multimedia & Network Technology
Guangxi University,
Nanning, Guangxi, China
Email: tfqin@gxu.edu.cn

***Abstract*—Medical sensors are usually attached to or implanted inside patient body. Since sensing results of Body Area Networks (BAN) can directly impact the control of medical equipment, the authenticity and integrity of sensing data is essential for safety of patients. Restricted by the limited resources available to BAN sensors, researchers have referred to Physical Unclonable Function of the nodes to achieve authentication. Existing approaches focus on the authentication between control unit and sensors. Mutual authentication among body sensors has not been carefully studied.**

In this paper, we propose to design a lightweight mutual authentication mechanism for BAN sensors with physical unclonable functions (PUF). Using control unit as a middle point, a pair of body sensors can establish shared secrets so that authenticity of exchanged data can be protected. The proposed approach does not require sensors to conduct any encryption operations, which suits the restricted resources available to BAN nodes. The analysis shows that the proposed approach has very low overhead and does not introduce new vulnerabilities into the system.

***Keywords*-body area network; sensor authentication; physical unclonable function**

I. Introduction

Among many application scenarios of Cyber-Physical Systems (CPS), intelligent healthcare is a very promising research direction. Deployment of smart sensors and treatment equipment on patient body will allow both healthcare providers and patients to better monitor the body status and respond more promptly to any changes. Such applications evolve into body area networks (BAN) [1]. A BAN is usually a wireless network formed by lightweight, small-size, ultra-low-power, and intelligent wearable devices. These sensors can be strategically placed on the body surface, around body, or implanted inside body. To reduce physical constraint on patients, the sensors transmit collected information to a control unit (cu), which is usually deployed outside of yet close to the patient. An example of BAN is shown in Fig. 1

Since intelligent sensors in BAN monitor vital signs of patients and could suggest or even directly perform medical treatment, the authenticity and integrity of collected information and operation commands are essential for safety of patients [2], [3]. Unfortunately, restricted by the size of and available power to BAN sensors, it is hard to adopt cryptography based security mechanisms in BAN. Therefore, researchers refer to hardware based mechanisms to enforce security. For example, due to differences in manufacturing, each hardware device may demonstrate some unique features even when they are built with the same design. This property leads to the discovery of "Physical Unclonable Function (PUF)" [4]. When the same challenge is provided to different devices of the same type, the PUF will guarantee that the responses produced by different devices are far apart with high probability. Therefore, PUF can be viewed as a 'fingerprint' of the device. The unique response produced by a device can be used for key generation or authentication [5].

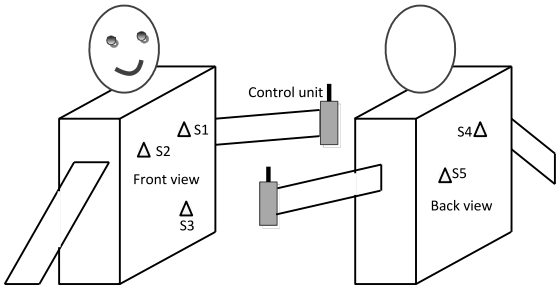

Figure 1. An example BAN setup: cu and 5 sensors.

The special properties of PUF make them a desirable candidate for security enforcement in Internet of Things (IoT). For example, in [6], the authors assume that an IoT device and the server share a challenge-response pair (CRP). This shared secret is used for mutual authentication and distribution of the next challenge-response pair. This approach works effectively in the environments in which a client-server model is used. For example, the challenge-response pair results can be pre-loaded into the cu when the BAN sensors are deployed. This mechanism, however, will not work for authenticated communication among sensors when we consider the dynamics in IoT environments.

In this paper, we propose to design a lightweight mutual

authentication scheme for sensors in Body Area Networks based on PUF. This problem arises from real scenarios. In a BAN, the body sensors may report measurement results to control unit and wait for further commands. However, to enable fast responses to some medical conditions, authenticated device-to-device communication could be a more attractive scheme. For example, if a sensor detects that the patient's glucose level drops suddenly, it needs to immediately notify the insulin pump to stop functioning. Any delay or single point of failure caused by the control unit may lead to life-threatening consequences.

