Lightweight authentication protocol using self-certified public keys for wireless body area networks in health-care applications

EVANGELINA LARA¹, LEOCUNDO AGUILAR², AND JESÚS A. GARCÍA.³

¹ Universidad Autónoma de Baja California, Calzada Universidad #14418, Parque Industrial Internacional, Tijuana B.C. 22390, Mexico
² Universidad Autónoma de Baja California, Calzada Universidad #14418, Parque Industrial Internacional, Tijuana B.C. 22390, Mexico (e-mail: laguilar@uabc.edu.mx)
³ Universidad Autónoma de Baja California, Calzada Universidad #14418, Parque Industrial Internacional, Tijuana B.C. 22390, Mexico (e-mail: garcia.jesus@uabc.edu.mx)

Corresponding author: Evangelina Lara (e-mail: evangelina.lara@uabc.edu.mx). ORCID of the corresponding author: https://orcid.org/0000-0002-1935-2319

This work was supported in part by CONACYT under Grant No. 536467 and No. 383573.

ABSTRACT Health-care centers have been meeting challenges due to the increase in the aging population and chronic diseases that need continuous medical monitoring. Wireless body area network (WBAN) is a non-invasive technology consisting of diverse connected bio-medical sensors placed in the human body, which measure physiological parameters and make the information accessible to health-care professionals ubiquitously. However, a major problem in WBAN is the security and privacy of the patient’s medical information. An essential security method to protect the physiological data is authentication. Several authentication protocols have been proposed for WBANs; however, some require many computing resources, and some have security vulnerabilities. In this article, the Two-Party Lightweight Authentication Protocol (TLAP) for WBANs is proposed. It uses self-certified public keys based on Elliptic Curve Cryptography (ECC), scalar point multiplication, symmetric key encryption, and the lightweight operations xor and conventional hash function to reduce the computational cost of the protocol. Formal and informal analyses were made to demonstrate that TLAP provides mutual authentication and resists potential attacks in WBANs. The security and performance of TLAP and similar existing protocols were analyzed and compared. The analyzes showed TLAP supports more security features and has lower execution time and communication cost than the other protocols, which is significant to decrease the energy consumption in WBANs.

INDEX TERMS Authentication, Healthcare, Internet of Things, Lightweight, M2M, Self-certified, TMIS, WBAN

I. INTRODUCTION

ONE of the current challenges humanity is facing is health-care. The world population is rapidly growing, but the number of healthcare facilities does not increase in proportion to the population size. Further, the aging population is also fast-growing, it has been predicted that by 2050 the population over 60 years old in America will be about 80 million, and in China, about 430 million [1]. Moreover, sedentary lifestyles and unhealthy diets have caused an increment in various chronic diseases that need continuous medical monitoring to control them and avoid risks to the patient’s life. Many fatal diseases can be controlled if they are detected at their initial stages [2]. Additionally, early disease diagnostics can help to reduce the cost of health-care systems [3]. Therefore, it is necessary to develop efficient, intelligent, accurate, proactive, and affordable systems for early risk detection and continuous health-condition monitoring, to help to decrease the pressure in health-care facilities [4]. A technology that can be used in medical-related services is wireless body area networks (WBANs), which can greatly improve the monitoring and delivery of health information. A WBAN is a wireless communication...
network consisting of diverse bio-medical sensors, portable personal terminals such as personal digital assistants (PDAs) and smartphones, and remote control centers [5]. The sensors can be wearable or embedded under the patient skin and placed in different body areas. They measure certain body parameters, such as blood glucose, weight, heart rate, blood pressure, temperature, respiration rate, electroencephalogram (EEG), electrocardiogram (ECG), etc. [6], [7]. The portable personal terminals collect and transmit the data from the sensors to the remote control centers to make them available to medical professionals. WBANs can create ubiquitous systems to provide health-care services anywhere and anytime, alleviating the workload in health centers [8].

One of the critical challenges of wireless health-care applications is the security and privacy of the patients’ medical records. The patient data are sent through an insecure wireless network environment, which can suffer from security risks such as interception, modification, or eavesdropping of data. Additionally, health-care providers are obligated by the Health Insurance Portability and Accountability Act (HIPAA) to protect the patients’ medical information from unauthorized access [8], [9]. Health information is sensitive because its unauthorized disclosure can cause family conflicts, patients experiencing psychological distress, and even the patients can be at risk of losing their jobs. Moreover, the malicious modification and fabrication of health data can result in incorrect medical diagnostics, extremely endangering the patients’ life [1].

On account of the security risks in WBANs, many authentication protocols have been proposed to achieve mutual authentication between the entities that transmit and receive health information to ensure that the patient’s medical data are not altered or disclosed to unauthorized parties. Some proposals are based on Public Key Infrastructure (PKI) [10], including [11]–[14]. However, the security mechanisms in PKI use many computational resources in the processing, storage, communication, and management tasks, which make PKI unsuitable for WBANs [15]. Other protocols without the requirement of PKI use the cryptographic operations bilinear pairing and map-to-point hash function [16], [17], such as the proposals in [18]–[22]. However, these operations have a high computational cost, which can have an impact on the batteries of the devices [23], [24]. Because some WBANs devices are implanted in the patient’s body, the batteries are inaccessible and difficult to recharge or replace; thus, these operations are not appropriated for this type of device. In contrast, the proposed authentication protocol is based only on the operations of Elliptic Curve Cryptography (ECC) scalar point multiplication, symmetric key encryption, xor, and conventional one-way hash function. The operations are lightweight. In comparison, the computation cost of bilinear pairing and the map-to-point hash function is many times higher than ECC scalar point multiplication [24]. Therefore, the proposal does not significantly impact the computing and battery resources of the resource-constrained WBANs devices.

A. MOTIVATION AND CONTRIBUTIONS

WBANs is a promising technology to provide economical, efficient, and proactive health-care services. It can give two significant advantages to patients and health professionals. The first is that WBANs can be location-independent monitoring systems. A WBAN node as an autonomous device can search for appropriate communication networks and transmit the medical information in a non-intrusive manner. The second advantage consists of allowing the patients to continue with their routine activities and be medical monitored in their homes, instead of staying in a hospital or frequently visiting health-care facilities for medical supervision [25].

However, to achieve the trust of patients and health-care providers, patient medical information must be effectively protected. Some of the privacy and security requirements of WBAN systems are data confidentiality, data integrity, data freshness, and authentication [6]. In consideration of the security problems in WBANs, we propose an authentication protocol named Two-Party Lightweight Authentication Protocol (TLAP), for the communication between the patient’s portable personal terminal and an application provider (AP). The protocol allows these two entities to be confident of each other identity and share a key to achieve the security properties of data confidentiality and integrity. The proposal ensures that the patient’s sensitive medical information is disclosed only to authorized entities, and it is not maliciously altered during transmission. The main contributions of this work are summarized as follows.

1) A secure self-certified authentication protocol. Because of the limited computational and energy resources in WBANs devices [26], the proposal does not use complex operations such as the associated with PKI, bilinear pairing, or map-to-point hash function. Instead, it is based on ECC scalar point multiplication, symmetric key encryption, and the lightweight operations xor and conventional hash function. Therefore, the proposal’s executing-time and communication-cost are low.

2) TLAP does not require a public key certificate. Schemes based on PKI use many computing resources in the tasks related to the generation, storage, verification, and revocation of certificates. Instead, the proposal uses self-certified public keys to avoid the overhead of managing certificates and verifying public keys before using them [27].

