

# A Novel Authentication and Key Agreement Scheme for Implantable Medical Devices Deployment

Mohammad Wazid, *Student Member, IEEE*, Ashok Kumar Das<sup>ID</sup>, *Member, IEEE*,  
Neeraj Kumar<sup>ID</sup>, *Member, IEEE*, Mauro Conti<sup>ID</sup>, *Senior Member, IEEE*,  
and Athanasios V. Vasilakos, *Senior Member, IEEE*

**Abstract**—Implantable medical devices (IMDs) are man-made devices, which can be implanted in the human body to improve the functioning of various organs. The IMDs monitor and treat physiological condition of the human being (for example, monitoring of blood glucose level by insulin pump). The advancement of information and communication technology enhances the communication capabilities of IMDs. In healthcare applications, after mutual authentication, a user (for example, doctor) can access the health data from the IMDs implanted in a patient's body. However, in this kind of communication environment, there are always security and privacy issues, such as leakage of health data and malfunctioning of IMDs by an unauthorized access. To mitigate these issues, in this paper, we propose a new secure remote user authentication scheme for IMDs communication environment to overcome security and privacy issues in existing schemes. We provide the formal security verification using the widely accepted Automated Validation of Internet Security Protocols and Applications tool. We also provide the informal security analysis of the proposed scheme. The formal security verification and informal security analysis prove that the proposed scheme is secure against known attacks. The practical demonstration of the proposed scheme is performed using the broadly accepted NS2 simulation tool. The computation and communication costs of the proposed scheme are also comparable with the existing schemes. Moreover, the scheme provides additional functionality features, such as anonymity, untraceability, and dynamic implantable medical device addition.

**Index Terms**—Anonymity, AVISPA, implantable medical devices, key agreement, NS2 simulation, security, user authentication.

Manuscript received April 21, 2017; revised May 30, 2017; accepted June 25, 2017. Date of publication June 28, 2017; date of current version June 29, 2018. (Corresponding author: Ashok Kumar Das.)

M. Wazid and A. K. Das are with the Center for Security, Theory and Algorithmic Research, International Institute of Information Technology, Hyderabad 500032, India (e-mail: mohammad.wazid@research.iiit.ac.in; iitkgp.akdas@gmail.com).

N. Kumar is with the Department of Computer Science and Engineering, Thapar University, Patiala 147004, India (e-mail: neeraj.kumar@thapar.edu).

M. Conti is with the Department of Mathematics, University of Padua, Padua 35122, Italy (e-mail: conti@math.unipd.it).

A. V. Vasilakos is with the Department of Computer Science, Electrical and Space Engineering, Lulea University of Technology, Lulea 971 87, Sweden (e-mail: th.vasilakos@gmail.com).

Digital Object Identifier 10.1109/JBHI.2017.2721545

## I. INTRODUCTION

IMPLANTABLE medical devices (IMDs) monitor and treat physiological conditions within the body of a patient. Different types of IMDs such as brain neurostimulator, pacemaker, gastric implant and cochlear implant provide remote monitoring and treatment to patients with severe medical conditions. The pervasiveness of IMDs is growing continuously, for example, 25 million US citizens reliant on them for their day to day life critical functions [1]. The global IMDs market was valued at \$72,265 million in 2015, and is projected to reach \$116,300 million by 2022, registering a compound annual growth rate (CAGR) of 7.1% from 2016 to 2022 [2]. Information and communication technology (ICT) facilitates the information exchange of IMDs and provides them capabilities to communicate with each other. IMDs have the ability to send the collected health related data of a patient to the nearby controller node (CN) using the communication technologies such as bluetooth, zigbee and infrared transmission. CN is more powerful node as compared to IMDs as it has more communication range, processing power and storage capability. CN is connected to the Internet using an access point. A user (for example, a doctor) can access the data of an IMD via CN after successful mutual authentication. However, in such kind of communication environment, there are several security and privacy related issues such as replay attack, man-in-the middle attack, impersonation attacks and privileged-insider attack [3]–[6].

## A. Motivation

An attacker can exploit the vulnerabilities in the IMDs, which can cause negative medical effects on the health of the patient. Such effects are commonly known as *adverse events* [7]. According to the report available in [8], the vulnerability in an implanted insulin pump could be exploited by a hacker (a remote malicious user) which can cause an overdose of insulin to the diabetic patients. The overdose of insulin could then cause hypoglycemia (low blood sugar level) which in extreme case becomes a diabetic shock to the patient. Therefore, security of IMDs becomes a serious concern so that an illegal party can not attack the IMDs implanted in a patient's body. Hence, there is a strong need to design a secure remote user

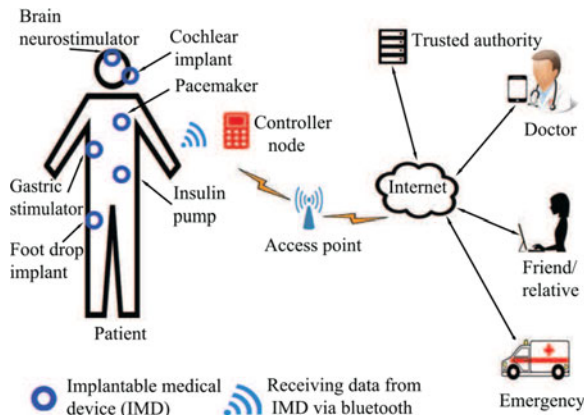


Fig. 1. Network model of *IMDs* communication environment.

authentication scheme for *IMDs* by which the controller node of a patient's *IMDs* and a user (for example, a doctor) can mutually authenticate each other. At the end, both entities establish a secret session key shared between them for their future secure communications. To address such an important issue for *IMDs* communication environment, we propose a new secure remote user authentication and key agreement scheme.

## B. Main Contributions

The contribution of this paper is manifold

- 1) We propose a new lightweight three-factor remote user authentication scheme for implantable medical devices in which the controller node of the implantable medical devices of a patient and remote user can authenticate each other.
- 2) The security analysis shows that the proposed scheme is secure. In addition, we test the formal security verification of the proposed scheme using the widely-accepted AVISPA tool to show the proposed scheme is also secure against the replay and man-in-the middle attacks.
- 3) We provide the practical implementation of the proposed scheme using the widely-used NS2 simulation tool to measure the impact of the scheme on network performance parameters such as end-to-end delay and throughput.

## C. System Models

The following two models are considered to describe and analyze the proposed scheme in the paper.

**1) Network Model:** The network model for the (*IMD*) s communication environment shown in Fig. 1 is used in the proposed scheme. In the given model, we have different types of *IMDs*, such as brain neurostimulator and gastric simulator, which are implanted in a patient's body. There is a controller node (*CN*) which collects data from all *IMDs* using wireless communication technologies (for example, bluetooth, zigbee and infrared transmission). *CN* is connected to the Internet through an access point. The users can access *IMDs* through *CN*. Suppose there is a user (for example, a doctor)  $U_i$  wants to access the data from the controller node belonging to a set of

implantable medical devices. In this scenario, we need authentication between  $U_i$  and *CN*.

**2) Threat Model:** The well-known Dolev-Yao threat model (DY model) [9] is used in the proposed scheme. Under the DY model, the communication takes place over insecure channels. Any two communicating parties can communicate each other using a public channel [10], in which the end-point entities, such as  $IMD_i$ , *CN*, and  $U_i$ , are not considered as trusted. An attacker  $\mathcal{A}$  can then have the opportunity to eavesdrop, modify or delete the exchanged messages during the transmission in order to tamper the communicated data.  $\mathcal{A}$  can also physically capture *CN* and can extract the stored information by using the power analysis attacks [11], [12] as these devices are non-tamper resistant. However, all *IMDs* are implanted inside the body of a patient, and hence, there is a rare possibility of physical capturing of *IMDs* from a patient's body. We further assume that the trusted authority (*TA*) is fully trusted party in the network, which is responsible for pre-deployment of *IMDs* and the user registration phase as described in Section III.

## D. Structure of the Paper

The rest of the paper is organized as follows. In Section II, we discuss the existing related authentication schemes proposed for *IMDs*. The various phases of the proposed scheme are discussed in Section III. The security analysis of the proposed scheme is provided in Section IV. The formal security verification of the proposed scheme using the widely-accepted AVISPA tool is given in Section V. The performance comparison of the related existing schemes and the proposed scheme is provided in Section VI. The practical demonstration of the proposed scheme using the widely-accepted NS2 simulation tool is also provided in Section VII. Finally, the paper is concluded in Section VIII.

## II. RELATED WORK

This section provides a brief review of the existing authentication schemes proposed for *IMD* communication environment.

An ultrasonic distance-bounding based scheme proposed by Rasmussen *et al.* [13] allows an *IMD* to give secure access to a programmer (reader) within a proximity range. The programmer has no constraint on power or computational ability. However their scheme did not provide non traceability property, session key security and also vulnerable to replay and man-in-the middle attacks. Ellouze *et al.*, [6] presented a scheme to secure cardiac *IMDs*. A Wireless Identification and Sensing Platform (WISP) is used in their scheme. They provided a solution to conserve the battery life of the *IMD* by harvesting energy using radio frequency signals from an UHF RFID reader to perform the key generation and authentication. Furthermore, they have also proposed an authentication mechanism using biometric keys for regular and emergency cases which facilitates a secure communication between the programmer and the WISP on the *IMD*. However their scheme did not provide anonymity and non traceability properties and also vulnerable to replay attack.

Jang *et al.* [3] provided a hybrid security scheme that uses two heterogeneous cryptosystems: symmetric and asymmetric.

The heterogeneous cryptosystems used to facilitate the different levels of security required by applications (for example, medical *versus* non-medical) in the wireless body area networks (WBANs). Their protocol contains two stages. In the first stage, the global authentication between bio-sensor node (BSN) and certificate authority (CA)/data server are performed, whereas in the second stage, the local authentication between BSN and base station (BS) is executed. However their scheme did not provide anonymity property and some functionality features such as dynamic controller node addition and *IMD* addition.