Since sensors could be dynamically added into or removed from a BAN network, pre-distribution of challenge-response pairs to sensors will not work. In this paper, we propose a secret establishment mechanism among the BAN sensors with the help of the control unit. Using PUF functions of different sensors, unique CRP pairs between every pair of sensors could be generated. The shared secret will enable them to verify the authenticity of exchanged information. During this procedure, the control unit uses pre-distributed CRP to protect confidentiality of the established secrets. Each CRP will be used by the control unit only once to prevent information leakage. Our analysis shows that neither of the BAN sensors can learn extraneous information about the CRPs of the other node.

The contributions of the paper are as follows. First, existing authentication mechanisms for BAN networks focus on the connections between sensors and control unit. It cannot be generalized to device-to-device authentication. Our approach can establish shared secrets between any pair of BAN sensors in order to protect the communication between them. From this point of view, we solve a different problem from state-of-the-art mechanisms. Second, this approach does not require the BAN sensors to conduct any symmetric/asymmetric encryption operations. Therefore, it suits the limited resources available to such networks. Third, our proposed approach introduces a limited amount of overhead to the BAN sensors without causing new security vulnerabilities. Properties two and three are essential for potential deployment of our approach in resource-constrained BAN networks.

The remainder of the paper is organized as follows. In Section II we present related work. In Section III, we describe the details of our approach. The procedure to generate secrets among BAN sensor pairs is presented. A new pair of CRP will be generated and provided to the control unit by each sensor so that each CRP will be used for only once. In Section IV we discuss the safety and overhead of our approach. Finally, Section V concludes the paper.

II. Related Work

The reason that each device demonstrates a different PUF is because even though the manufacturing procedure is the same, each integrated circuit could still be impacted by the variability of the procedure. Such variability can be leveraged to become a hardware 'signature'. Because of the randomness, it is almost impossible to manufacture two identical chips. In addition to using uniqueness of IC, researchers have also designed implementations of PUF upon the special features of Printed Circuit Board [7] and SRAM and DRAM in computers [8], [9].

The uniqueness of PUF is usually demonstrated through the challenge-response pairs (CRP). Here the PUF is treated as a black-box. When a challenge c is received by the PUF, a response r will be generated. Each device will generate a different CRP (c, r) even when the challenge is the same.

Depending on the number of CRP a PUF can generate, we can classify them into weak PUF and strong PUF. For a weak PUF, the function can generate only a limited number of CRP and it is not hard to enumerate all possible challenges. For weak PUF, the values of CRP must be carefully guarded since the disclosure of the information may lead to node impersonation attacks. For strong PUF, a large number of CRP can be generated and an adversary cannot enumerate all possible challenges. Under this condition, new CRP can be continuously generated for subsequent operations.

Research efforts on using PUF for authentication can be traced back to [10]. In [4], the authors described several ways to implement PUF in hardware and using the technique to achieve device authentication and key generation. To avoid information leakage during authentication, Frikken *et. al.* [5] proposed to use zero-knowledge-proof in the authentication procedures. They also designed a mechanism that required physical contact with the device during authentication to defend against man-in-the-middle attack. Most of the authentication mechanisms described above assume a resource-tight prover and a resource-rich verifier. In [11], the authors flip the assumption and design an authentication protocol for resource tight verifiers. A survey of the authentication mechanisms based on strong PUF can be found in [12].

The application scenario of our approach is different from the cases described above since both the prover and the verifier have very limited resources. We plan to design an authentication mechanism between sensors with the help of control unit in BAN.

III. The Proposed Approach

A. System Assumptions

Our BAN network consists of n sensors and one control unit (cu). Sensors and cu use wireless technique to communicate with each other. Sensors are carried on a patient's body so that they can continuously measure her physiological data. When the patient moves, the sensors on her body will also move. Each sensor supports a strong PUF that can generate a response with a random challenge. Responses of different sensors to the same challenge could be quite different. We assume that the sensors can conduct

secure hash function. However, the nodes do not support any symmetric or asymmetric encryption functions. Sensors in BAN are not malicious but curious. This assumption is supported by the cases in which hardware manufactures try to collect user data through wearable devices.

The control unit (cu) is in charge of information collection and aggregation of the BAN. Based on the information processing results, cu could choose to transmit data to physicians or caregivers. It could also upload data to some cloud servers so that different persons in need could access it. Before a sensor is deployed in a BAN, it needs to exchange some information with the cu through a secure channel (e.g. physical contact). We assume that the cu could not be compromised.