3) The proposal does not have the key escrow problem present in protocols that use Identity-based cryptography (IBC). One limitation of IBC is that the principals’ private keys are generated by a third party known as a private key generator (PKG). This creates a key escrow problem. Also, a malicious PKG could perform a man-in-the-middle (MITM) attack using the principals’ keys [27], [28]. In the proposed protocol, the principals create themselves their private keys, avoiding the key escrow problem. Further, the keys are known only by
the principals; thus, a third party cannot execute a MITM attack.
4) Detailed security analysis shows TLAP achieves high security. The security of the protocol was formally evaluated using the "Automated Validation of Internet Security Protocols and Applications (AVISPA) tool" [29] and "Burrows–Abadi–Needham (BAN) logic" [30]. These mechanisms showed TLAP achieves mutual authentication and is secure against replay and MITM attacks. Additionally, an informal analysis was performed to show the proposal is secure against potential attacks in WBANs.
5) A performance study of the TLAP’s computing requirements is presented. We analyzed TLAP in terms of the execution time of the operations involved in the protocol and the communication cost of the messages transmitted. The execution time and communication cost of TLAP and similar existing protocols were compared, which showed the proposal requires fewer computing resources.
6) The security features of TLAP and similar existing protocols were analyzed and compared. The analysis shows the proposal achieves better resistance to attacks and has more security features than the other schemes.

**B. PAPER ORGANIZATION**
The remainder of the article is organized as follows. Section II presents Related work. Section III contains preliminary information on the protocol. Section IV describes the proposed TLAP. Section V provides a formal security verification of TLAP using the AVISPA tool. Section VI presents a formal security verification of TLAP using BAN logic. In Section VII, an informal security analysis of TLAP is presented. Section VIII provides performance and security evaluations of TLAP and similar existing protocols. Section IX presents the discussion. Finally, Section X concludes the article.

**II. RELATED WORK**
In the following, we briefly analyze current authentication protocols for WBANs and IoT.

Some proposed authentication protocols are based on PKI, including [12], [13], [31]. PKI systems require a certificate authority (CA) to generate a certificate that binds the user’s identity with his/her public key. Before executing the authentication process, the certificate has to be verified. However, as the number of users in the system increases, the management of certificates becomes more difficult [32].

In [33], a certificateless authentication protocol for WBANs is proposed. The scheme aims to provide anonymity in the communication of the patient’s medical data with an AP. Also, APs and network managers are prevented from disclosing users’ identities or impersonating users. Similarly, in [34], a certificateless authentication protocol for WBANs in health-care applications is presented. Some of its features are client anonymity, non-repudiation, revocability, and key escrow resistance. However, these protocols use bilinear pair-

ing operations. Thus, their execution time is high, which is inappropriate to WBAN devices due to their limited computation capability and memory space, and their low power [35].

Furthermore, the scheme proposed in [33] is analyzed in [36], where it is indicated that the scheme does not provide anonymity and it is insecure against stolen verifier attack. Then, [36] also proposes an authentication scheme using IBC. Some of the advantages of the new proposal are that it does not involve bilinear pairing operations, the AP does not need to maintain a verifier table, and the protocol is scalable with respect to the addition of new clients. However, in [32] is pointed out that the protocol does not achieve anonymity; thus, tracking attacks are possible. Later, in [32] is proposed a new scheme to resolve the weakness of the [33] and [36] protocols and to achieve real anonymity. Unfortunately, in [37] is discovered that the scheme is vulnerable to impersonation attack, allowing a legal client and an adversary to impersonate another legal client. Later, in the same work, an anonymous authentication scheme is proposed with better performance than the protocol in [32]. However, both schemes, [37] and [32], are found insecure to impersonation attack in [38], where it is demonstrated that an AP can effortlessly impersonate a client. Additionally, in [39] is shown that the protocol in [37] allows an adversary to impersonate a legal client and that the scheme does not achieve mutual authentication.

An anonymous authentication protocol for WBAN is proposed in [35]. The proposal only requires one round of communication between the parties, aiming for high computational efficiency and low energy consumption to be suitable for resource-constrained WBAN devices. However, the security of the scheme is analyzed in [40], and it is shown the protocol is vulnerable to Denial of Service (DoS), key compromise impersonation, and stolen-verifier attacks. Then, the authors of [40] proposed an improved version of the protocol to resolve the vulnerabilities they found. Unfortunately, in [41], the improved protocol is also found insecure to replay and MITM attacks.

The previous works were examined when developing our authentication protocol. Considering that WBAN devices are resource-constrained and low-power, TLAP neither uses complex operations such as bilinear pairing nor is PKI-based. Thus, it has low execution time and communication cost. TLAP does not suffer from the key escrow problem because the principals generate the private keys. As a result, some attacks such as a MITM performed by a malicious PKG are avoided. Further, the TLAP’s security analysis showed the protocol achieves mutual authentication, and it resists potential attacks in WBANs.

**III. PRELIMINARIES**
In this section, preliminary information is presented. It includes the network model, security requirements, threat model, a brief introduction to ECC, computationally intractable problems, and the protocol assumptions.
A. NETWORK MODEL

The typical network model of WBANs is illustrated in Fig. 1. A WBAN consists of sensors, a controller, a Network Manager (NM), and Application Providers (APs). The sensors can be implanted in or worn on the user’s body. The sensors measure body parameters and send the data to a portable personal terminal, i.e., the controller. The latter collects health-related data and sends them to telemedicine APs to make them available to health-care professionals. Through examining the patient’s health information, medical practitioners can diagnose and give treatments to patients remotely.

The health information in WBANs is communicated in two phases. First, the data is transmitted from the sensors to the controller. Second, the controller sends information collected from various sensors to APs. In this article, we propose an authentication protocol to secure the second phase, i.e., the communication between a controller and an AP. The first phase of the communication, i.e., the data transmission over an open channel.

TLAP protocol considers the following three types of participants.

- WBAN Client: It is the controller of the WBAN. It consists of a portable personal terminal such as a smartphone, PDA, or a medical device that collects the sensors’ data. It is used by the patient to access medical services provided by APs.
- NM: It is the management server in the WBAN. It is responsible for the system’s parameters generation, the enrollment of WBAN Clients and APs, and the creation of the public keys of WBAN Clients and APs. Because NMs could belong to commercial organizations, NMs could misbehave such, as illegally accessing and collecting users’ private data to obtain commercial benefits. TLAP prevents these behaviors by avoiding the key escrow problem. Thus, NMs cannot impersonate WBAN Clients and APs to access sensitive information [43].
- AP: It provides remote services to authorized users, such as physician consults, patient monitoring, and medical treatments. It can be hospitals, clinics, physicians, etc.

B. SECURITY REQUIREMENTS

A telecare medicine information system (TMIS) should have the following security requirements [20].

- A secure and efficient mutual authentication and key agreement procedure to achieve secure communication over an open channel.
- The mutual authentication and key agreement procedure should provide anonymity and un-traceability.
- The session key should be authenticated to prevent attacks such as privileged insider and key impersonation.
- The mutual authentication and key agreement procedure should have low computing requirements because IoT devices have limited battery and computing resources.

C. THREAT MODEL

We follow the threat model for the IoT environment discussed in [43]. The notation $\mathcal{A}$ symbolizes a polynomial-time ($t$) bounded adversary. We briefly describe below the threat model.

The Dolev-Yao (DY) threat model is adopted in TLAP. In this model, adversary $\mathcal{A}$ (passive or active) has total control of the communication channel. $\mathcal{A}$ can eavesdrop upon, intercept, modify, decompose, and forge messages transmitted in the communication channel. However, $\mathcal{A}$ can see messages’ content only if $\mathcal{A}$ possesses the appropriated decryption keys [27], [44], [45].

We also apply the CK-adversary model [46], [47], a stronger threat model, which is currently considered the current de facto standard. Under this model, $\mathcal{A}$ has all the DY-model capabilities, but also, the adversary can compromise secret information such as session states and keys. Therefore, a protocol has to ensure that if ephemeral secrets are revealed, other parties’ secret information is not exposed [43].