Several authentication schemes have been proposed in the literature for the healthcare applications using radio-frequency identification (RFID), wireless medical sensor networks and wireless body area networks [14]–[23]. He and Zeadally [4] proposed an authentication scheme by using the ambient intelligence, specifically for an Ambient Assisted Living (AAL) system that helps to monitor health and also to provide tele-health care services. Their system used the wearable sensors in the wireless body area networks (WBANs) and assistive robotics. The system has three levels of communications: 1) Intra-BAN: The wireless body sensors communicate with the WBAN controllers; 2) Inter-BAN: The WBAN controllers communicate with external devices (for example, home service robots) and 3) Beyond-BAN: The AAL server connects to the Internet.

Xu *et al.* [24] proposed a secure scheme for implantable cardiac devices, called IMDGuard. It provides two mechanisms for *IMDs* protection: the first one is an electrocardiogram sensor (ECG) based key establishment without prior shared secrets, and other one is an access control mechanism to protect spoofing attacks. Rushanan *et al.* [25] provided a survey of existing techniques, which improve security and privacy in *IMDs* and health BANs. A comprehensive survey of security and privacy issues in *IMDs* is also provided in [7]. Moreover, Denning *et al.* [26] discussed the human values and security issues associated with the *IMDs*.

### III. THE PROPOSED SCHEME

In this section, we present a new three-factor remote user authentication protocol for implantable medical devices communication environment, which uses the elliptic curve cryptography (ECC).

The network model presented in Fig. 1 is followed in the proposed scheme, in which there is a user (for example, a patient) whose body is implanted with implantable medical devices *IMDs*, such as pacemaker and insulin pump. All these *IMDs* monitor the patient's health. *IMDs* have their own functionalities and give services to the patient on the basis of his/her symptoms. *IMDs* also have wireless communication feature (for example, bluetooth technology) using which they can send the patient's monitored data to the nearby controller node, say  $CN_j$ .  $CN_j$  collects the sensed information securely from *IMDs*. Suppose there is a user (for example, a doctor)  $U_i$  wants to access the real-time data from a particular  $CN_j$  for monitoring and diagnosis of the patient remotely. In this scenario, we require authentication between  $U_i$  and  $CN_j$ . After mutual authentication between  $U_i$  and  $CN_j$ , they establish a session key for the

TABLE I  
NOTATIONS USED IN THIS PAPER

Notation	Description
$U_i, MD_i$	$i^{th}$ user and his/her mobile device
$CN_j$	$j^{th}$ controller node
$IMD_l$	$l^{th}$ implantable medical device
$TA$	Trusted authority
$ID_i, PW_i, BIO_i$	$U_i$ 's identity, password and biometric information
$ID_{TA}, ID_{CN_j}, RID_i, RID_{CN_j}$	Identities of trusted authority and controller node
$N$	1024-bit secret number of $TA$
$r_i, r_j$	160-bit random nonces of $U_i$ and $CN_j$
$RTS_{CN_j}$	Registration timestamp of $CN_j$
$\Delta t$	Generated current timestamp
$\Delta T$	Maximum transmission delay associated with a message
$Gen(\cdot)$	Probabilistic generation procedure used in fuzzy extractor
$Rep(\cdot)$	Deterministic reproduction procedure used in fuzzy extractor
$\sigma_i$	Biometric secret key of $U_i$
$\tau_i$	Public reproduction parameter of $U_i$
$t$	Error tolerance threshold used in fuzzy extractor
$E_p(a, b)$	A non-singular elliptic curve: $y^2 = x^3 + ax + b \pmod{p}$ over a prime finite field $Z_p$ (Galois field $GF(p)$ ) with $a, b \in Z_p^*$ are constants with $4a^3 + 27b^2 \neq 0 \pmod{p}$
$k.P$	Elliptic curve point multiplication; $k \in Z_p^*$ & $P \in E_p(a, b)$
$h(\cdot)$	Collision-resistant cryptographic hash function
$\ , \oplus$	Concatenation and bitwise XOR operations

future secure communication. After this successful mutual authentication only,  $U_i$  can access the live data from the implanted *IMDs* in the patient's body with the help of  $CN_j$ .

In this work, we use three factors: 1) mobile device  $MD_i$  of a user  $U_i$ ; 2) password  $PW_i$  of  $U_i$ ; and 3) biometrics  $BIO_i$  of  $U_i$ . The proposed scheme consists of the following seven phases: 1) pre-deployment; 2) offline user registration; 3) login; 4) authentication and key agreement; 5) password and biometric update; 6) dynamic controller node addition; and 7) dynamic *IMD* addition.

Table I contains the notations which are used for describing and analyzing the proposed scheme. We have used random nonces and current timestamps to protect against strong replay attack against an active adversary. For this purpose, we assume that all the network entities are synchronized with their clocks.

In the proposed scheme, we use the elliptic curve point multiplication operations. For better presentation of the paper, in the following we present the basic properties of an elliptic curve, and its two basic operations, such as point addition and point multiplication. A non-singular elliptic curve  $y^2 = x^3 + ax + b$  over a finite field  $GF(p)$  is denoted by the set  $E_p(a, b)$  consisting of the solutions  $(x, y) \in Z_p \times Z_p$  to the congruence  $y^2 \equiv x^3 + ax + b \pmod{p}$ . Here,  $a, b \in Z_p$  are constants with the condition  $4a^3 + 27b^2 \neq 0 \pmod{p}$ , together with a special point  $\mathcal{O}$ , called the point at infinity or zero point,  $Z_p = \{0, 1, \dots, p-1\}$  and  $p > 3$  be a large prime.  $E_p(a, b)$  forms an abelian group (commutative group) under addition modulo  $p$  operation [27] with the additive identity  $\mathcal{O}$  and the additive inverse  $-P \in E_p(a, b)$  of a point  $P \in E_p(a, b)$  such that if  $P = (x_P, y_P)$ , we have  $-P = (x_P, -y_P)$ , where  $x_P$  and  $y_P$  denote the  $x$  and  $y$  coordinates of the point  $P \in E_p(a, b)$ ,



respectively. If we consider  $P = (x_P, y_P)$  and  $Q = (x_Q, y_Q)$  as two points on an elliptic curve  $E_p(a, b)$ ,  $R = (x_R, y_R) = P + Q \in E_p(a, b)$  is computed as follows [27]:  $x_R = (\lambda^2 - x_P - x_Q) \pmod{p}$ ,  $y_R = (\lambda(x_P - x_R) - y_P) \pmod{p}$ , where

$$\lambda = \begin{cases} \frac{y_Q - y_P}{x_Q - x_P} \pmod{p}, & \text{if } P \neq Q \\ \frac{3x_P^2 + a}{2y_P} \pmod{p}, & \text{if } P = Q. \end{cases}$$

In elliptic curve cryptography (ECC), point multiplication (scalar multiplication) is defined as the repeated point additions. For example, if  $P \in E_p(a, b)$ ,  $5P$  is computed as  $5P = P + P + P + P + P$ .

Given a scalar  $k \in \mathbb{Z}_p$  and a point  $P \in E_p(a, b)$ , computing the scalar multiplication  $Q = k.P$  is relatively easy. However, given  $P$  and  $Q$  in  $E_p(a, b)$ , it is computationally infeasible to compute the scalar  $k \in \mathbb{Z}_p$ , where  $Q = k.P$ . This problem is called the elliptic curve discrete logarithm problem (ECDLP).

For biometric authentication, we use the fuzzy extractor technique [28], [29]. A fuzzy extractor consists of the following two procedures: 1) probabilistic generation function  $Gen(\cdot)$  and 2) deterministic reproduction function  $Rep(\cdot)$ . Upon input as a user personal biometrics  $BIO_i$ ,  $Gen(\cdot)$  produces output consisting of a secret biometric key of fixed length, say  $\eta$  bits,  $\sigma_i \in \{0, 1\}^\eta$  and a public reproduction parameter  $\tau_i$ . On the other hand,  $Rep(\cdot)$  takes the current biometrics entered by the user, say  $BIO'_i$  and also the public reproduction parameter  $\tau_i$  as input, provided the Hamming distance between  $BIO'_i$  and  $BIO_i$  is less than or equal to an error tolerance threshold value,  $t$ . The output of  $Rep(\cdot)$  is then the original biometric key  $\sigma_i$ , that is,  $\sigma_i = Rep(BIO'_i, \tau_i)$ .

### A. Pre-Deployment Phase

In this phase, a trusted authority (TA) is responsible for registering each controller node  $CN_j$  and each implantable medical device  $IMD_l$  prior to their deployment in a deployment field (for example, a hospital) and the patient's body. For this purpose, the TA first selects a unique 1024-bit secret number  $N$  for each  $CN_j$  and the  $IMDs$  attached with  $CN_j$ , and computes its pseudo identity using its own identity  $ID_{TA}$  as  $RID_{TA} = h(ID_{TA} || N)$ . The TA then chooses a unique identity  $ID_{CN_j}$  for each  $CN_j$ , and calculates its corresponding pseudo identity  $RID_{CN_j} = h(ID_{CN_j} || N)$  and the temporary credential of  $CN_j$  using its registration timestamp  $RTS_{CN_j}$  as  $TC_{CN_j} = h(ID_{TA} || RTS_{CN_j} || N)$ . The TA finally stores the information  $\{RID_{CN_j}, TC_{CN_j}, RID_{TA}\}$  in the memory of  $CN_j$  and deploys it in the deployment field.

For the pairwise key establishment between a deployed  $CN_j$  and  $IMD_l$ s in a patient's body, we use the existing polynomial-based key distribution protocol proposed by Blundo *et al.* [30]. For each  $CN_j$ , the TA first selects a unique symmetric bivariate polynomial  $\mathcal{P}(x, y) = \sum_{i=0}^n \sum_{j=0}^n g_{i,j} x^i y^j \in GF(p)[x, y]$  of degree  $n$  over a finite field (Galois field)  $GF(p)$ , where the co-efficients  $g_{i,j}$ 's are taken from  $GF(p)$ . Note that the prime  $p$  is chosen as a large number and  $n$  is also large, which is much larger than the number of  $IMDs$  deployed in a patient's

body attached with  $CN_j$  in order to preserve unconditional security and  $n$ -collusion resistant property against  $IMD$  capture attack by an attacker [31]. For example, if a bivariate polynomial  $\mathcal{P}(x, y) = x^4 + 3x^3 + 2x^2y^2 + 3y^3 + y^4$  over  $GF(5)$  is symmetric as  $\mathcal{P}(y, x) = y^4 + 3y^3 + 2y^2x^2 + 3x^3 + x^4 = \mathcal{P}(x, y)$ .