An attacker may compromise some BAN sensors and get access to all data stored in the nodes. More importantly, the attacker could fully control the wireless communication component of the compromised sensor and send out packets in other nodes' names. The attacker may also eavesdrop on the communication channel of the BAN network and try to impersonate the control unit or a sensor. While there is research on modeling the PUF functions [13], [14], [15], here we assume that an attacker cannot generate new valid challenge-response pairs based on eavesdropped ones.

B. The Proposed Approach

The overall objective of the research is to design a mechanism through which two BAN sensors equipped with PUF functions can establish shared secrets for authenticity verification of exchanged information. Here we focus on the authenticity of the information instead of its confidentiality. Therefore, the established secret will not be used as encryption keys. Considering the special properties of a BAN network, this procedure will be accomplished with help from the control unit.

• System Initiation

We assume that each sensor in a BAN has a unique ID s_i, $(i = 1 \dots n)$. The control unit has the name cu. The PUF function owned by sensor s_i can be represented as a general function $F_i()$. Here a challenge-response pair (CRP) (c, r) of $F_i()$ can be represented as $r = F_i(c)$. Since during the system initiation procedure we cannot predict which sensors will be added into the network later, every sensor will only share some CRPs with the control unit cu. Here we assume that cu stores m CRPs $(c_{i,1}, r_{i,1})$, $(c_{i,2}, r_{i,2})$, $\dots$, $(c_{i,m}, r_{i,m})$ for each sensor s_i. (Please note that $r_{i,j} = F_i(c_{i,j})$, $j = 1 \dots m$). In addition to these CRPs, for each sensor s_i the control unit will also have $(s_i, F_i(s_i))$. Here the challenge is the ID of the sensor and the response can be used for authentication between cu and s_i in the future. At this point, the sensor can be deployed in the network. Please note that since the PUF $F_i()$ is an internal property of s_i, the sensor does not need to keep any CRPs in its storage. The following table summarizes the symbols that we use.

cu	ID of the control unit
s_i	ID of sensor i
$F_i(x)$	PUF function of s_i with input x
$F_i(s_i)$	secret shared between cu and s_i
$F_i(s_j)$, $F_j(s_i)$	secrets shared b/w s_j and s_i
$(c_{i,j}, r_{i,j})$	the jth challenge-response pair of s_i

• Security Requirements of the Approach

In the investigated scenario, we propose to help sensor pairs to establish secrets based on their PUF functions. For example, for sensor s_2, it will receive the challenge-response pair $(s_2, F_1(s_2))$ from the node s_1. Similarly, node s_1 will receive the value of $F_2(s_1)$ from s_2. Since the values are derived from the PUF functions, only one side needs to store the secret while the other one can re-calculate it in real time. The two nodes can then use the shared secrets to protect integrity and authenticity of exchanged information.

While the objective is clear, we face several difficulties when we design the mechanism. First, since the two sensors do not have any pre-distributed information from each other, the secret establishment procedure needs help from the control unit. Second, to suit the environments of real BANs, we assume that the sensors do not support any symmetric or asymmetric encryption algorithms. Therefore, the confidentiality of the secrets must be carefully protected through some other methods. Last but not least, during the distribution procedures, the control unit and sensors need to use CRPs to hide the secrets. To prevent attackers from learning information from these messages through eavesdropping, a CRP needs to be replaced after its usage. Below we describe the details of the design. Without losing generality, we assume that the control unit wants to help sensors s_1 and s_2 to establish the secrets.

• Request the Value from PUF Owner

When the control unit learns that sensor s_2 wants to communicate with s_1 directly, it will request $F_1(s_2)$ from the node s_1 and transmit the value to s_2. As we discuss earlier, cu has received m challenge-response pairs $(c_{1,1}, r_{1,1})$, $(c_{1,2}, r_{1,2})$, $\dots$, $(c_{1,m}, r_{1,m})$ from s_1 during system initiation. During the information exchange procedure, cu and s_1 will use the CRP $(c_{1,1}, r_{1,1})$ to hide the value of $F_1(s_2)$. The communication protocol looks as follows.