Furthermore, $\mathcal{A}$ is capable of physically capturing smart devices because devices can be located in unattended environments. $\mathcal{A}$ can extract the secret credentials stored in a device to compromise the communication between the device and a legitimate entity.

Servers and gateways are trusted entities. They are fully reliable, and $\mathcal{A}$ is unable to compromise them.

D. ELLIPTIC CURVE CRYPTOGRAPHY

Let $p$ and $q$ be two large prime numbers and $E$ a non-singular elliptic curve over a finite field $F_p$: $E = y^2 = x^3 + ax + b \mod p$, where $a, b \in R_{F_p}$ and $4a^3 + 27b^2 \neq 0$. The points on the curve form an additive cyclic group $G$, and the point $P$ is the generator of the whole group with order $q$.

Elliptic curve has the following group properties [27].
1. **Point addition**: Let \( P \) and \( Q \) be two points over \( E_p(a,b) \), then point addition is \( P + Q = R \). \(-R\) is located where a line that joins \( P \) and \( Q \) intersects \( E_p(a,b) \), the reflection of \(-R\) on the x-axis is point \( R \).

2. **Point doubling**: Let \( P \) be a point over \( E_p(a,b) \), then point doubling is \( Q = 2P \). \(-Q\) is located where a tangent line at \( P \) intersects \( E_p(a,b) \), the reflection of \(-Q\) on the x-axis is point \( Q \).

3. **Scalar point multiplication**: Let \( P \) be a point over \( E_p(a,b) \) and \( x \in \mathbb{Z}_q^* \), then scalar point multiplication is \( Q = xP = \{P + P + \ldots + P \text{ (x times)}\} \), which consists of adding \( x \) times the point \( P \).

**E. COMPUTATIONALLY INTRACTABLE PROBLEMS**

The security of TLAP is based on the following computationally intractable problems [48], [49].

- **Elliptic Curve Discrete Logarithm Problem (ECDLP)**: Given points \( P \) and \( Q \) over \( E_p(a,b) \), where \( Q = xP \) and \( x \in \mathbb{Z}_q^* \). The probability of computing \( x \) from \( \{P, Q\} \) is \( \text{Adv}_{ECDLP}^\mathcal{A}(t) = \Pr[\mathcal{A}(P, Q, xP) = x : x \in \mathbb{Z}_q^*] \leq \epsilon. \text{Adv}_{ECDLP}^\mathcal{A}(t) \) is negligible.

- **Elliptic Curve Computational Diffie-Hellman Problem (ECDHP)**: Given points \( P, Q, R \) over \( E_p(a,b) \), where \( Q = xP, R = yP, \) and \( x, y \in \mathbb{Z}_q^* \). The probability of computing \( xyP \) from \( \{P, Q, R\} \) is \( \text{Adv}_{ECDHP}^\mathcal{A}(t) = \Pr[\mathcal{A}(P, Q, R) = xyP : x, y \in \mathbb{Z}_q^*] \leq \epsilon. \text{Adv}_{ECDHP}^\mathcal{A}(t) \) is negligible.

- **Elliptic Curve Decisional Diffie–Hellman Problem (ECDHDP)**: Given points \( P, xP, yP, \) and \( zP \) over \( E_p(a,b) \), where \( x, y, z \in \mathbb{Z}_q^* \), decide whether \( z = xy \) or a uniform value. ECDHDP is computationally infeasible when \( p \) is large, e.g., \( p \) is chosen at least as a 160-bit prime number [50].

**F. ASSUMPTIONS**

The following security properties are assumed in the proposed protocol.

1. The communication channel used in the registration phase is secure.
2. The one-way hash function used in TLAP has the collision-resistance property.
3. The Network Manager (NM) is a secure entity. \( \mathcal{A} \) is unable to compromise it.

**IV. PROPOSED PROTOCOL**

In this section, TLAP is described. The proposal is comprised of three phases: setup, registration, and mutual authentication. The setup phase is executed once at the environment’s start-up. The registration phase runs every time a new entity wants to become a member of the environment. Finally, the mutual authentication phase is performed each time two entities wish to communicate securely. The participants in TLAP and the working flow are shown in Fig. 2.

The notations used in the protocol description are explained in Table I.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>( P, q )</td>
<td>Two large prime numbers.</td>
</tr>
<tr>
<td>( E_p(a,b) )</td>
<td>( \text{Elliptic curve } E : y^2 = x^3 + ax + b \mod p ). Where ( a, b \in \mathbb{F}_p ) and ( 4a^3 + 27b^2 \neq 0 ) must hold. Generator of group ( G ).</td>
</tr>
<tr>
<td>Client ( a ), ID ( a )</td>
<td>A client in the IoT environment and its identity, respectively.</td>
</tr>
<tr>
<td>AP ( b ), ID ( b )</td>
<td>An application provider (AP) in the IoT environment and its identity, respectively.</td>
</tr>
<tr>
<td>Entity ( i ), ID ( i )</td>
<td>An entity in the IoT environment and its identity, respectively. The entity can be a client or an AP.</td>
</tr>
<tr>
<td>( sP ) (or ( sPb ))</td>
<td>The NM’s private and public keys pair.</td>
</tr>
<tr>
<td>( i ), ID ( i )</td>
<td>The predefined maximum allowable delay for message reception.</td>
</tr>
<tr>
<td>( \Delta T )</td>
<td>Secure hash function.</td>
</tr>
<tr>
<td>( H )</td>
<td>Symmetric key encryption and decryption of data ( x ) using the key ( k ), respectively.</td>
</tr>
<tr>
<td>( E_k(x), D_k(x) )</td>
<td>XOR operation.</td>
</tr>
<tr>
<td>( \oplus )</td>
<td>Given points ( P ) and ( Q ) over ( E_p(a,b) ), the xor operation of the x and y coordinates of ( P ) and ( Q ) is denoted as ( P \oplus Q ) [51].</td>
</tr>
<tr>
<td>Given ( x, y \in \mathbb{Z}_q^* ), the xor operation in ( \mathbb{Z}_q^* ) of ( x ) and ( y ) is denoted as ( x \oplus y ).</td>
<td></td>
</tr>
</tbody>
</table>

**A. SETUP PHASE**

In the setup phase, NM generates the system parameters. NM selects a random number \( s \in \mathbb{F}_p \mathbb{Z}_q^* \) as its private key and computes its public key, \( Pub_q = sP \). Then, it selects a one-way hash function \( H : \{0,1\}^* \rightarrow \{0,1\}^k \), where \( k \in \mathbb{Z}_q^* \). Finally, NM publishes the parameters \( \{P, p, q, Pub_q, H\} \) and keeps \( s \) as a secret.

**B. REGISTRATION PHASE**

In this procedure, a principal denoted Entity \( i \), registers with the NM using a secure channel. The results of this phase are the Entity \( i \’s \) public key \( Pub_i \), generated by the NM, and the Entity \( i \’s \) private key \( i \), created by the principal to avoid the key escrow problem. The procedure is described following.

1. Entity \( i \) selects its identity \( ID_i \) and a random number \( r_0 \in \mathbb{F}_p \mathbb{Z}_q^* \), and computes \( b_0 = H(ID_i||r_0).P \). Then,
it sends to NM: \(ID_i\) and \(b_0\).

2) NM selects a random number \(r_1 \in R\ \mathbb{Z}_q^*\) and computes \(b_1 = H(ID_i[|r_1|]|b_0).Pub_{a_0} = \{H(ID_i[|r_1|]|b_0) + H(ID_i[|b_0|]|b_1)\}.s\) and \(Pub_{b_1} = b_1 + H(ID_i[|b_0|]|b_1).Pub_{a_0} + b_0\). Then sends to Entity i: \(ID_i, b_1, a_0\).

3) Entity i computes its private key: \(i = a_0 + H(ID_i[|r_0|])\), and checks the validity of \((ID_i, b_1, a_0)\) through \(i.P = b_1 + H(ID_i[|b_0|]|b_1).Pub_{a_0} + b_0\). If the expression holds, Entity i accepts \(\{Pub_{b_1}, i\}\) as its keys pair.