For each deployed  $IMD_l$ , the TA generates a unique identity  $ID_{IMD_l}$ , computes the corresponding pseudo identity  $RID_{IMD_l} = h(ID_{IMD_l} || N)$  and the polynomial share  $\mathcal{P}(RID_{IMD_l}, y)$  which is a univariate polynomial of degree  $n$  in  $GF(p)$ , and stores this polynomial share and the pseudo identity  $RID_{IMD_l}$  in the memory of  $IMD_l$ . Note that to store  $\mathcal{P}(RID_{IMD_l}, y)$ , the storage space required in  $IMD_l$  is  $(n+1)\log_2(p)$  bits as the coefficients are from  $GF(p)$ . In a similar way, for  $CN_j$  the TA also computes the polynomial share  $\mathcal{P}(RID_{CN_j}, y)$  which is a univariate polynomial of degree  $n$  in  $GF(p)$ , and stores this polynomial share in the memory of  $CN_j$ . Finally, the information  $\{RID_{CN_j}, TC_{CN_j}, RID_{TA}, \mathcal{P}(RID_{CN_j}, y)\}$  are stored in  $CN_j$ 's memory.

The motivation behind the use of the Blundo *et al.*'s scheme [30] for pairwise key establishment between a deployed  $CN_j$  and  $IMD_l$ s in a patient's body is as follows. If an adversary  $\mathcal{A}$  is able to compromise  $(n+1)$  or more shares of  $\mathcal{P}(x, y)$ , he/she can easily reconstruct the original  $\mathcal{P}(x, y)$  uniquely using *Lagrange's interpolation* [32]. Hence, the disclosure of up to  $n$  shares does not reveal  $\mathcal{P}(x, y)$  to  $\mathcal{A}$ , and thus, non-compromised shared keys based on  $\mathcal{P}(x, y)$  remain completely secure. Since the degree  $n$  of  $\mathcal{P}(x, y)$  is much larger than the number of  $IMDs$  deployed in a patient's body attached with  $CN_j$ , the proposed scheme preserves unconditional security and  $n$ -collusion resistant property [31], [33].

### B. Post-Deployment Phase

Once the  $IMDs$  and  $CN_j$  are deployed, the first task of  $CN_j$  and  $IMDs$  is to establish pairwise secret keys using the pre-loaded information stored in their memory during the pre-deployment phase described in Section III-A.

Suppose deployed  $IMD_l$  and  $CN_j$  want to establish a pairwise secret key between them.  $IMD_l$  first sends its pseudo identity  $RID_{IMD_l}$  to  $CN_j$ . In a similar manner,  $CN_j$  also sends its pseudo identity  $RID_{CN_j}$  to  $IMD_l$ . After that  $IMD_l$  computes the secret key shared with  $CN_j$  using its own polynomial share as  $SK_{IMD_l, CN_j} = \mathcal{P}(RID_{IMD_l}, RID_{CN_j})$ . On the other hand,  $CN_j$  also computes the same secret key shared with  $IMD_l$  using its own polynomial share as  $SK_{CN_j, IMD_l} = \mathcal{P}(RID_{CN_j}, RID_{IMD_l}) = \mathcal{P}(RID_{IMD_l}, RID_{CN_j}) (= SK_{IMD_l, CN_j})$  since the polynomial  $\mathcal{P}(x, y)$  is symmetric. Hence, both  $IMD_l$  and  $CN_j$  will communicate securely in order to bring the information sensed by the  $IMD_l$  to  $CN_j$  using the established shared key  $SK_{IMD_l, CN_j}$ .

### C. User Registration Phase

This phase discusses the registration procedure for a user (for example, a doctor)  $U_i$  to access the information from the controller node  $CN_j$  of a patient's implantable medical devices

IMDs. For this purpose,  $U_i$  requires to register at the  $TA$  securely either in person or via a secure channel. This procedure is performed by the  $TA$  and  $U_i$  with the following steps:

**Step REG1.**  $U_i$  selects an identity  $ID_i$  and sends it to the  $TA$  securely. After receiving registration request, the  $TA$  computes pseudo identity of  $U_i$  as  $RID_i = h(ID_i || N)$  using its corresponding secret number  $N$  and  $A_i = h(RID_{TA} || ID_i)$ , and sends the registration reply message  $\langle RID_i, A_i, RID_{TA} \rangle$  to  $U_i$  securely.

**Step REG2.** After receiving registration reply from the  $TA$ ,  $U_i$  chooses a non-singular elliptic curve  $E_p(a, b): y^2 = x^3 + ax + b \pmod{p}$  over a prime finite field  $Z_p$ , where  $p$  is a large prime and  $a, b \in Z_p^*$  are constants such that  $4a^3 + 27b^2 \neq 0 \pmod{p}$ .  $U_i$  further chooses a base point  $P$  of order  $m$  over  $E_p(a, b)$  such that  $m \cdot P = \mathcal{O}$ , where  $\mathcal{O}$  is called the point at infinity or zero point.  $U_i$  then selects a private key  $k$  and computes the corresponding public key  $Q = k \cdot P$ , and makes  $Q$  as public.

**Step REG3.**  $U_i$  selects a password  $PW_i$  on his/her choice, and inputs his/her biometric  $BIO_i$  at the sensor of his/her mobile device, say  $MD_i$ .  $MD_i$  applies the fuzzy extractor probabilistic generation function  $Gen(\cdot)$  to generate the secret biometric key  $\sigma_i$  and the corresponding public parameter  $\tau_i$  as  $Gen(BIO_i) = (\sigma_i, \tau_i)$  as provided in [29], [34], [35]. The detailed information about the fuzzy extractor functions  $Gen(\cdot)$  and  $Rep(\cdot)$  can be found in [29].

**Step REG4.**  $MD_i$  calculates  $RID'_i = RID_i \oplus h(PW_i || \sigma_i)$ , masked password  $RPW_i = h(PW_i || k)$ ,  $D_i = k \oplus h(ID_i || PW_i || \sigma_i)$ ,  $RID'_{TA} = RID_{TA} \oplus h(ID_i || k || \sigma_i)$  and  $A'_i = A_i \oplus h(k || \sigma_i)$ . After these computations,  $MD_i$  also computes the parameters  $B_i = h(A_i || RPW_i)$  and  $C_i = h(ID_i || RID_{TA} || B_i || \sigma_i)$ . Finally,  $MD_i$  stores the information  $\{RID'_i, RID'_{TA}, A'_i, C_i, D_i, \tau_i, Gen(\cdot), Rep(\cdot), h(\cdot), t\}$  in its memory, where  $t$  is the error tolerance threshold value used in  $Rep(\cdot)$  in order to recover the original biometric key  $\sigma_i$ .

## D. Login Phase

$U_i$  performs the following steps to execute the login phase:

**Step L1.**  $U_i$  inputs his/her identity  $ID_i$  and password  $PW_i$  into the interface of  $MD_i$ , and also imprints his/her biometrics  $BIO'_i$  at the sensor of  $MD_i$ .  $MD_i$  then extracts biometric key  $\sigma'_i = Rep(BIO'_i, \tau_i)$  provided that the Hamming distance between the original biometrics  $BIO_i$  at the time of registration and the recent entered  $BIO'_i$  is less than the error tolerance threshold value  $t$ . Then,  $MD_i$  computes  $k' = D_i \oplus h(ID_i || PW_i || \sigma'_i)$ ,  $RPW'_i = h(PW_i || k')$ ,  $A^*_i = A'_i \oplus h(k' || \sigma'_i)$ ,  $B^*_i = h(A^*_i || RPW'_i)$ ,  $RID^*_{TA} = RID'_{TA} \oplus h(ID_i || k' || \sigma'_i)$  and  $RID^*_i = RID'_i \oplus h(PW_i || \sigma'_i)$  and  $C^*_i = h(ID_i || RID^*_{TA} || B^*_i || \sigma'_i)$ . After computing these values,  $MD_i$  checks whether the condition  $C^*_i = C_i$  holds or not. If it holds,  $U_i$  passes both password and biometric verification. Otherwise, the login process is terminated immediately.

**Step L2.**  $MD_i$  generates the current timestamp  $T_1$  and 160-bit random nonce  $r_i$ .  $MD_i$  then computes  $a_i = h(r_i || T_1 || RID^*_i || RPW'_i || \sigma'_i)$ ,  $b_i = h(RID^*_{TA} || T_1)$ ,  $M_1 = a_i \cdot P$  and the ElGamal type signature  $M_2 = a_i + k' \cdot b_i \pmod{p}$ .

Finally,  $MD_i$  sends the login request message  $\langle M_1, M_2, T_1 \rangle$  to  $CN_j$  via a public channel.

## E. Authentication and Key Agreement Phase

After receiving the login request  $\langle M_1, M_2, T_1 \rangle$  from  $U_i$  at time  $T_1^*$  by  $CN_j$ , the following steps are executed for mutual authentication and key establishment between  $U_i$  and  $CN_j$ :

**Step AKE1.**  $CN_j$  first checks the timeliness of  $T_1$  by the condition  $|T_1 - T_1^*| < \Delta T$ , where  $\Delta T$  is the maximum transmission delay. If timeliness matches,  $CN_j$  computes  $b'_i = h(RID_{TA} || T_1)$  and verifies the signature by the condition  $M_2 \cdot P = M_1 + b'_i \cdot Q$ . Note that  $M_2 \cdot P = (a_i + k \cdot b_i) \cdot P = a_i \cdot P + k \cdot b_i \cdot P = M_1 + b_i \cdot Q = M_1 + b'_i \cdot Q$ . If verification matches,  $CN_j$  chooses current timestamp  $T_2$  and 160-bit random nonce  $r_j$ , and computes  $c_j = h(r_j || T_2 || RID_{CN_j} || TC_{CN_j})$ ,  $M_4 = c_j \cdot P$ ,  $k_{ij} = c_j \cdot M_1 = (a_i c_j) \cdot P$ . After these computations,  $CN_j$  computes the session key  $SK_{ij} = h(k_{ij} || RID_{TA} || T_1 || T_2)$  shared with  $U_i$  and  $M_5 = h(SK_{ij} || T_2)$ . Then,  $CN_j$  sends the authentication reply  $\langle M_4, M_5, T_2 \rangle$  to  $U_i$  via a public channel.