First, cu will send a request to s_1 for $F_1(s_2)$. Since the message is transmitted in clear text, s_1 needs the sender to prove its identity. Therefore, in step (2) s_1 sends a nonce y_1 to cu. The authenticity of the random number is protected by the keyed hash $hash(F_1(s_1), y_1)$ since only s_1 and cu have the value of $F_1(s_1)$. In step (3), cu sends out the sensor's name s_2, a new nonce y_2, and the challenge $c_{1,1}$ that it wants to use to hide the secret. The authenticity of the message is again protected by the keyed hash value. Since the hash value covers both y_1 and y_2, s_1 will be able to verify the identity of cu and freshness of the message.

$$
\begin{aligned}
&(1)\ cu \rightarrow s_1 :\ I\ need\ a\ secret\ for\ your\ communication\ with\ s_2;\\
&(2)\ s_1 \rightarrow cu :\ random\ number\ y_1,\ hash(F_1(s_1),\ y_1);\\
&(3)\ cu \rightarrow s_1 :\ s_2,\ c_{1,1},\ y_2,\ hash(F_1(s_1),\ y_1,\ y_2,\ s_2,\ c_{1,1});\\
&(4)\ s_1 \rightarrow cu :\ hash^2(r_{1,1}) \oplus F_1(s_2),\ hash(F_1(s_1),\ y_1,\ y_2,\ F_1(s_2));
\end{aligned}
$$

Now s_1 will calculate $F_1(s_2)$, which will be its shared secret with s_2. In step (4), s_1 will use hash function and Exclusive-Or operation to hide the values. Please note that s_1 uses $hash^2(r_{1,1})$ to protect $F_1(s_2)$. The reason will be explained in subsequent sections. Since cu has the CRP pair $(c_{1,1}, r_{1,1})$, it will be able to recover $F_1(s_2)$ from the received message. The authenticity of the message is protected by the hash value with $F_1(s_1)$. The value of $F_1(s_2)$ will be delivered to s_2 for secure communication with s_1 in the next step. If s_2 has eavesdropped on the communication channel, it will be able to derive out $hash^2(r_{1,1})$ with an XOR operation. However, if the hash function is secure, s_2 will not learn $r_{1,1}$ or $hash(r_{1,1})$.

• Deliver the Secret to s_2

After cu receives the secret $F_1(s_2)$ from the sensor s_1, it needs to send the value to s_2. When we design the data transmission procedure, an issue needs to be carefully handled. Since the sensor s_1 generates the value $F_1(s_2)$, we need to guarantee that s_1 cannot derive out any secret of s_2 through eavesdropping on the communication channel.

The secret delivery procedure works as follows. In this group of messages, s_2 will first request the secret from cu. Since cu needs s_2 to prove its identity, it sends a random number y_3 to s_2 in step (2). It will also identify a challenge $c_{2,1}$ with which it will hide the secret. The integrity of the message is protected by the value of $F_2(s_2)$ shared between cu and s_2. After receiving message (2), s_2 understands that the CRP pair $(c_{2,1}, r_{2,1})$ will be used to hide the information. It sends a new random number y_4 to cu. The keyed hash result of message (3) covers the secret $F_2(s_2)$, the two random numbers, the identity of s_1, and the challenge $c_{2,1}$. After cu receives message (3), it will first verify the authenticity of the message. If it confirms the identity of s_2, it will calculate the exclusive-or result of $F_1(s_2)$ and $hash^2(r_{2,1})$. When s_2 receives message (4), it will calculate $r_{2,1}$ and recover $F_1(s_2)$.

$$
\begin{aligned}
&(1)\ s_2 \rightarrow cu :\ please\ give\ me\ the\ secret\ for\ secure\ communication\ with\ s_1;\\
&(2)\ cu \rightarrow s_2 :\ random\ number\ y_3,\ c_{2,1},\ hash(F_2(s_2),\ y_3,\ c_{2,1});\\
&(3)\ s_2 \rightarrow cu :\ y_4,\ hash(F_2(s_2),\ y_3,\ y_4,\ s_1,\ c_{2,1});\\
&(4)\ cu \rightarrow s_2 :\ hash^2(r_{2,1}) \oplus F_1(s_2),\ hash(F_2(s_2),\ y_3,\ y_4,\ F_1(s_2));
\end{aligned}
$$