4) NM publishes the public key \(Pub_{b_1}\) of Entity i.

The verification of the Entity i’s private key is as follows:

\[i.P = \{H(ID_i[|r_1|]|b_0) + H(ID_i[|b_0|]|b_1)\}.s + H(ID_i[|r_0|]).P\]

\[i.P = H(ID_i[|r_1|]|b_0).Pub_{a_0} + H(ID_i[|b_0|]|b_1).Pub_{a_0} + H(ID_i[|r_0|]).P\]

\[i.P = H(ID_i[|r_1|]|b_0).Pub_{a_0} + H(ID_i[|b_0|]|b_1).Pub_{a_0} + H(ID_i[|r_0|]).P\]

\[i.P = b_1 + H(ID_i[|b_0|]|b_1).Pub_{a_0} + b_0\]

**C. MUTUAL AUTHENTICATION PHASE**

Before exchanging medical information, WBAN clients and APs have to authenticate each other. The TLAP’s mutual authentication phase allows participants to identify with each other and be sure that the other party believes in their identity. For convenience, the WBAN client is denominated Client \(a\), and the AP is named AP \(b\). The mutual authentication phase consists of the following four steps. In Fig. 3, the steps are illustrated.

1) Client \(a\) selects a timestamp \(T_1\) and a random number \(r_1 \in R\ \mathbb{Z}_q^*\) and computes the following values: \(D_1 = (r_1 \oplus a).P, D_2 = (r_1 \oplus a).Pub_{b_0}, D_3 = EH_{(D_2)}(ID_a \oplus D_2), D_4 = H(D_1[|D_2]||ID_a[|ID_b][|T_1]).\) Where \(a\) is Client \(a\)’s private key and \(Pub_{b_0}\) is AP \(b\)’s public key. \(D_3\) is the encryption of \(ID_a \oplus D_2\) using \(ID_b\) as the encryption key. Then Client \(a\) sends to AP \(b\):

\(M_1 = \{T_1, D_1, D_3, D_4\}\).

2) AP \(b\) selects a timestamp \(T_2\) and verifies \(|T_2 - T_1| \leq \Delta T\). If true, it computes \(D_2 = b.D_1\) and \(ID_a = (D_2(H(D_2(D_3))) \oplus D_2), b\) is AP \(b\)’s private key and \(D_2(H(D_2(D_3)))\) is the decryption of \(D_3\) using the hash \(D_2\) as the decryption key. AP \(b\) verifies \(D_4 \oplus H(D_1[|D_2]||ID_a[|ID_b][|T_1]).\) If any of the verifications is false, AP \(b\) aborts the session. Otherwise, it selects a random number \(r_2 \in R\ \mathbb{Z}_q^*\), and computes the following:

\(D_5 = (r_2 \oplus b).P, D_6 = (r_2 \oplus b).Pub_{a_0}, \) and \(D_7 = H(D_5[|D_6]||D_2[|ID_a[|ID_b][|T_2]).\)

Where \(Pub_{a_0}\) is Client \(a\)’s public key. Then sends to Client \(a\):

\(M_2 = \{T_2, D_5, D_7\}\).

3) Client \(a\) selects a timestamp \(T_3\) and verifies \(|T_3 - T_2| \leq \Delta T\). If true, it computes \(D_6 = a.D_5\) and verifies

\(D_7 \oplus H(D_5[|D_6]||D_2[|ID_a[|ID_b][|T_2]).\) If any of the verifications is false, Client \(a\) aborts the session. Otherwise, it computes:

\(K = H(D_2[|D_6]||ID_a[|ID_b][|T_3]), \)

\(SK = H(K[|ID_a[|ID_b]).\) And sends to AP \(b\):

\(M_3 = \{T_3, D_8\}\).

4) AP \(b\) selects a timestamp \(T_4\) and verifies \(|T_4 - T_3| \leq \Delta T\). If true, it verifies \(D_8 = H(D_2[|D_6]||ID_a[|ID_b][|T_3]).\) If any of the verifications is false, it aborts the session. Otherwise, AP \(b\) computes \(K = D_2 \oplus D_6\), and the session key \(SK = H(K[|ID_a[|ID_b]).\) The mutual authentication between Client \(a\) and AP \(b\) is successful, and the principals have generated the same session key \(SK\) to exchange data with confidentiality.

**V. FORMAL VERIFICATION THROUGH AVISPA TOOL**

In this section, we present a formal security verification on TLAP through the SPAN+AVISPA tool.

AVISPA is an automated validation tool for security protocols. The language “High Level Protocol Specification Language (HLPSL)” is used to describe the protocol under verification. HLPSL is a role-oriented language in which the roles specify the information known by the principals, including pre-shared keys and security algorithms, role interactions, and state transitions [52]. SPAN tool is an animator of protocols in HLPSL. SPAN tool is used to build message sequence charts of the messages transmitted both during normal execution of the protocol and under attack conditions in case of vulnerabilities found [53]. AVISPA tool uses four back-ends to analyze the protocol and search for attacks in the protocol’s security properties. The back-ends consist of “On-the-fly Model-Checker (OFMC), "Constraint-Logic-based Attack Searcher (CL-AtSe)," "SAT-based Model-Checker (SATMC)," and "Tree Automata based on Automatic Approximations for the Analysis of Security Protocols (TA4SP)" [29]. AVISPA assumes a Dolev–Ya’alon threat model.

During the security analysis, the AVISPA tool performs three verifications. First, it verifies if the HLPSL protocol description can execute completely; thus, it can reach a state where possible attacks can happen. Second, the possibility of replay attacks is checked. For this validation, the back-ends supply the intruder with the knowledge of normal sessions between authorized participants, check if the legitimate agents can execute the protocol by searching for a passive intruder, and verify if a replay attack exists. Third, the back-ends perform the Dolev–Ya’alon checking, where it is verified whether a MITM attack can be mounted by \(\alpha\). After the analysis, the AVISPA tool outputs its conclusion about the security of the protocol, whether it is safe, unsafe, or the study is inconclusive [54].

The analysis output includes the sections described below [55].

- **SUMMARY**: Indicates the analysis result, whether the protocol is safe, unsafe, or the analysis is inconclusive.
- **DETAILS**: Specifies the analysis results, explaining why the protocol is concluded as safe, why the analysis is inconclusive, or in case of attacks found, under what conditions attacks can be performed in the protocol.
Select: T₁ and r₁ ∈ R
Compute:
D₁ = (r₁ ⊕ a). P
D₂ = (r₁ ⊕ a). Pubb
D₃ = E_{H(D₂)}(IDa ⊕ D₂)
D₄ = H(D₁||IDa||IDb||T₁)
M₄ = (T₁, D₁, D₂, D₃)

Select: T₂
Verify: |T₂ - T₁| ≤ a T
Compute:
D₅ = b. D₁
ID₅ = (H(ID₂||D₃)) ⊕ D₂
Verify: D₅ ?= H(D₁||ID₂||ID₅||ID₆||T₁)
Select: r₂ ∈ Rₗ
Compute:
D₆ = (r₂ ⊕ b). P
D₇ = (r₂ ⊕ b). Pubb
D₈ = H(D₇||ID₇||ID₈||ID₉||T₂)
M₆ = (T₂, D₅, D₆)

Select: T₃
Verify: |T₃ - T₂| ≤ a T
Verify: |T₄ - T₃| ≤ a T
Verify: D₄ ?= H(D₇||ID₇||ID₈||ID₉||T₃)
Compute:
K₄ = D₅ ⊕ D₆
SK₄ = H(K₄||ID₆||ID₇)

Select: T₄
Verify: |T₄ - T₃| ≤ a T
Verify: |T₄ - T₃| ≤ a T
Verify: D₈ ?= H(D₇||ID₇||ID₈||ID₉||T₃)
Compute:
K₄ = D₅ ⊕ D₆
SK₄ = H(K₄||ID₆||ID₇)

FIGURE 3. Authentication phase of TLAP protocol.