**Step AKE2.** After receiving the authentication reply  $\langle M_4, M_5, T_2 \rangle$  from  $CN_j$  at time  $T_2^*$ ,  $U_i$  checks the timeliness of  $T_2$  by the verification condition  $|T_2 - T_2^*| < \Delta T$ . If it does not hold,  $U_i$  immediately terminates the connection. Otherwise,  $U_i$  computes  $k^*_{ij} = a_i \cdot M_4 = (a_i c_j) \cdot P$  and session key  $SK^*_{ij} = h(k^*_{ij} || RID^*_{TA} || T_1 || T_2)$  shared with  $U_i$ , and  $M_6 = h(SK^*_{ij} || T_2)$ .  $U_i$  then checks whether the condition  $M_6 = M_5$  holds. If it is so,  $CN_j$  is authenticated by  $U_i$ . Furthermore,  $U_i$  generates the current timestamp  $T_3$  and computes  $M_7 = h(SK^*_{ij} || T_3)$  and sends acknowledgment message  $\langle M_7, T_3 \rangle$  to  $CN_j$  via an open channel.

**Step AKE3.** After receiving the message  $\langle M_7, T_3 \rangle$  from  $U_i$  at time  $T_3^*$ ,  $CN_j$  checks the timeliness of  $T_3$  by the condition  $|T_3 - T_3^*| < \Delta T$ . If this condition holds,  $CN_j$  calculates  $M_8 = h(SK_{ij} || T_3)$  and checks if  $M_8 = M_7$  holds. If it does not match, it immediately terminates the connection. Otherwise, it is considered that the calculated session key  $SK^*_{ij}$  by  $U_i$  is correct, and both  $U_i$  and  $CN_j$  stores the same session key  $SK_{ij} = (SK^*_{ij})$  for future secure communication.

The login, and authentication and key agreement phases of the proposed scheme are summarized in Fig. 2.

## F. Password and Biometric Update Phase

In this phase, we provide the password & biometric update facility in which a legitimate user  $U_i$  can change his/her password as well as biometrics at any time without involving the  $TA$  for security reasons. The following steps are required for this phase:

**Step PBI.**  $U_i$  inputs his/her identity  $ID_i$ , old password  $PW_i^{old}$  to the interface of  $MD_i$ , and also imprints his/her old biometrics  $BIO_i^{old}$  to the sensor of  $MD_i$ .  $MD_i$  then extracts biometric key  $\sigma_i^{old} = Rep(BIO_i^{old}, \tau_i)$  provided that the Hamming distance between the original biometrics  $BIO_i$  at the time of registration and the entered  $BIO_i^{old}$  is less than the error tolerance threshold value  $t$ . In addition,  $MD_i$  calculates  $k = D_i \oplus h(ID_i || PW_i^{old} || \sigma_i^{old})$ ,  $RPW_i^{old} = h(PW_i^{old} || k)$ ,

User ( $U_i$ )/Mobile device ( $MD_i$ )	Controller node ( $CN_j$ )
Input $ID_i, PW_i, BIO_i$ . Extract $\sigma'_i = \text{Rep}(BIO_i, \tau_i)$ . Compute $k' = D_i \oplus h(ID_i    PW_i    \sigma'_i)$ , $RPW'_i = h(PW_i    k')$ , $A_i^* = A'_i \oplus h(k'    \sigma'_i)$ , $B_i^* = h(A_i^*    RPW'_i)$ , $RID_{TA}^* = RID_{TA}' \oplus h(ID_i    k'    \sigma'_i)$ , $RID_i^* = RID_i' \oplus h(PW_i    \sigma'_i)$ , $C_i^* = h(ID_i    RID_{TA}^*    B_i^*    \sigma'_i)$ . Check if $C_i^* = C_i$ ? if so, choose $T_1, r_i$ . Compute $a_i = h(r_i    T_1    RID_i^*    RPW'_i    \sigma'_i)$ , $b_i = h(RID_{TA}^*    T_1)$ , $M_1 = a_i.P$ , $M_2 = a_i + k' \cdot b_i \pmod{p}$ . $\langle M_1, M_2, T_1 \rangle$ (via open channel)	Check $ T_1 - T_3^*  < \Delta T$ , if so compute $b'_i = h(RID_{TA}    T_1)$ . Verify if $M_2.P = M_1 + b'_i.Q$ ? If holds, choose $T_2$ and $r_j$ . Compute $c_j = h(r_j    T_2    RID_{CN_j}    TC_{CN_j})$ , $M_4 = c_j.P$ , $k_{ij} = c_j.M_1 = (a_i c_j).P$ , $SK_{ij} = h(k_{ij}    RID_{TA}    T_1    T_2)$ , $M_5 = h(SK_{ij}    T_2)$ . $\langle M_4, M_5, T_2 \rangle$ (via open channel)
Check if $ T_2 - T_2^*  < \Delta T$ ? Compute $k_{ij}^* = a_i.M_4 = (a_i c_j).P$ , $SK_{ij}^* = h(k_{ij}^*    RID_{TA}    T_1    T_2)$ , $M_6 = h(SK_{ij}^*    T_2)$ . Check if $M_6 = M_5$ ? If so, choose $T_3$ . Compute $M_7 = h(SK_{ij}^*    T_3)$ . $\langle M_7, T_3 \rangle$ (via open channel)	Check if $ T_3 - T_3^*  < \Delta T$ ? Compute $M_8 = h(SK_{ij}    T_3)$ . Check if $M_8 = M_7$ ?
Both $U_i$ and $CN_j$ store session key $SK_{ij} (= SK_{ij}^*)$ .	

Fig. 2. Summary of login, and authentication and key agreement phases.

$A_i^{old} = A'_i \oplus h(k || \sigma_i^{old})$ ,  $B_i^{old} = h(A_i^{old} || RPW_i^{old})$ ,  $RID_{TA} = RID_{TA}' \oplus h(ID_i || k || \sigma_i^{old})$ ,  $RID_i = RID_i' \oplus h(PW_i^{old} || \sigma_i^{old})$  and  $C_i^{old} = h(ID_i || RID_{TA} || B_i^{old} || \sigma_i^{old})$ . After computing these values,  $MD_i$  checks the condition  $C_i^{old} = C_i$ . If it holds,  $U_i$  is treated as an actual user who passes both password and biometric verification, and he/she can proceed for the password and biometric update procedure. Otherwise, the password and biometric update process is terminated immediately.

**Step PB2.**  $U_i$  provides a new password  $PW_i^{new}$ , and also imprints new biometrics  $BIO_i^{new}$ , if  $U_i$  is desired to change  $BIO_i^{old}$ . It is also noted that if  $U_i$  does not want to change his/her biometrics, he/she still can keep the same old biometrics  $BIO_i^{old}$ , and in this situation,  $BIO_i^{new}$  is considered as  $BIO_i^{old}$ . After these inputs,  $MD_i$  computes  $\sigma_i^{new} = \text{Rep}(BIO_i^{new}, \tau_i^{new})$ ,  $RPW_i^{new} = h(PW_i^{new} || k)$ ,  $A_i^{new} = A_i^{old} \oplus h(k || \sigma_i^{new})$ ,  $B_i^{new} = h(A_i^{new} || RPW_i^{new})$ ,  $RID_{TA}'' = RID_{TA} \oplus h(ID_i || k || \sigma_i^{new})$ ,  $RID_i'' = RID_i \oplus h(PW_i^{new} || \sigma_i^{new})$ ,  $C_i^{new} = h(ID_i || RID_{TA}'' || B_i^{new} || \sigma_i^{new})$  and  $D_i'' = k \oplus h(ID_i || PW_i^{new} || \sigma_i^{new})$ .

**Step PB3.** Finally,  $MD_i$  replaces  $RID_i'$ ,  $RID_{TA}'$ ,  $A_i'$ ,  $C_i$ ,  $D_i$  and  $\tau_i$  with  $RID_i''$ ,  $RID_{TA}''$ ,  $A_i^{new}$ ,  $C_i^{new}$ ,  $D_i''$  and  $\tau_i^{new}$ , respectively.

We summarize the password and biometric update phase related to the proposed scheme in Fig. 3.

### G. Dynamic Controller Node Addition Phase

This phase is required to deploy a new controller scheme, say  $CN_j^{new}$  in the existing network. The  $TA$  performs the following steps for the dynamic controller node addition:

**Step 1.** The  $TA$  first assigns a new unique identity  $ID_{CN_j}^{new}$ , which is different from the identities of the already deployed controller nodes. The  $TA$  then computes the pseudo identity

User ( $U_i$ )	Mobile device ( $MD_i$ )
Input identity $ID_i$ , old password $PW_i^{old}$ , and imprint biometrics $BIO_i^{old}$ .	Extract $\sigma_i^{old} = \text{Rep}(BIO_i^{old}, \tau_i)$ . Calculate $k = D_i \oplus h(ID_i    PW_i^{old}    \sigma_i^{old})$ , $RPW_i^{old} = h(PW_i^{old}    k)$ , $A_i^{old} = A'_i \oplus h(k    \sigma_i^{old})$ , $B_i^{old} = h(A_i^{old}    RPW_i^{old})$ , $RID_{TA} = RID_{TA}' \oplus h(ID_i    k    \sigma_i^{old})$ , $RID_i = RID_i' \oplus h(PW_i^{old}    \sigma_i^{old})$ , $C_i^{old} = h(ID_i    RID_{TA}    B_i^{old}    \sigma_i^{old})$ . Check if $C_i^{old} = C_i$ ? If so, ask $U_i$ to enter new password and imprint new biometrics.
Input new password $PW_i^{new}$ . Imprint new $BIO_i^{new}$ .	Calculate $\sigma_i^{new} = \text{Rep}(BIO_i^{new}, \tau_i^{new})$ , $RPW_i^{new} = h(PW_i^{new}    k)$ , $A_i^{new} = A_i^{old} \oplus h(k    \sigma_i^{new})$ , $B_i^{new} = h(A_i^{new}    RPW_i^{new})$ , $RID_{TA}'' = RID_{TA} \oplus h(ID_i    k    \sigma_i^{new})$ , $RID_i'' = RID_i \oplus h(PW_i^{new}    \sigma_i^{new})$ , $C_i^{new} = h(ID_i    RID_{TA}''    B_i^{new}    \sigma_i^{new})$ , $D_i'' = k \oplus h(ID_i    PW_i^{new}    \sigma_i^{new})$ . Replace $RID_i'$ , $RID_{TA}'$ , $A_i'$ , $C_i$ , $D_i$ and $\tau_i$ with $RID_i''$ , $RID_{TA}''$ , $A_i^{new}$ , $C_i^{new}$ , $D_i''$ and $\tau_i^{new}$ , respectively.