• Distribute new CRP pair

During the exchange of $F_1(s_2)$, both sensors s_1 and s_2 use a CRP pair to hide the information. To avoid future information leakage, such pairs should be replaced with a new pair. In this part, we present the mechanism for this update. Without losing generality, we assume that cu asks s_1 for a new challenge-response pair. For clarity of presentation, below we illustrate it as a separate procedure. In real applications, it can be merged with the request and delivery procedures of $F_1(s_2)$.

$$
\begin{aligned}
&(1)\ cu \rightarrow s_1 :\ I\ need\ a\ new\ challenge\ response\ pair\ to\ replace\ c_{1,1};\\
&(2)\ s_1 \rightarrow cu :\ random\ number\ y_5,\ new\ challenge\ c_{1,(m+1)},\ hash(F_1(s_1),\ y_5,\ c_{1,(m+1)});\\
&(3)\ cu \rightarrow s_1 :\ random\ number\ y_6,\ hash(F_1(s_1),\ y_5,\ y_6,\ c_{1,1},\ c_{1,(m+1)});\\
&(4)\ s_1 \rightarrow cu :\ hash(r_{1,1}) \oplus r_{1,(m+1)},\ hash(F_1(s_1),\ y_5,\ y_6,\ c_{1,(m+1)},\ r_{1,(m+1)});
\end{aligned}
$$

In the exchanged messages, cu first tells s_1 that it wants to replace the challenge $c_{1,1}$ and its response. In step (2), s_1 generates a random number y_5 to verify the identity of cu. It also provides the new challenge $c_{1,(m+1)}$. In step (3), cu generates a random number y_6 and calculates keyed hash of the random numbers and the challenges. In step (4), s_1 will first verify the authenticity of message (3) through examining the hash value. It will then calculate the response $r_{1,(m+1)}$ of the challenge $c_{1,(m+1)}$. The new response is protected through an XOR operation with $hash(r_{1,1})$. A keyed hash value of the random numbers and the new CRP pair is attached to protect the integrity of the information. When cu receives the message, it can use $r_{1,1}$ to recover the new response $r_{1,(m+1)}$ and discard the old CRP pair. In this way, the total number of CRP pairs that cu knows for s_1 remains unchanged.

Now let us re-examine the design of the protocol for safety. When cu receives the secret $F_1(s_2)$ and the new CRP response $r_{1,(m+1)}$ from s_1, it needs the knowledge of $r_{1,1}$ to recover the information. However, the two values

are protected by $hash^2(r_{1,1})$ and $hash(r_{1,1})$, respectively. The reason is as follows. If s_2 has eavesdropped on the network, it will get $hash(r_{1,1}) \oplus r_{1,(m+1)}$ and $hash^2(r_{1,1}) \oplus F_1(s_2)$. Since it will get $F_1(s_2)$ from cu, it can derive out $hash^2(r_{1,1})$ with an XOR operation. However, if the hash function is secure, s_2 will not learn $r_{1,1}$ or $hash(r_{1,1})$. Under this condition, only cu will be able to recover the new CRP pair $(c_{1,(m+1)}, r_{1,(m+1)})$.

• **Authenticated Communication between Sensors**

While the protocol above introduces only the distribution of $F_1(s_2)$ from s_1 to s_2, a similar procedure can be followed for s_1 to get $F_2(s_1)$. Once the sensors exchange their PUF secrets, they can use the values to protect the authenticity of data traffic between them as follws.

$$s_1 \rightarrow s_2 : I\ have\ data\ to\ report\ to\ you;$$
$$s_2 \rightarrow s_1 : random\ number\ y_7;$$
$$s_1 \rightarrow s_2 : (data\ d_{s1},\ hash(F_1(s_2),\ F_2(s_1),\ d_{s1},\ y_7)$$

Here the nonce y_7 generated by s_2 is used to prevent replay attacks. The keyed hash value covers the two secrets shared between s_1 and s_2, the random number y_7, and the data entry d_{s1}. The authenticity and freshness of the data are protected.

C. Support of Network Dynamics

As a kind of sensor networks, the BAN network could also experience dynamics in nodes. Below we describe the secret revocation and update procedures when such changes happen.