- PROTOCOL: It shows the file path of the protocol under validation.
- GOAL: Indicates the security goals of the analysis performed by AVISPA, specified in the HLPSL protocol. In case of attacks found, shows the unachieved goals.
- BACKEND: Mentions the back-end used in the analysis.
- The final section includes statistics of the state and time took in the analysis. In the case of attacks found, it includes the trace of the vulnerabilities.

The TLAP’s security goals specified in the HLPSL description were mutual authentication and secrecy of the nonces used in the session key. The mutual authentication goal requires that agents be correct in believing the intended party is in the current session. He/she has reached a certain state, and he/she accepts some value that can only be used once with the same participant. During the analysis of the HLPSL protocol, if a security goal is violated, the AVISPA tool determines the protocol as unsafe. It displays an attack trace with the message sequence that leads to an attack. The back-ends that we used in the verification of TLAP were CL-

% Version of 2006/02/13
SUMMARY
SAFE
DETAILS
BOUNDED_NUMBER_OF_SESSIONS
TYPE_DMODEL
PROTOCOL
/home/span/span/testsuite/results/protocol.if
GOAL
As Specified
BACKEND
OFMC
COMMENTS
STATISTICS
parseTime: 0.00s
searchTime: 2.48s
visitedNodes: 518 nodes
depth: 10 plies

FIGURE 4. Analysis results of the AVISPA tool. a) Analysis result under OFMC back-end. b) Analysis result under CL-AtSe back-end.

AtSe and OFMC because they support the xor operation [56]. The security verification output is in Fig. 4, which shows that both CL-AtSe and OFMC back-ends concluded TLAP is secure against replay and MITM attacks. Also, the mutual authentication and secrecy goals were met.
TABLE II. BAN logic notations.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>P Q</td>
<td>Principals.</td>
</tr>
<tr>
<td>K</td>
<td>Encryption key.</td>
</tr>
<tr>
<td>X, Y</td>
<td>Statements.</td>
</tr>
<tr>
<td>P ∥ X</td>
<td>P believes X.</td>
</tr>
<tr>
<td>P ∽ X</td>
<td>P once said X.</td>
</tr>
<tr>
<td>P ∼ X</td>
<td>P sees X.</td>
</tr>
<tr>
<td>P ⇒ X</td>
<td>P has jurisdiction over X.</td>
</tr>
<tr>
<td>P K Q</td>
<td>X is fresh, has not been used in old protocol executions.</td>
</tr>
<tr>
<td>P X Q</td>
<td>P and Q may use K as a shared key in their communication.</td>
</tr>
<tr>
<td>{X}</td>
<td>X is a secret known only by P and Q. Principals they trust may also know it.</td>
</tr>
<tr>
<td>⟨X⟩Y</td>
<td>Encryption of X using K.</td>
</tr>
</tbody>
</table>

TABLE III. BAN logic rules.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1)</td>
<td>P</td>
</tr>
<tr>
<td>(2)</td>
<td>P</td>
</tr>
<tr>
<td>(3)</td>
<td>P</td>
</tr>
<tr>
<td>(4)</td>
<td>P</td>
</tr>
<tr>
<td>(5)</td>
<td>P</td>
</tr>
<tr>
<td>(6)</td>
<td>P</td>
</tr>
</tbody>
</table>

VI. FORMAL VERIFICATION THROUGH BAN LOGIC

In this section, we present a formal security analysis of TLAP using BAN logic.

BAN logic consists of a set of principles and postulates used for reasoning about what principals believe about each other and verify the trustworthiness of the exchanged information in the protocol [30].

A. BAN NOTATIONS AND RULES

In Table II the BAN logic notations are presented, and in Table III the BAN logic rules. The notations and rules are used in the TLAP’s security proof.

B. TLAP’S GOALS

The following security goals need to be achieved by TLAP to demonstrate its secure mutual authentication. The notation \(E_a\) represents the Client, and \(E_b\) the AP b.

**Goal 1:** \(E_a \equiv (E_a SK \mapsto E_b)\).

**Goal 2:** \(E_b \equiv (E_a SK \mapsto E_b)\).

**Goal 3:** \(E_a \equiv E_b \equiv (E_a SK \mapsto E_b)\).

**Goal 4:** \(E_b \equiv E_a \equiv (E_a SK \mapsto E_b)\).

C. TLAP’S IDEALIZED FORM

The idealized forms of the messages transmitted in TLAP are as follows:

**Message 1:** \(\{I D_a\}_{(r_1 a)}\), \(\{I D_a, E_a \equiv E_b, T_1\}_{(r_1 a)}\).

**Message 2:** \(\{E_a \equiv E_b\}_{(r_2 b)}, \{E_a SK \mapsto E_b\}_{(r_2 b)}\).

**Message 3:** \(\{E_a \equiv E_b\}_{(r_2 b)}, \{E_a SK \mapsto E_b\}_{(r_2 b)}\).

D. TLAP’S ASSUMPTIONS

The assumptions of the TLAP’s initial state are the following:

**Assumption 1** \(E_a \equiv E_b \Rightarrow (r_2, T_2)\).

**Assumption 2** \(E_a \equiv \#(T_2)\).

**Assumption 3** \(E_a \equiv P_{ab}\mapsto E_b\).

**Assumption 4** \(E_b \equiv E_a \Rightarrow (r_1, T_1, T_3)\).

**Assumption 5** \(E_b \equiv \#(T_1, T_3)\).

**Assumption 6** \(E_b \equiv P_{ab}\mapsto E_a\).

E. TLAP’S BAN LOGIC PROOF

Below is the BAN logic proof demonstrating that TLAP achieves the mutual authentication goals.

From Message 1 of TLAP idealized form, we obtain:

**Step 1:** \(E_b \equiv \{I D_a\}_{(r_1 a)}\), \(\{I D_a, E_a \equiv E_b, T_1\}_{(r_1 a)}\).

**Step 2:** From Step 1, applying Rule 6, we get: \(E_b \equiv \{I D_a\}_{(r_1 a)}\).

**Step 3:** From Step 1 and Step 2, applying Rule 6, we get: \(E_b \equiv \{I D_a, E_a \equiv E_b, T_1\}\).

**Step 4:** From Step 3, applying Rule 4 and Assumption 4, we get: \(E_b \equiv \#(I D_a, E_a \equiv E_b, T_1)\).

From Message 2, we obtain:

**Step 5:** \(E_a \equiv \{E_a \equiv E_b\}_{(r_2 b)}, \{E_a \equiv E_b\}_{(r_2 b)}\).

**Step 6:** From Step 5, applying Rule 6, we get: \(E_a \equiv \{E_a \equiv E_b\}_{(r_2 b)}\).

**Step 7:** From Step 5 and Step 6, applying Rule 6, we get: \(E_a \equiv \{E_a \equiv E_b\}_{(r_2 b), E_a \equiv E_b, T_2}\).

**Step 8:** From Step 7, applying Rule 4 and Assumption 2, we get: \(E_a \equiv \#(E_a \equiv E_b, E_a \equiv E_b, T_2)\).

**Step 9:** From Step 7, applying Rule 1 and the assumption...
From Step 10, applying Rule 2, we get: $E_a \equiv E_b \iff (E_a^{(r_1 \oplus a)} = E_b^{(r_2 \oplus b)}) \iff E_a \leftrightarrow E_b, T_2$.

**Step 11:** From Step 10, applying Rule 3 and Assumption 1, we get: $E_a \equiv E_b \iff (E_a^{(r_2 \oplus b)} = E_b, E_a \leftrightarrow SK, E_b, T_2)$.