Fig. 3. Summary of password and biometric update phase.

for  $CN_j^{new}$  as  $RID_{CN_j}^{new} = h(ID_{CN_j}^{new} || N)$  and  $TC_{CN_j}^{new} = h(ID_{TA} || RTS_{CN_j}^{new} || N)$ , where  $RTS_{CN_j}^{new}$  is newly generated registration timestamp for  $CN_j^{new}$ .  $TA$  also computes polynomial share  $\mathcal{P}(RID_{CN_j}^{new}, y)$ , which is a univariate polynomial in  $GF(p)$ .

**Step 2.** Finally, the  $TA$  stores the credentials  $\{RID_{CN_j}^{new}, TC_{CN_j}^{new}, RID_{TA}, \mathcal{P}(RID_{CN_j}^{new}, y)\}$  into the memory of  $CN_j^{new}$  prior to its deployment.

### H. Dynamic IMD Addition Phase

To deploy a new  $IMD$  or to replace an existing  $IMD$  by another new  $IMD$ , say  $IMD'_i$ , the  $TA$  executes the following steps:

**Step 1.** The  $TA$  generates a unique identity  $ID'_{IMD_i}$ , and computes the corresponding pseudo identity  $RID'_{IMD_i} = h(ID'_{IMD_i} || N)$  and the polynomial share  $\mathcal{P}(RID'_{IMD_i}, y)$ .

**Step 2.** The  $TA$  then stores  $\mathcal{P}(RID'_{IMD_i}, y)$  and  $RID'_{IMD_i}$  in the memory of  $IMD'_i$ .

Note that there is no need to update any polynomial share in  $CN_j$ . The  $TA$  only needs to inform  $CN_j$  about the deployment of  $IMD'_i$ . After deployment of  $IMD'_i$ , it can establish pairwise key with  $CN_j$  as  $SK_{IMD'_i, CN_j} = \mathcal{P}(RID_{IMD'_i}, RID_{CN_j}) = \mathcal{P}(RID_{CN_j}, RID_{IMD'_i})$  and start secure communication using the established key  $SK_{IMD'_i, CN_j}$  with the help of the post-deployment phase given in Section III-B.

## IV. SECURITY ANALYSIS

In this section, we show that the proposed scheme is secure against the following possible known attacks:

**Replay attack:** In the proposed scheme, during the login, and authentication and key agreement phases, the messages  $Msg_1 = \langle M_1, M_2, T_1 \rangle$ ,  $Msg_2 = \langle M_4, M_5, T_2 \rangle$  and  $Msg_3 = \langle M_7, T_3 \rangle$



are exchanged between a user  $U_i$  and a controller node  $CN_j$ . These messages involve different current timestamps  $T_1$ ,  $T_2$  and  $T_3$ . If an adversary  $\mathcal{A}$  intercepts these messages and tries to replay these messages later, the validity of timestamps in these messages will fail, and as a result, the messages will be treated as the old messages. Hence, our scheme provides protection against replay attack.

**Man-in-the-middle attack:** Suppose  $\mathcal{A}$  intercepts the message  $Msg_1$  and attempts to modify this message to create a valid login message. For creating the valid login message,  $\mathcal{A}$  can generate a random nonce  $r_{ia}$  and current timestamp  $T_{1a}$ . Then,  $\mathcal{A}$  is not able to compute  $M'_1 = a_{ia}.P$ ,  $a_{ia} = h(r_{ia}||T_{1a}||RID_i^*||RPW_i'|\sigma_i)$  because  $\mathcal{A}$  does not know the values of  $RID_i^*$ ,  $RPW_i'$  and the biometric secret key  $\sigma_i$  of the user  $U_i$ . Similarly, if  $\mathcal{A}$  tries to compute the signature  $M'_2 = a_{ia} + k.b'_i \pmod{p}$ , he/she needs  $a_{ia}$  and  $b'_i = h(RID_{TA}'||T_{1a})$ .  $\mathcal{A}$  can not also compute  $M'_2$  because he/she does not know  $RID_{TA}'$  and  $k$ , which is the secret key of  $U_i$ . Therefore,  $\mathcal{A}$  is not able to modify  $Msg_1$ . In a similar way,  $\mathcal{A}$  can not modify other messages  $Msg_2$  and  $Msg_3$ . Therefore, our scheme provides protection against man-in-the-middle attack.

**Privileged-insider and offline password guessing attacks:** A privileged user of the  $TA$ , who may be an internal adversary  $\mathcal{A}$ , can obtain  $RID_i$  during the user  $U_i$ 's registration phase. In addition, suppose the mobile device  $MD_i$  of  $U_i$  is lost or stolen by  $\mathcal{A}$  after the registration process is finished.  $MD_i$  contains information  $\{RID_i', RID_{TA}', A_i', C_i, D_i, \tau_i, Gen(\cdot), Rep(\cdot), h(\cdot), t\}$ . Even by retrieving all stored information from  $MD_i$  using the power analysis attacks [11], [12],  $\mathcal{A}$  can not guess the correct password  $PW_i$  because he/she does not know the private key  $k$  of  $U_i$  and his/her secret biometric key  $\sigma_i$  from  $C_i$  and  $D_i$ . Therefore, the correct guess of  $PW_i$  will not be successful by  $\mathcal{A}$ . Hence, the proposed scheme is secure against both privileged-insider and offline password guessing attacks.

**User impersonation attack:** Suppose  $\mathcal{A}$  intercepts the message  $Msg_1 = \langle M_1, M_2, T_1 \rangle$  during the login phase, and tries to impersonate as a legal user  $U_i$  by sending a valid login request message to  $CN_j$ .  $\mathcal{A}$  can not compute the secret  $a_i$  from  $M_1 = a_i.P$  due to hardness of the ECDLP problem (discussed in Section III). In order to perform user impersonation attack,  $\mathcal{A}$  can generate the current timestamp  $T'_1$  and a random nonce  $r_{ia}$ . To generate valid login request message, say  $Msg'_1 = \langle M'_1, M'_2, T'_1 \rangle$ ,  $\mathcal{A}$  requires to compute  $M'_1 = a_{ia}.P$  and  $M'_2 = a_{ia} + k.b_{ia} \pmod{p}$ , where  $a_{ia} = h(r_{ia}||T'_1||RID_i^*||RPW_i'|\sigma_i)$  and  $b_{ia} = h(RID_{TA}'||T'_1)$ .  $\mathcal{A}$  can not compute  $Msg'_1$  as he/she does not know  $RID_i^*$ ,  $RID_{TA}'$ ,  $RPW_i'$ ,  $\sigma_i$  and the secret key  $k$  of  $U_i$ . Therefore, the user impersonation attack is protected in our scheme.

**Controller node impersonation attack:** Suppose  $\mathcal{A}$  intercepts the message  $Msg_2 = \langle M_4, M_5, T_2 \rangle$  during the authentication and key establishment phase, and tries to impersonate as a controller node  $CN_j$  by sending a valid authentication reply message to  $U_i$ .  $\mathcal{A}$  can not compute the secret  $c_j$  from  $M_4 = c_j.P$  due to hardness of the ECDLP problem.  $\mathcal{A}$  can generate the current timestamp  $T'_2$ , and random nonces  $r_{ia}$  and  $r_{ja}$ . To generate a valid message, say  $Msg'_2 = \langle M'_4, M'_5, T'_2 \rangle$ ,  $\mathcal{A}$  needs to compute  $M'_4 = c_{ja}.P$ ,  $M'_5 = h(SK'_{ij}||T'_2)$ ,

where  $c_{ja} = h(r_{ja}||T'_2||RID_{CN_j}||TC_{CN_j})$ ,  $k'_{ij} = c_{ja}.M'_1 = (a_{ia}.c_{ja}).P$ ,  $a_{ia} = h(r_{ia}||T_1||RID_i^*||RPW_i'|\sigma_i)$ , and  $SK'_{ij} = h(k'_{ij}||RID_{TA}'||T_1||T'_2)$ .  $\mathcal{A}$  cannot compute  $Msg'_2$  as he/she does not know  $RID_{CN_j}$ ,  $TC_{CN_j}$ ,  $RID_i^*$ ,  $RID_{TA}'$ ,  $RPW_i'$  and  $\sigma_i$ . Thus, our scheme is secure against such an attack.

**Session key security:** During the login and authentication & session key agreement phases,  $U_i$  sends the message  $Msg_1 = \langle M_1, M_2, T_1 \rangle$  to  $CN_j$ . Then  $CN_j$  replies to  $U_i$  with the message  $Msg_2 = \langle M_4, M_5, T_2 \rangle$ . Further,  $U_i$  sends the acknowledgment message  $Msg_3 = \langle M_7, T_3 \rangle$  to  $CN_j$ . In all these messages session key  $SK_{ij} (= SK_{ij}^*)$  is protected by the one-way hash function  $h(\cdot)$ . Moreover, without the knowledge of short term secrets such as random nonces  $r_i$  and  $r_j$ , and long term secrets, such as identities  $RID_{TA}$ ,  $RID_i$ ,  $RID_{CN_j}$  and  $TC_{CN_j}$ ,  $\mathcal{A}$  can not compute session key  $SK_{ij}$ . Therefore, due to the use of these short term and long term secrets, and also the collision resistance property of  $h(\cdot)$ , the computation of  $SK_{ij}$  is computationally infeasible for  $\mathcal{A}$ . As a result, the proposed scheme provides session key security.

**Anonymity and untraceability:** Suppose an adversary  $\mathcal{A}$  intercepts the messages  $Msg_1 = \langle M_1, M_2, T_1 \rangle$ ,  $Msg_2 = \langle M_4, M_5, T_2 \rangle$  and  $Msg_3 = \langle M_7, T_3 \rangle$  during the login and authentication & key agreement phases. Due to usage of random nonces  $r_i$ ,  $r_j$  and current timestamps, each of  $a_i$ ,  $b_i$ ,  $c_j$  and  $k_{ij}$  becomes dynamic and "unique" in all messages for each session. Moreover, none of these messages directly includes  $ID_i$  and  $ID_{CN_j}$ . Hence, the proposed scheme preserves both anonymity and untraceability properties.