• **Node Removal and Secret Revocation**

When a BAN sensor needs to be removed from the network, the control unit will notify all remaining sensors of the change. Without losing generality, we assume that node s_2 needs to be removed. When cu notifies all remaining sensors in the network with authenticated messages, all sensors will check their storage devices. For example, node s_1 will search in its storage device and find that it has established a pair wise secret $F_2(s_1)$ with s_2. s_1 will just delete the information. Later, if s_2 tries to communicate with s_1 again, s_1 cannot respond directly since it no longer has the pairwise secret. It has to ask for help from cu. cu can check the list of removed nodes and reject the request. In this way, the sensors do not need to keep a record of the removed nodes.

Please note that the removed nodes may still store the pair wise secrets that it established with other nodes. For example, s_2 may still have the secret $F_1(s_2)$. However, the remaining sensors in the system will not use those secrets to send out data. Therefore, no further information leakage will happen. At the same time, since the PUF functions are different from those polynomial based key distribution mechanisms, multiple revoked nodes cannot pool their information together to derive out new CRP pairs of another sensor.

• **Adding New Sensors into the Network**

Before a node can be added into the network, it needs to go through the initiation procedure with the cu as described in Section III.B. This procedure needs to go through a secure channel such as physical contact so that the secrets will not disclose. The sensor needs to provide multiple CRPs and the PUF output with its ID as input to cu. Once initiated, the cu can communicate with the new node securely and also help the node to establish secrets with other sensors in the network.

• **Change of Node ID**

Very occasionally, a sensor in BAN may need to change its ID. When this event happens, the node can be first removed from the network then added back with the new ID. During this procedure, since the PUF function owned by the node will not change, its CRP pairs will also stay the same.

IV. Discussion

• **Security of the Approach**

Since the proposed approach tries to improve security of BAN, the safety of it must be carefully studied. An attacker may try several schemes to compromise the mechanism or to abuse it. Below we discuss several issues. First, it is very hard for an attacker to impersonate another node in the network because of the uniqueness of PUF function. Even if an attacker compromises multiple sensors and gets access to the CRP pairs stored at these nodes for a specific target, he will not be able to derive out new CRP pairs of the target. Therefore, the attacker cannot impersonate the node to other sensors. It is also very difficult for an attacker to impersonate the cu to BAN sensors. For each sensor s_i, it shares a secret $F_i(s_i)$ with the cu that is only used for communication between the two nodes. Even if an attacker compromises several BAN sensors, it cannot derive out the secrets between other sensors and the cu. Last but not least, a malicious party can launch a power drain attack upon a sensor. (While a similar attack on cu is also possible, the impacts of power drain attack on cu are very limited.) For example, an attacker can pretend to be cu and ask for some CRP from a sensor. To verify the identity of the requester, the sensor has to generate a random number and calculate several hash functions. While the attacker will fail to pass the verification, the power consumption at the sensor is real. Fortunately, if a sensor experiences repeated unsuccessful verification requests in a short period of time, it can report the anomaly to cu. Since BAN often has a relatively small size, it will not be hard to identify the attacker.

• **Impacts on System Efficiency**

Frequently, a security mechanism will introduce extra overhead into the system and bring negative impacts on

efficiency. Below we will discuss the storage and power consumption overhead of the approach. From the storage aspect, a sensor s_i needs to store only the secret $F_j(s_i)$ for each sensor s_j that it wants to communicate with directly. This will cause a very small overhead. For power consumption, the measurement results in [16], [17] have shown that the total power consumption for the generation, transmission, reception, and verification of a keyed hash based on a message will be about 223 μJ. As a comparison, the transmission of a 64-byte packet to the control unit will consume about 82 μJ energy. Since the pair-wise secret establishment procedure needs to run only once for each sensor pair, it will cause very limited power consumption.

Despite the limited overhead during secret establishment, the proposed approach has potential to reduce power consumption at sensors in the long run. Enabling device-to-device communication can eliminate intermediate transfer by the cu. Therefore, the number of transmissions will reduce drastically if such communication happens frequently in BAN networks. This would reduce collisions and interference among sensors, thus improving the overall system efficiency.

V. Conclusion

In this paper we propose an approach that helps BAN sensors to establish pair wise secrets through Physical Unclonable Functions with the help of control unit. These secrets enable sensors to communicate with each other directly with authenticated messages. We describe the mechanisms to establish the secrets and update the challenge-response pairs. It can support node dynamics in the network. Our analysis shows that the approach introduces very little overhead into the system and causes no new vulnerabilities.