**Step 12:** From Step 10, applying Rule 5, we get: $E_a \equiv E_b \iff (E_a^{SK} \leftrightarrow E_b)$. (Goal 3)

**Step 13:** From Step 11, applying Rule 5, we get: $E_a \equiv (E_a^{SK} \leftrightarrow E_b)$. (Goal 1)

From Message 3, we obtain:

**Step 14:** $E_b \Leftrightarrow (E_a^{(r_1 \oplus a)} = (E_a^{(r_1 \oplus a)})) \iff (E_a^{(r_1 \oplus a)} = (E_a^{(r_1 \oplus a)})) \iff E_a \leftrightarrow E_b, T_3$.

**Step 15:** From Step 14 and Step 2, applying Rule 6, we get: $E_b \Leftrightarrow (E_a^{(r_1 \oplus a)} = (E_a^{(r_1 \oplus a)})) \iff E_a \leftrightarrow E_b, T_3$.

**Step 16:** From Step 15, applying Rule 4 and Assumption 5, we get: $E_b \equiv E_a \iff (E_a^{(r_1 \oplus a)} = (E_a^{(r_1 \oplus a)})) \iff E_a \leftrightarrow E_b, T_3$.

**Step 17:** From Step 15, applying Rule 1 and the assumption $E_b \equiv E_a \iff (E_a^{(r_1 \oplus a)} = (E_a^{(r_1 \oplus a)})) \iff E_a \leftrightarrow E_b, T_3$.

**Step 18:** From Step 17, applying Rule 2, we get: $E_b \Rightarrow (E_a^{(r_1 \oplus a)} = (E_a^{(r_1 \oplus a)})) \iff E_a \leftrightarrow E_b, T_3$.

**Step 19:** From Step 18, applying Rule 3 and Assumption 4, we get: $E_b \Rightarrow (E_a^{(r_1 \oplus a)} = (E_a^{(r_1 \oplus a)})) \iff E_a \leftrightarrow E_b, T_3$.

**Step 20:** From Step 18, applying Rule 5, we get: $E_b \Rightarrow (E_a^{(r_1 \oplus a)} = (E_a^{(r_1 \oplus a)})) \iff E_a \leftrightarrow E_b, T_3$.

**Step 21:** From Step 19, applying Rule 5, we get: $E_b \Rightarrow (E_a^{(r_1 \oplus a)} = (E_a^{(r_1 \oplus a)})) \iff E_a \leftrightarrow E_b, T_3$.

The four security goals to have mutual authentication are satisfied.

**VII. INFORMAL SECURITY ANALYSIS**

In this section, we present an analysis of the security properties and attack resistance of TLAP.

**A. CONFIDENTIALITY**

The random numbers $r_1$ and $r_2$ have to be known only by the legitimate parties because they are used in the construction of the session key $SK$. Additionally, the Client a’s identity $ID_a$ has to be confidential to prevent tracing attacks.

All these data are sent protected through the high entropy of the operations ECC scalar point multiplication and xor. $r_1$ and $r_2$ are xor-ed with the private keys of their creators in $D_1$, $D_2$, $D_3$, and $D_4$; therefore, the random numbers are only known by their creators. After the xor operation, the numbers are ciphered through the scalar point multiplication with the base point $P$. Therefore, it is infeasible for $\mathcal{A}$ to obtain $r_1$ and $r_2$ or even $(r_1 \oplus a)$ and $(r_2 \oplus b)$ from the messages sent.

Finally, $ID_a$ is ciphered through an encryption algorithm, e.g., AES, in $D_3$. The hash of $D_2$ is the ephemeral encryption key. Because only AP b knows the private key $b$, only AP b can compute $D_2$ from the data in $M_1$. Therefore, only AP b can decipher $D_1$ to know the client that wants to communicate with him/her.

**B. DATA INTEGRITY**

We describe below what happens if a message is maliciously modified during transmission. In all cases, the recipient can detect the data integrity violation; thus, he/she terminates the communication.

- **Modification of $M_1$:** If any of $T_1$, $D_1$, $D_4$, or $D_4$ are modified, the verification of the message digest $D_4$ will be false. Even if $D_4$ is altered to hide the changes, Client a can detect the modifications in $M_2$ because it will contain a different $D_2$ from the one he/she created. Similarly, AP b can detect in $M_3$ the modifications of $M_1$, because $M_3$ will have a different $D_2$ from the received in $M_4$.

- **Modification of $M_2$:** If any of $T_2$, $D_5$, or $D_7$ are modified, the verification of the message digest $D_7$ will be false. $\mathcal{A}$ cannot alter $D_7$ in a deterministic manner to hide the changes, considering it is constructed using the ephemeral secret $D_2$. Only AP b can obtain $D_2$ because $D_2$ was ciphered with AP b’s public key.

- **Modification of $M_3$:** If $T_3$ or $D_8$ are modified, the verification of the message digest $D_8$ will be false. $\mathcal{A}$ cannot alter $D_8$ in a deterministic manner to hide the changes, considering $D_8$ is constructed using the ephemeral secrets $D_2$ and $D_6$ obtained only from $\mathcal{A}$. Only Client a can obtain $D_6$ because $D_6$ was ciphered with his/her public key.

**C. CLIENT ANONYMITY**

$\mathcal{A}$ cannot know the Client a’s identity $ID_a$ to perform tracing attacks because $ID_a$ is never sent in clear-text. When transmitted in the public channel, $ID_a$ is either encapsulated by a hash function or ciphered. In $D_3$, $D_7$, and $D_8$, $ID_a$ is hashed. And in $D_3$, $ID_a$ is ciphered using the hash of $D_2$ as the ephemeral encryption key. $D_2$ is known only by AP b and Client a. Therefore, the Client’s identity remains anonymous to $\mathcal{A}$.

Furthermore, all transmitted messages contain ephemeral random numbers that belong to the current session. Thus, the messages are refreshed in every session, impeding $\mathcal{A}$ from associating old sessions with a particular client.

**D. PERFECT FORWARD AND BACKWARD SECRECY**

If $\mathcal{A}$ in some way captures the current session key $SK$, or even the private keys a and b, he/she cannot use the keys to obtain $SK$’s of old and future sessions because each $SK$ is built with values associated with the current session. $SK$ is built using the secret points $D_2$ and $D_6$. And $D_2$ and $D_6$ are constructed using the random numbers $r_1$ and $r_2$. Both $r_1$ and $r_2$ are secret, ephemeral, unpredictable, and of single use. Therefore, $\mathcal{A}$ can neither generate $SK$’s of previous sessions nor $SK$’s of future sessions.
Furthermore, the dependence of SK with the current session also enforces that clients and APs can only get data transmitted when they were part of the network, neither the previously transmitted data nor the future one.

**E. KNOWN SESSION KEY SECURITY**

Each run of a key agreement protocol should result in a unique session key. Therefore, if $\mathcal{A}$ captures a session key, he/she is not able to compromise others [57]. In TLAP, the session key SK is built using the ephemeral and fresh random numbers $r_1$ and $r_2$. They are unpredictable and different per session. Therefore, $\mathcal{A}$ cannot create a new SK even if he/she compromises an old one.

**F. MUTUAL AUTHENTICATION AND KEY AGREEMENT**

In TLAP, the entities authenticate each other by demonstrating the knowledge of their private keys. Client $a$ proves to AP $b$ the knowledge of the private key $a$ through computing $D_6 = a.D_5$, and sending to $AP$ the hash of $D_6$ in message $M_5$. AP $b$ proves to Client $a$ the knowledge of the private key $b$ through computing $D_2 = b.D_1$, and sending to Client $a$ the hash of $D_2$ in message $M_2$. Additionally, all the transmitted messages are authenticated through the verifications $D_4 \overset{?}{=} H(D_1||D_2||ID_a||ID_b||T_1)$, $D_7 \overset{?}{=} H(D_2||ID_a||ID_b||T_2)$, and $D_8 \overset{?}{=} H(D_2||ID_a||ID_b||T_3)$. If any of the verifications is false, the communication is terminated. After authenticating messages and participants, the principals compute the same session key using the values $D_2$ and $D_6$. Furthermore, all transmitted messages change with each session because they contain ephemeral secret random numbers. Therefore, the proposal achieves the security goals of mutual authentication and key agreement.