**Resilience against controller node physical capture attack:** As in [36], [37], the resilience against controller node physical capture attack of the proposed scheme in the IMD communication environment is as follows. Assume that  $c$  controller nodes are physically captured by an adversary  $\mathcal{A}$ . It is then measured as the total secure communications compromised by a capture of  $c$  controller nodes *not including* the communication in which the compromised controller nodes are directly involved. Let  $P_e(c)$  denote the probability that  $\mathcal{A}$  can decrypt the secure communication between a user  $U_i$  and a non-compromised controller node  $CN_j$  when  $c$  controller nodes are already compromised. If  $P_e(c) = 0$ , a user authentication scheme is known as unconditionally secure against controller node capture attack. By physically capturing a controller node  $CN'_j$ ,  $\mathcal{A}$  can extract the information  $\{RID_{CN_j}, TC_{CN_j}, RID_{TA}, \mathcal{P}(RID_{CN_j}, y)\}$  from its memory using power analysis attacks [11], [12]. Note that all  $RID_{CN_j}$ ,  $TC_{CN_j}$  and  $\mathcal{P}(RID_{CN_j}, y)$  are distinct for all the controller nodes, and these are generated by the  $TA$ . Therefore, by capturing  $CN'_j$ ,  $\mathcal{A}$  can only compromise the session key between that the user  $U_i$  and  $CN'_j$ . However, the session keys between that user  $U_i$  and other non-compromised controller nodes  $CN_j$  s can not be compromised by  $\mathcal{A}$ . Then, compromise of a controller node does not lead to compromise secure communications among the user and other non-compromised controller nodes. Hence, our scheme is unconditionally secure against controller node physical capture attack.

**Denial-of-service attack (DoS):** Even if a legal user  $U_i$  enters incorrect  $ID_i$  and/or  $PW_i$  during login phase, it is locally checked through the verification  $C_i^* = C_i$  (Step L1 in

Section III-D). The login request of the user  $U_i$  is sent to the controller node only after successful verification. As a result, the proposed scheme is secure against such DoS attack.

*Stolen mobile device attack:* Suppose the mobile device  $MD_i$  of a legal user  $U_i$  is lost or stolen by an attacker  $\mathcal{A}$ .  $\mathcal{A}$  can then extract all information  $\{RID'_i, RID'_{TA}, A'_i, C_i, D_i, \tau_i, Gen(\cdot), Rep(\cdot), h(\cdot), t\}$  stored in  $MD_i$  using the power analysis attacks. To correctly guess  $ID_i$  and  $PW_i$  from the extracted information  $C_i$  and  $D_i$ ,  $\mathcal{A}$  needs to know both the secrets  $k$  and  $\sigma_i$ . Thus, it is computationally infeasible for  $\mathcal{A}$  to correctly guess both  $ID_i$  and  $PW_i$ . Therefore, the proposed scheme is secure against stolen mobile device attack.

## V. FORMAL SECURITY VERIFICATION USING AVISPA

In this section, we provide the formal security verification of the proposed scheme using the widely-accepted AVISPA tool [38], [39].

In AVISPA, we first implement the security protocol in the role-based expressive formal language, called the High Level Protocol Specification Language (HLPSP). HLPSP is translated into the intermediate format (IF) using the translator, called HLPSP2IF. IF is a lower-level language than HLPSP and is read directly by the back-ends to the AVISPA tool. There are four backends in AVISPA tool: 1) On-the-fly Model-Checker (OFMC); 2) Constraint-Logic-based Attack Searcher (CL-AtSe); 3) SAT-based Model-Checker (SATMC) and 4) Tree Automata based on Automatic Approximations for the Analysis of Security Protocols (TA4SP). Finally, the backends produce the output format (OF), which precisely tells whether the protocol is safe or unsafe. If it is unsafe, the OF also lists the attack trace. In AVISPA, the communication channel is public and it is modeled using the Dolev-Yao threat model [9]. Thus, the intruder (which is always denoted by  $i$  in HLPSP) can also take a legitimate role in the protocol run. The detailed description of the AVISPA tool and the HLPSP is available in [38], [39].

In HLPSP, each entity in the network (user  $U_i$ , the  $TA$  and controller node  $CN_j$ ) is implemented in a role. Apart from these basic roles, we have other two mandatory roles, called session, and goal and environment. Each role contains global constants and a composition of one or more sessions, where the intruder may play some roles as legitimate user. For the replay attack checking, OFMC checks whether the legitimate agents can execute the specified protocol by performing a search of a passive intruder. For the Dolev-Yao model check, this backend also checks whether there is any man-in-the-middle attack possible by the intruder. We have simulated the proposed scheme using the Security Protocol ANimator for AVISPA (SPAN) [40] for the OFMC and CL-AtSe back-ends since these backends supports bitwise XOR operation. The simulation results of the analysis provided in Fig. 4 show that the proposed scheme is secure against replay and man-in-the-middle attacks.

## VI. COMPARATIVE STUDY

In this section, we compare the computation and communication costs, and functionality features of the proposed scheme with other related existing schemes, such as the schemes of

OFMC	SUMMARY
Version of 2006/02/13	SAFE
SUMMARY	DETAILS
SAFE	BOUNDED NUMBER OF SESSIONS
DETAILS	TYPED MODEL
BOUNDED NUMBER OF SESSIONS	PROTOCOL
PROTOCOL	C: progra 1 SPAN testsuite
C: progra 1 SPAN testsuite	results auth imd.if
results auth imd.if	GOAL
GOAL	As Specified
as specified	BACKEND
BACKEND	CL AtSe
OFMC	STATISTICS
COMMENTS	
STATISTICS	
parseTime: 0.00s	Analysed : 1111 states
searchTime: 21.09s	Reachable : 1108 states
visitedNodes: 315 nodes	Translation: 0.13 seconds
depth: 12 plies	Computation: 8.69 seconds

Fig. 4. The result of the analysis using OFMC and CL-AtSe backends.

TABLE II  
COMPARISON OF FUNCTIONALITY FEATURES

Feature	Rasmussen <i>et al.</i>	Jang <i>et al.</i>	Ellouze <i>et al.</i>	He- Zeaddally	Our
$FN F_1$	✓	✓	✓	✓	✓
$FN F_2$	✓	×	×	✓	✓
$FN F_3$	×	✓	×	✓	✓
$FN F_4$	✓	✓	✓	✓	✓
$FN F_5$	×	✓	✓	✓	✓
$FN F_6$	✓	✓	✓	✓	✓
$FN F_7$	✓	✓	✓	✓	✓
$FN F_8$	×	✓	×	✓	✓
$FN F_9$	×	✓	✓	✓	✓
$FN F_{10}$	N/A	N/A	N/A	✓	✓
$FN F_{11}$	N/A	N/A	N/A	N/A	✓
$FN F_{12}$	N/A	N/A	N/A	N/A	✓
$FN F_{13}$	N/A	×	N/A	×	✓
$FN F_{14}$	×	×	×	×	✓
$FN F_{15}$	✓	N/A	×	✓	✓
$FN F_{16}$	✓	✓	✓	✓	✓
$FN F_{17}$	×	×	×	×	✓

Note:  $FN F_1$ : mutual authentication;  $FN F_2$ : anonymity;  $FN F_3$ : non-traceability;  $FN F_4$ : session-key agreement;  $FN F_5$ : session key security;  $FN F_6$ : confidentiality;  $FN F_7$ : integrity;  $FN F_8$ : strong replay attack;  $FN F_9$ : man-in-the-middle attack;  $FN F_{10}$ : efficient login phase;  $FN F_{11}$ : password update phase;  $FN F_{12}$ : biometric update phase;  $FN F_{13}$ : dynamic controller node addition;  $FN F_{14}$ : dynamic  $IMD$  addition;  $FN F_{15}$ : protection against stolen mobile device/programmer attack;  $FN F_{16}$ : protection against impersonation attack;  $FN F_{17}$ : formal security verification using AVISPA tool.

×: a scheme is insecure against a particular attack or does not support a particular feature; ✓: a scheme is secure against a particular attack or supports a particular feature; N/A: not applicable in a scheme.

Rasmussen *et al.* [13], Ellouze *et al.* [6], Jang *et al.* [3] and He-Zeaddally [4].

The comparison of functionality features of the existing schemes and our scheme is given in Table II. It is evident from the table that Rasmussen *et al.*'s scheme does not provide  $FN F_3$ ,  $FN F_5$ ,  $FN F_8$ ,  $FN F_9$ ,  $FN F_{14}$  and  $FN F_{17}$ ; Jang *et al.*'s scheme does not provide the features  $FN F_2$ ,  $FN F_{13}$ ,  $FN F_{14}$  and  $FN F_{17}$ ; Ellouze *et al.*'s scheme does not provide  $FN F_2$ ,  $FN F_3$ ,  $FN F_8$ ,  $FN F_{14}$ ,  $FN F_{15}$  and  $FN F_{17}$ ; and He-Zeaddally's scheme does not also provide  $FN F_{13}$ ,  $FN F_{14}$  and  $FN F_{17}$ . On the other hand, the proposed scheme provides all the functionality features listed in the table.