Immediate extensions to our approach consist of the following aspects. First, we plan to implement the proposed approach on real personal medical devices and evaluate its performance in different scenarios. Second, we will continue to investigate the special properties of hardware in BAN networks to improve the system security.

References

[1] S. Movassaghi, M. Abolhasan, J. Lipman, D. Smith, and A. Jamalipour, "Wireless body area networks: A survey," *IEEE Communications Surveys Tutorials*, vol. 16, no. 3, pp. 1658–1686, Third 2014.

[2] D. Halperin and et. al., "Pacemakers and implantable cardiac defibrillators: Software radio attacks and zero-power defenses," in *IEEE Sympo. Security and Privacy*, 2008, pp. 129–142.

[3] M. Li, W. Lou, and K. Ren, "Data security and privacy in wireless body area networks," *IEEE Wireless Commun.*, vol. 17, no. 1, pp. 51–58, February 2010.

[4] G. E. Suh and S. Devadas, "Physical unclonable functions for device authentication and secret key generation," in *Proceedings of the 44th annual Design Automation Conference*, 2007, pp. 9–14.

[5] K. B. Frikken, M. Blanton, and M. J. Atallah, *Robust Authentication Using Physically Unclonable Functions*. Springer Berlin Heidelberg, 2009, pp. 262–277.

[6] M. N. Aman, K. C. Chua, and B. Sikdar, "Position paper: Physical unclonable functions for iot security," in *Proceedings of the ACM International Workshop on IoT Privacy, Trust, and Security*, 2016, pp. 10–13.

[7] L. Wei, C. Song, Y. Liu, J. Zhang, F. Yuan, and Q. Xu, "Boardpuf: Physical unclonable functions for printed circuit board authentication," in *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design*, 2015, pp. 152–158.

[8] M. S. Hashemian, B. Singh, F. Wolff, D. Weyer, S. Clay, and C. Papachristou, "A robust authentication methodology using physically unclonable functions in dram arrays," in *Proceedings of the Design, Automation & Test in Europe Conference & Exhibition*, 2015, pp. 647–652.

[9] P. Koeberl, J. Li, A. Rajan, C. Vishik, and W. Wu, "A practical device authentication scheme using sram pufs," in *Proceedings of the International Conference on Trust and Trustworthy Computing*, 2011, pp. 63–77.

[10] R. S. Pappu, "Physical one-way functions," Ph.D. dissertation, MIT, 2001.

[11] U. Kocabaş, A. Peter, S. Katzenbeisser, and A.-R. Sadeghi, "Converse puf-based authentication," in *Proceedings of the International Conference on Trust and Trustworthy Computing*, 2012, pp. 142–158.

[12] J. Delvaux, R. Peeters, D. Gu, and I. Verbauwhede, "A survey on lightweight entity authentication with strong pufs," *ACM Comput. Surv.*, vol. 48, no. 2, pp. 26:1–26:42, Oct. 2015.

[13] U. Rührmair, F. Sehnke, J. Sölter, G. Dror, S. Devadas, and J. Schmidhuber, "Modeling attacks on physical unclonable functions," in *Proceedings of the ACM Conference on Computer and Communications Security*, 2010, pp. 237–249.

[14] U. Rührmair and J. Sölter, "Puf modeling attacks: An introduction and overview," in *Proceedings of the Conference on Design, Automation & Test in Europe*, 2014, pp. 348:1–348:6.

[15] T. Xu, D. Li, and M. Potkonjak, "Adaptive characterization and emulation of delay-based physical unclonable functions using statistical models," in *Proceedings of the Annual Design Automation Conference*, 2015, pp. 76:1–76:6.

[16] E. Casilari, J. M. Cano-García, and G. Campos-Garrido, "Modeling of current consumption in 802.15.4/zigbee sensor motes," *Sensors*, vol. 10, no. 6, pp. 5443–5468, 2010.

[17] J. Lee, K. Kapitanova, and S. H. Son, "The price of security in wireless sensor networks," *Comput. Netw.*, vol. 54, no. 17, pp. 2967–2978, Dec 2010.