**G. RESISTANCE TO TRACING ATTACK**

In this attack, $\mathcal{A}$ tries to guess the identity of the client behind the messages from different sessions [58], [59].

In TLAP, the Client identity $ID_a$ is never sent in plain text on the public channel to prevent $\mathcal{A}$ from tracing the messages of a particular client. $ID_a$ is sent ciphered in $D_3$ using the hash of $D_2$ as the ephemeral encryption key. And is sent encapsulated in a hash in $D_4$, $D_7$, and $D_8$. $\mathcal{A}$ cannot decipher $D_3$ because $\mathcal{A}$ does not know $D_2$, as $D_2$ is ciphered with AP $b$’s public key. Further, it is infeasible for $\mathcal{A}$ to obtain $ID_a$ from $D_4$, $D_7$, and $D_8$ because these data come from a one-way and collision-resistant hash function.

Furthermore, all messages contain at least one random number. This makes the messages different and unpredictable in each session and impedes $\mathcal{A}$ from finding a relationship between messages of other sessions to guess the client behind them.

**H. RESISTANCE TO OFF-LINE IDENTITY GUESSING ATTACK**

In this attack, $\mathcal{A}$ tries to find the Client identity $ID_a$ employing non-interactive guess techniques in captured messages.

In TLAP, the identity $ID_a$ is transmitted ciphered using an encryption algorithm, e.g., AES. The hash of $D_2$ is the ephemeral encryption key. The transmission of $ID_a$ is in this form: $D_3 = E_{H(D_2)}(ID_a \oplus D_3)$. $\mathcal{A}$ cannot decrypt $D_3$ because he/she knows neither $H(D_2)$ nor $D_2$.

The messages $D_4$, $D_7$, and $D_8$, contain the hash of $ID_a$. It is infeasible for $\mathcal{A}$ to obtain $ID_a$ from them because the hash function is one-way and collision-resistant. Further, $\mathcal{A}$ cannot generate $D_3$, $D_7$, and $D_8$ because they include secret parameters such as $D_2$ and $D_6$, unknown to $\mathcal{A}$.

If the adversary tries to guess $ID_a$ using $D_3$, he/she has to perform the following steps to verify if the value is correct. He/she guesses $ID_a$ and $D_2$, computes $ID_a \oplus D_2$ and $H(D_2)$, and encrypts $ID_a \oplus D_2$ using $H(D_2)$ as the key. Finally, he/she compares the result with $D_3$. If they are equal, the guessed $ID_a$ is correct. However, guessing two parameters is not computationally feasible in polynomial time. If the lengths of $ID_a$ and $D_2$ are $n$ and $s$ bits, respectively, the probability of guessing the values at the same time approximates $\frac{1}{2^{n+s}}$ [60], [61].

Furthermore, reductions of the ECDLP on an elliptic curve $E/F_p$ to $F_{p^k}^*$, where the smallest possible $k$ is named the embedding degree, are only practical when $k < \log^2(p)$ [62]. ECDHP holds in an elliptic curve when $p$ is chosen large [50]. Therefore, $\mathcal{A}$ cannot use Weil or Tate pairings in form $e(D_1, Pub_b) \overset{?}{=} e(P, D_2)$ to find $D_2$ and use it to decrypt $ID_a$.

**I. RESISTANCE TO IMPersonATION ATTACK**

In TLAP, all the transmitted messages are constructed using at least one of the random numbers $r_1$ and $r_2$, and only their generators know these numbers, Client $a$ and AP $b$, respectively. To maintain the secrecy of $r_1$ and $r_2$, the numbers are xor-ed with the generators’ private keys. Therefore, $\mathcal{A}$ cannot get them because he/she does not have the private keys. After xor-ing, $r_1$ and $r_2$ are ciphered through ECC scalar point multiplication with the recipient’s public key in $D_2$ and $D_6$. Consequently, only the legitimate recipient can get $D_2$ and $D_6$ because only he/she knows the corresponding private key to decipher. After obtaining $D_2$ and $D_6$, Client $a$ and AP $b$ sent each other the values in $D_7$ and $D_8$ to demonstrate they are the legitimate possessors of the private keys. Thus, proving their identities. It is infeasible for $\mathcal{A}$ to impersonate Client $a$ and AP $b$ because $\mathcal{A}$ does not have the necessary keys to compute $D_2$ and $D_6$.

**J. RESISTANCE TO INJECTION ATTACK**

Client $a$ and AP $b$ can detect counterfeit messages in a similar manner as described in subsection Resistance to Impersonation Attack. All the transmitted messages contain at least one of the shared secrets $D_2$ and $D_6$, unknown to $\mathcal{A}$. $D_2$ and $D_6$ are constructed using the sender’s private key and are ciphered with the recipient’s public key. Thus, only the legitimate recipient can decipher them. Further, the messages contain a message-digest consisting of a hash of...
the timestamp and the shared secrets $D_2$ and $D_6$. If $\mathcal{A}$ alters a message, the principals will detect the change because the digest will be invalid. $\mathcal{A}$ cannot construct a valid digest or modify one in a deterministic manner to make it authentic because $\mathcal{A}$ does not know the random numbers $r_1$ and $r_2$, and the private keys $a$ and $b$, used to construct $D_2$ and $D_6$. Additionally, $\mathcal{A}$ cannot get $D_2$ from $D_1$ nor $D_6$ from $D_5$ since $\mathcal{A}$ does not have the private keys necessary to compute $D_2$ and $D_6$.

Furthermore, $\mathcal{A}$ cannot fabricate messages because in $D_2$ and $D_6$, the parties have to demonstrate they were able to compute the secrets $D_2$ and $D_6$. If $\mathcal{A}$ sends a fabricated message, the recipient will detect that the message does not contain the secret value that he/she previously sent.

K. RESISTANCE TO MITM ATTACK

Suppose $\mathcal{A}$ captures $M_1$, $M_2$, and $M_3$ during a session between Client $a$ and AP $b$. $\mathcal{A}$ wants to modify $M_1$ to make it appear as another valid message. Thus, $\mathcal{A}$ generates a new random number $r_1$, computes $D_1$, $D_2$, $D_3$, and $D_4$ using his/her private key or a fraudulent one, and sends $M_1$ to AP $b$. After receiving $M_1$, AP $b$ generates a new random number $r_2$, computes $D_5$, ciphers $r_2$ with the public key of the legitimate Client $a$ in $D_6$, computes $D_7$, and sends $M_2$ to $\mathcal{A}$. $M_3$ contains a challenge that only the legitimate Client $a$ can answer. In this message, AP $b$ asks for the computation of $D_6$, which can only be done with the Client $a$’s private key. As $\mathcal{A}$ does not have the key, $\mathcal{A}$ is unable to answer the challenge correctly. Thus, when AP $b$ receives an incorrect $D_6$ from $\mathcal{A}$, AP $b$ terminates the communication.

A similar situation will happen when $\mathcal{A}$ wants to modify $M_2$ to make it appear as another valid message sent by AP $b$. This case consists of the following. First Client $a$ generates a new $r_1$, computes $D_1$ and $D_2$ using his/her private key, ciphers $D_2$ with the public key of the legitimate AP $b$, computes $D_3$ and $D_4$, and finally, sends $M_1$ to $\mathcal{A}$. In $M_1$, $\mathcal{A}$ is challenged to compute $D_2$ and send the value to Client $a$ in $M_2$. $\mathcal{A}$ does not have the private key of AP $b$; thus, $\mathcal{A}$ cannot respond correctly to the challenge. Consequently, when Client $a$ receives an incorrect $D_2$ from $\mathcal{A}$, Client $a$ aborts the communication.