TABLE III

APPROXIMATE TIME REQUIRED FOR VARIOUS OPERATIONS [41]

Notation	Description (time to compute)	Approx. computation time (seconds)
$T_h$	One-way hash function	0.00032
$T_{ecm}$	ECC point multiplication	0.0171
$T_{eca}$	ECC point addition	0.0044
$T_{senc}$	Symmetric encryption	0.0056
$T_{sdec}$	Symmetric decryption	0.0056
$T_{me}$	Modular exponentiation	0.0192
$T_{fe} \approx T_{ecm}$ [41]	Fuzzy extractor function	0.0171

TABLE IV

COMPUTATION COSTS COMPARISON

Scheme	Computation cost
Rasmussen <i>et al.</i>	$2T_{me} + 1T_h \approx 0.03872s$
Jang <i>et al.</i>	$25T_{ecm} + 15T_{eca} + 5T_h \approx 0.4951s$
Ellouze <i>et al.</i>	$6T_h + 2T_{senc}/T_{sdec} \approx 0.01312s$
He-Zeadally	$6T_{ecm} + 8T_{senc}/T_{sdec} + 4T_h \approx 0.1487s$
Proposed scheme	$T_{fe} + 6T_{ecm} + 17T_h \approx 0.12514s$

For computation costs comparison, we have listed the approximate time needed for various cryptographic operations in Table III. We use the existing experimental results for these operations [41]. The comparison of computation costs of existing related schemes [6], [13] (for regular mode), [3] (for global authentication), [4] and the proposed scheme is given in Table IV. Though the computation cost of our scheme is more than that for the schemes of Rasmussen *et al.* and Ellouze *et al.*, it can be considered as our scheme provides more security and functionality features as compared to those schemes.

For communication costs comparison, we have taken the timestamp, sequence number or random nonce is of 32 bits each. If the SHA-1 [42] hash function is used, the size of hash digest is 160 bits. All identities are assumed to be 160 bits each. We further assume that the 1024-bit Diffie-Hellman key is used in Rasmussen *et al.*'s protocol, since an 160-bit ECC cryptosystem provides the same security as 1024-bit RSA cryptosystem [43]. Thus, each elliptic curve point of the form  $P = (x_P, y_P)$  requires  $(160 + 160) = 320$  bits. The public key cryptosystem used in Jang *et al.*'s [3] hybrid protocol is considered as ECC. Therefore, in that case ECC encryption of a plaintext, which is an ECC point  $P_m$ , using the public key produces the ciphertext  $(C_1, C_2)$ , where both  $C_1$  and  $C_2$  are ECC points and the ciphertext requires  $(320 + 320) = 640$  bits. Additionally, symmetric encryption/decryption is of 128 bits (if we apply the Advanced Encryption Standard (AES) [44]). Table V shows the comparison of communication costs of the related existing schemes [6], [13] (for regular mode), [3] (for global authentication), [4] and our scheme in terms of the number of messages and number of bits. The results in this table show that though the communication cost of Ellouze *et al.*'s scheme is less than our scheme, but it can be accepted as our scheme provides more security and more functionality features as compared to other schemes.

TABLE V

COMMUNICATION OVERHEADS COMPARISON

Scheme	No. of messages	No. of bits
Rasmussen <i>et al.</i>	6	2210
Jang <i>et al.</i>	8	5920
Ellouze <i>et al.</i>	3	961
He-Zeadally	4	3232
Proposed scheme	3	1216

TABLE VI

VARIOUS SIMULATION PARAMETERS

Parameter	Description
Platform	Ubuntu 14.04 LTS
Network scenarios	1, 2 and 3
Number of users	3,5,9 for scenarios 1,2,3
Number of controller nodes	3 for all scenarios
Number of implantable medical devices	15 for all scenarios
Simulation time	1800 seconds

## VII. PRACTICAL PERSPECTIVE: NS2 SIMULATION STUDY

In this section, to measure the impact of the proposed scheme on the network performance parameters, such as end-to-end delay (in seconds) and throughput (in bits per second), we have used widely accepted NS2 2.35 simulator [45], [46] on Ubuntu 14.04 LTS platform.

### A. Simulation Parameters

The parameters used in the NS2 simulation are given in Table VI. The network coverage area is taken as  $80 \times 80 m^2$ . The communication ranges of implantable medical devices and controller nodes are taken as 25 meters and 50 meters, respectively. The medium access control type is the standard IEEE 802.15.4 and the network was simulated for the duration of 1800 seconds (30 minutes). Apart from these, all other standard parameters are considered for the simulation.

### B. Simulation Environment

We have considered the following three network scenarios, in which there are three controller nodes  $CN_j$ s and three patients implanted with five  $IMD_j$ s each. Hence, there are a total of 15  $IMD_j$ s are deployed in the simulation.

- 1) *Scenario 1.* In this scenario, there are three users ( $U_i$ s), three controller nodes ( $CN_j$ s) and 15  $IMD_j$ s.
- 2) *Scenario 2.* Under this scenario, we have taken five users ( $U_i$ s), three controller nodes ( $CN_j$ s) and 15  $IMD_j$ s.
- 3) *Scenario 3.* Here, there are nine users ( $U_i$ s), three controller nodes ( $CN_j$ s) and 15  $IMD_j$ s.

In each scenario, we have considered the three messages:  $\{M_1, M_2, T_1\}$  from  $U_i$  to  $CN_j$ ,  $\{M_4, M_5, T_2\}$  from  $CN_j$  to  $U_i$ , and  $\{M_7, T_3\}$  from  $U_i$  to  $CN_j$ , which are of sizes 512 bits, 512 bits, 192 bits, respectively.

### C. Simulation Results and Discussions

In order to measure the impact of the proposed scheme, we have calculated the network performance parameters, such as end-to-end delay and throughput.

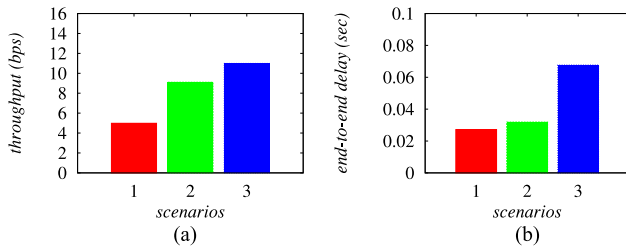


Fig. 5. Network performance: (a) throughput and (b) end-to-end delay.

1) **Impact on End-to-end Delay:** The end-to-end delay ( $EED$ ) is formulated as the average time taken by the data packets (messages) to arrive at the destination from the source. The  $EED$  is then calculated as  $\sum_{i=1}^{n_{pkt}} (T_{rec_i} - T_{send_i}) / n_{pkt}$ , where  $T_{rec_i}$  and  $T_{send_i}$  are the receiving and sending time of a packet  $i$ , respectively, and  $n_{pkt}$  is the total number of packets. The  $EED$ s of the proposed scheme for different scenarios are provided in Fig. 5(a). The values of  $EED$ s are 0.02719, 0.03188 and 0.06765 seconds for the scenarios 1, 2 and 3, respectively. Note that the value of  $EED$  increases with the increasing number of users. The increment in number of users results more number of exchanged messages, which further incurs congestion, and therefore,  $EED$  increases in scenarios 2 and 3.

2) **Impact on Throughput:** The throughput is measured as the number of bits transmitted per unit time. The network throughput (in bps) of the proposed scheme under different network scenarios is provided in Fig. 5(b). The throughput is formulated as  $\frac{n_r \times |pkt|}{T_d}$ , where  $T_d$  is the total time (in seconds),  $|pkt|$  the size of a packet and  $n_r$  the total number of received packets. Note that the simulation time as 1800 s, which is considered as the total time. The throughput values are 4.98, 9.10 and 10.99 bps for the scenarios 1, 2 and 3, respectively. The throughput also increases in scenarios 2 and 3.

## VIII. CONCLUSION

The use of  $IMDs$  facilitates the remote monitoring of the health of a patient. The  $IMDs$  specially improve the quality of life of elderly people, who other has problem to move easily. A doctor can provide them remote consultation on the basis of their health data, which is collected by the help of  $IMDs$ . However, wireless communication raises serious threats in the  $IMD$  deployment. In this paper, we proposed a remote user authentication scheme through which a user (a doctor) and a controller node can mutually authenticate each other and establish a session key for their future secure communication. Apart from that the pairwise key establishment between a controller node and its  $IMDs$  is also provided in the proposed scheme for the secure communication between them. The computation and communication costs of the proposed scheme are comparable with the existing related schemes. In addition, the proposed scheme also provides better security and more functionality features, such as password and biometric update phase, dynamic controller node and  $IMD$  addition phases, as compared to other existing related schemes.

## ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers and the Associate Editor for their valuable feedback on this paper which helped them to improve its quality and presentation. Mauro Conti is supported by a Marie Curie Fellowship funded by the European Commission (agreement PCIG11-GA-2012-321980). This work is also partially supported by the EU TagItSmart! Project (agreement H2020-ICT30-2015-688061), the EU-India REACH Project (agreement ICI+/2014/342-896), and by the projects “Physical-Layer Security for Wireless Communication”, and “Content Centric Networking: Security and Privacy Issues” funded by the University of Padua. This work is partially supported by the grant n. 2017-166478 (3696) from Cisco University Research Program Fund and Silicon Valley Community Foundation. This work is also partially funded by the project CNR-MOST/Taiwan 2016-17 “Verifiable Data Structure Streaming”.

## REFERENCES

- [1] D. Halperin, T. S. Heydt-Benjamin, K. Fu, T. Kohno, and W. H. Malsel, “Security and privacy for implantable medical devices,” *IEEE Pervasive Comput.*, vol. 7, no. 1, pp. 30–39, Jan.–Mar. 2008.
- [2] R. Thakur, “Implantable medical devices market is expected to reach \$116,300 million by 2022, globally—Allied market research.” [Online]. Available: <http://www.prnewswire.com/news-releases/implantable-medical-devices-market-is-expected-to-reach-116300-million-by-2022-globally—allied-market-research-6-13835833.html>. Accessed on Feb. 2017.
- [3] C. S. Jang, D. G. Lee, J.-W. Han, and J. H. Park, “Hybrid security protocol for wireless body area networks,” *Wireless Commun. Mobile Comput.*, vol. 11, no. 2, pp. 277–288, 2011.
- [4] D. He and S. Zeadally, “Authentication protocol for an ambient assisted living system,” *IEEE Commun. Mag.*, vol. 53, no. 1, pp. 71–77, Jan. 2015.
- [5] K. B. Rasmussen, C. Castelluccia, T. S. Heydt-Benjamin, and S. Capkun, “Proximity-based access control for implantable medical devices,” in *Proc. 16th ACM Conf. Comput. Commun. Security*, Chicago, IL, USA, 2009, pp. 410–419.
- [6] N. Ellouze, M. Allouche, H. Ben Ahmed, S. Rekhiss, and N. Boudriga, “Securing implantable cardiac medical devices: Use of radio frequency energy harvesting,” in *Proc. 3rd Int. Workshop Trustworthy Embedded Devices*, Berlin, Germany, 2013, pp. 35–42.
- [7] C. Camara, P. P. Lopez, and J. E. Tapiador, “Security and privacy issues in implantable medical devices: A comprehensive survey,” *J. Biomed. Informat.*, vol. 55, pp. 272–289, 2015.
- [8] J. Finkle, “J & J warns diabetic patients: Insulin pump vulnerable to hacking.” [Online]. Available: <http://www.reuters.com/article/us-johnson-johnson-cyber-insulin-pumps-e-idUSKCN12411L>. Accessed on Feb. 2017.
- [9] D. Dolev and A. C. Yao, “On the security of public key protocols,” *IEEE Trans. Inf. Theory*, vol. 29, no. 2, pp. 198–208, Mar. 1983.
- [10] M. L. Das, “Two-factor user authentication in wireless sensor networks,” *IEEE Trans. Wireless Commun.*, vol. 8, no. 3, pp. 1086–1090, Mar. 2009.
- [11] T. S. Messerges, E. A. Dabbish, and R. H. Sloan, “Examining smart-card security under the threat of power analysis attacks,” *IEEE Trans. Comput.*, vol. 51, no. 5, pp. 541–552, May 2002.
- [12] P. Kocher, J. Jaffe, and B. Jun, “Differential power analysis,” in *Proc. 19th Annu. Int. Cryptol. Conf.*, Santa Barbara, CA, USA, 1999, vol. 1666, pp. 388–397.
- [13] K. B. Rasmussen, C. Castelluccia, T. S. Heydt-Benjamin, and S. Capkun, “Proximity-based access control for implantable medical devices,” in *Proc. 16th ACM Conf. Comput. Commun. Security*, Chicago, IL, USA, 2009, pp. 410–419.
- [14] D. He and S. Zeadally, “An analysis of RFID authentication schemes for internet of things in healthcare environment using elliptic curve cryptography,” *IEEE Internet Things J.*, vol. 2, no. 1, pp. 72–83, Feb. 2015.
- [15] D. He, N. Kumar, J. Chen, C. C. Lee, N. Chilamkurti, and S. S. Yeo, “Robust anonymous authentication protocol for health-care applications using wireless medical sensor networks,” *Multimedia Syst.*, vol. 21, no. 1, pp. 49–60, 2015.

- [16] D. He, S. Zeadally, N. Kumar, and J. H. Lee, "Anonymous authentication for wireless body area networks with provable security," *IEEE Syst. J.*, to be published, DOI: [10.1109/jsyst.2016.2544805](https://doi.org/10.1109/jsyst.2016.2544805).
- [17] X. Li, J. Niu, S. Kumari, F. Wu, and K. K. R. Choo, "A robust biometrics based three-factor authentication scheme for global mobility networks in smart city," *Future Gener. Comput. Syst.*, to be published, DOI: [10.1016/j.future.2017.04.012](https://doi.org/10.1016/j.future.2017.04.012).
- [18] X. Li, M. H. Ibrahim, S. Kumari, A. K. Sangaiah, V. Gupta, and K. K. R. Choo, "Anonymous mutual authentication and key agreement scheme for wearable sensors in wireless body area networks," *Comput. Netw.*, to be published, DOI: [10.1016/j.comnet.2017.03.013](https://doi.org/10.1016/j.comnet.2017.03.013).
- [19] X. Li, J. Peng, S. Kumari, F. Wu, M. Karuppiah, and K. K. R. Choo, "An enhanced 1-round authentication protocol for wireless body area networks with user anonymity," *Comput. Elect. Eng.*, to be published, DOI: [10.1016/j.compeleceng.2017.02.011](https://doi.org/10.1016/j.compeleceng.2017.02.011).
- [20] X. Li, J. Niu, M. K. Khan, J. Liao, and X. Zhao, "Robust three-factor remote user authentication scheme with key agreement for multimedia systems," *Security Commun. Netw.*, vol. 9, no. 13, pp. 1916–1927, 2016.
- [21] D. He, C. Chen, S. Chan, J. Bu, and A. V. Vasilakos, "ReTrust: Attack-resistant and lightweight trust management for medical sensor networks," *IEEE Trans. Inf. Technol. Biomed.*, vol. 16, no. 4, pp. 623–632, Jul. 2012.
- [22] Z. Zhang, H. Wang, A. V. Vasilakos, and H. Fang, "ECG-cryptography and authentication in body area networks," *IEEE Trans. Inf. Technol. Biomed.*, vol. 16, no. 6, pp. 1070–1078, Nov. 2012.
- [23] J. Zhou, Z. Cao, X. Dong, N. Xiong, and A. V. Vasilakos, "4S: A secure and privacy-preserving key management scheme for cloud-assisted wireless body area network in m-healthcare social networks," *Inf. Sci.*, vol. 314, pp. 255–276, 2015.
- [24] F. Xu, Z. Qin, C. C. Tan, B. Wang, and Q. Li, "IMDGuard: Securing implantable medical devices with the external wearable guardian," in *Proc. IEEE Conf. Comput. Commun.*, Shanghai, China, 2011, pp. 1862–1870.
- [25] M. Rushanan, A. D. Rubin, D. F. Kune, and C. M. Swanson, "SoK: Security and privacy in implantable medical devices and body area networks," in *Proc. IEEE Symp. Security Privacy*, San Jose, CA, USA, 2014, pp. 524–539.
- [26] T. Denning, A. Borning, B. Friedman, B. T. Gill, T. Kohno, and W. H. Maisel, "Patients, pacemakers, and implantable defibrillators: Human values and security for wireless implantable medical devices," in *SIGCHI Conference on Human Factors in Computing Systems*. Atlanta, Georgia, USA: ACM, 2010, pp. 917–926.
- [27] N. Kobitz, A. Menezes, and S. A. Vanstone, "The state of elliptic curve cryptography," *Des., Codes Cryptography*, vol. 19, no. 2/3, pp. 173–193, 2000.
- [28] Y. Dodis, L. Reyzin, and A. Smith, "Fuzzy extractors: How to generate strong keys from biometrics and other noisy data," in *Advances in Cryptology-Eurocrypt 2004*. Interlaken, Switzerland: Springer, 2004, pp. 523–540.
- [29] V. Odelu, A. K. Das, and A. Goswami, "A secure biometrics-based multi-server authentication protocol using smart cards," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 9, pp. 1953–1966, Sep. 2015.
- [30] C. Blundo, A. D. Santis, A. Herzberg, S. Kutten, U. Vaccaro, and M. Yung, "Perfectly-secure key distribution for dynamic conferences," in *Proc. 12th Annu. Int. Cryptology Conf.*, Santa Barbara, CA, USA, 1993, vol. 740, pp. 471–486.
- [31] A. K. Das and I. Sengupta, "An effective group-based key establishment scheme for large-scale wireless sensor networks using bivariate polynomials," in *Proc. 3rd IEEE Int. Conf. Commun. Syst. Softw. Middleware*, Bangalore, India, 2008, pp. 9–16.
- [32] H. Wang and Y. Zhang, "Cryptanalysis of an efficient threshold self-healing key distribution scheme," *IEEE Trans. Wireless Commun.*, vol. 10, no. 1, pp. 1–4, Jan. 2011.
- [33] D. Liu, P. Ning, and R. Li, "Establishing pairwise keys in distributed sensor networks," *ACM Trans. Inf. Syst. Security*, vol. 8, no. 1, pp. 41–77, 2005.
- [34] A. K. Das, "A secure and robust temporal credential-based three-factor user authentication scheme for wireless sensor networks," *Peer-to-Peer Netw. Appl.*, vol. 9, no. 1, pp. 223–244, 2016.
- [35] V. Odelu, A. K. Das, and A. Goswami, "SEAP: Secure and efficient authentication protocol for NFC applications using pseudonyms," *IEEE Trans. Consum. Electron.*, vol. 62, no. 1, pp. 30–38, Feb. 2016.
- [36] A. K. Das, P. Sharma, S. Chatterjee, and J. K. Sing, "A dynamic password-based user authentication scheme for hierarchical wireless sensor networks," *J. Netw. Comput. Appl.*, vol. 35, no. 5, pp. 1646–1656, 2012.
- [37] S. Kumari *et al.*, "On the design of a secure user authentication and key agreement scheme for wireless sensor networks," *Concurrency Comput., Pract. Experience*, to be published, DOI: [10.1002/cpe.3930](https://doi.org/10.1002/cpe.3930).
- [38] AVISPA, "Automated validation of internet security protocols and applications." [Online]. Available: <http://www.avispa-project.org/>. Accessed on Apr. 2016.
- [39] A. Armando *et al.*, "The AVISPA tool for the automated validation of internet security protocols and applications," in *17th International Conference on Computer Aided Verification* (Lecture Notes in Computer Science). New York, NY, USA: Springer-Verlag, 2005, vol. 3576, pp. 281–285.
- [40] AVISPA, "SPAN, the Security Protocol ANimator for AVISPA." [Online]. Available: <http://www.avispa-project.org>. Accessed on Apr. 2017.
- [41] D. He, N. Kumar, J. H. Lee, and R. S. Sherratt, "Enhanced three-factor security protocol for consumer USB mass storage devices," *IEEE Trans. Consum. Electron.*, vol. 60, no. 1, pp. 30–37, Feb. 2014.
- [42] *Secure Hash Standard*, FIPS PUB 180-1, Natl. Inst. Stand. Technol., U.S. Dept. Commerce, Apr. 1995. [Online]. Available: <http://csrc.nist.gov/publications/fips/fips180-4/fips-180-4.pdf>. Accessed on Sep. 2015.
- [43] S. Vanstone, "Responses to NIST's proposal," *Commun. ACM*, vol. 35, no. 7, pp. 50–52, 1992.
- [44] *Advanced Encryption Standard (AES)*, FIPS PUB 197, Natl. Inst. Stand. Technol., U.S. Dept. Commerce, Nov. 2001. [Online]. Available: <http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf>. Accessed on Apr. 2016.
- [45] "The Network Simulator-ns-2." Jan. 2017. [Online]. Available: <http://www.isi.edu/nsnam/ns/>
- [46] J. Wang, "NS-2 tutorial." Apr. 2016. [Online]. Available: <http://www.cs.virginia.edu/c&#x303;s757/slidespdf/cs757-ns2-tutorial1.pdf>