Finally, $\mathcal{A}$ is unable to modify $M_3$ to deceive AP $b$ because $M_3$ contains the answers to the two challenges described above. $\mathcal{A}$ does not have the private keys of Client $a$ and AP $b$; thus, $\mathcal{A}$ cannot compute $D_2$ and $D_6$ to create a valid $M_3$. Therefore, TLAP resists MITM attacks.

L. RESISTANCE TO PRIVILEGED INSIDER ATTACK

A privileged insider user is any entity that has access to resources that would result in significant damage to an organization if compromised. This attack consists of a privileged insider user wanting to impersonate a client in other systems, using the client’s credentials of this system [63].

A major advantage of TLAP is the avoidance of the key escrow problem. In TLAP, the private key is not generated by a PKG or an NM. The private key is created by the entity to which it belongs, i.e., Client $a$ and AP $b$; thus, only that entity knows the private key and can use it as proof of identity. Additionally, the public key’s authenticity can be verified publicly without requiring a certificate issued by the NM. Therefore, the NM or any privileged insider is unable to impersonate a legitimate entity of this system when accessing other services, as the NM does not know the entity’s private key.

M. RESISTANCE TO REPLAY ATTACK

All the transmitted messages contain timestamps and ephemeral random numbers used to guarantee the messages’ freshness. If $\mathcal{A}$ captures valid messages and maliciously repeats or delays them, Client $a$ and AP $b$ will detect that the messages’ timestamps have a transmission delay longer than permitted. $\mathcal{A}$ is unable to modify the timestamps to make them valid for the current session, as described in subsection Data Integrity. Consequently, Client $a$ and AP $b$ will abort the communication. Furthermore, Client $a$ and AP $b$ can detect replay attacks because the ephemeral ECC points $D_2$ and $D_6$ of the inauthentic messages will be different from the previously shared in the messages $M_1$ and $M_2$, and $\mathcal{A}$ cannot alter the ECC points in the messages to make them valid because $\mathcal{A}$ does not know the new random numbers and the private keys used to compute them. Therefore, TLAP is resistant to replay attacks.

N. RESISTANCE TO KNOWN SESSION-SPECIFIC TEMPORARY INFORMATION ATTACK

In this attack, $\mathcal{A}$ compromises the session key through the exposure of session-temporal secrets, such as random numbers. $\mathcal{A}$ can perform this attack due to random numbers are not usually stored in a secure memory, as is done with long-term secrets such as keys [64].

In TLAP, the session key is composed of both short-term and long-term secrets. $SK$ comprises the ephemeral ECC points $D_2$ and $D_6$, which are constructed with the random numbers $r_1$ and $r_2$, respectively, and the private keys $a$ and $b$, respectively. The random numbers $r_1$ and $r_2$ are secret, ephemeral, unpredictable, and of single-use. The private keys $a$ and $b$ are secrets only known by Client $a$ and AP $b$, respectively. Even if $\mathcal{A}$ compromises $r_1$ and $r_2$, $\mathcal{A}$ is unable to obtain $SK$ because he/she does not know $a$ and $b$.

O. RESISTANCE TO DOS ATTACK

Client $a$ and AP $b$ can detect intents of blocking their access to services or exhausting their resources. If $\mathcal{A}$ sends many replayed messages to a principal, the principal will detect that the timestamps have a transmission delay longer than permitted. If the timestamps in plain text are modified to make them appear valid in the current session, the recipient will detect that the messages’ digests are incorrect. $\mathcal{A}$ cannot change the digests in a deterministic manner to make them valid, as described in subsection Data Integrity. On the other hand, if $\mathcal{A}$ does not send replayed messages, but he/she sends many valid and fresh messages originated by him/her,
the principal will detect the malicious intent when receiving many messages from the same party. $A$ cannot disguise his/her identity and pretend he/she is another entity because he/she would require to have the other entity’s private key to respond correctly to the challenges received in the messages, as described in subsection Resistance to Impersonation Attack.

**P. RESISTANCE TO THE DE-SYNCHRONIZATION ATTACK**

In TLAP, $A$ cannot de-synchronize the parties in the values required for the authentication because no long-term secret is modified in the protocol. The only values that change per session are the timestamps and the random numbers used to generate the ephemeral ECC points $D_2$ and $D_6$. In case that Client $a$ receives an invalid $D_2$ in the message digest $D_2$ of $M_2$, Client $a$ terminates the session. The same applies for AP $b$ when receiving an incorrect $D_6$ in the message digest $D_6$ of $M_3$. Because the parties do not modify long-term secrets after a protocol run, the principals can use the long-term secrets to start new sessions even after de-synchronization intenlets from $A$.

**Q. RESISTANCE TO KEY DISCLOSURE ATTACK**

Long-term key disclosure attack: The private keys $a$ and $b$ of Client $a$ and AP $b$, respectively, are never used in invertible operations that could cause $A$ to obtain them. Key $a$ is used to generate the ECC points $D_2$ and $D_3$, and key $b$ is used to create $D_5$ and $D_6$. Because of the high entropy of ECC scalar point multiplication, it is infeasible for $A$ to invert the operation to obtain $a$ and $b$. $D_2$, $D_3$, $D_5$, and $D_6$ are also used in the messages’ digests. However, because the hash function is one-way and collision-resistant, it is infeasible for $A$ to invert the hash output to get the values.

Short-term key disclosure attack: The session key is constructed using the secret points $D_2$ and $D_6$, the AP $b$’s identity $ID_b$, and the Client $a$’s secret identity $ID_a$. $D_2$ and $D_6$ are sent in the public channel after hashing them with a one-way collision-resistant hash function; thus, it is infeasible for $A$ to get them. $ID_a$ is sent encrypted using the hash of $D_2$ as the ephemeral encryption key.. Because $A$ does not know $D_2$, he/she cannot decipher the identity. Finally, the ECC points $D_1$ and $D_5$ are used by the recipients to compute $D_2$ and $D_6$, respectively. However, $D_2$ and $D_6$ can only be generated if the recipient possesses the appropriated private key, $a$ and $b$, respectively. Because of the ECDLP, it is infeasible for $A$ to compute $D_2$ and $D_6$ from $D_1$ and $D_5$ without knowing $a$ and $b$.

**VIII. PERFORMANCE AND SECURITY EVALUATION**

We evaluated the performance of TLAP in terms of the following criteria. The execution time of the operations that comprise the authentication phase. The communication cost concerning the number of bits the parties transmit to authenticate. Finally, the security features that the protocol provides. Many proposals of authentication protocols in the literature use these parameters to evaluate the protocols’ performances, including [50], [51], [65]–[68]. The parameters can help to determine the viability of a scheme to secure a device. There are some operational requirements that IoT networks have to meet, including real-time operation. The latter is related to the ability of a system to respond correctly and predictably and meeting deadlines. The use of cryptographic operations should not delay a system in meeting its response deadlines [69].

This section presents a comparative analysis of the performances of TLAP and similar existing protocols, such as the proposed by Ying et al. [65], Hsieh et al. [51], Islam et al. [66], He et al. [67], Kim et al. [68], and Das et al. [50].

**A. EXECUTION-TIME EVALUATION**

For the analysis of the protocols’ execution times, we added the execution times of the operations involved in the authentication phase of the schemes. We do not consider the operations used in the registration phase since the entities execute this phase only once.

For ECC-based protocols, we use an additive group $G$ with order $q$, generated by point $P$ over a non-singular elliptic curve $y^2 = x^3 + ax + b \mod p$. Parameters $p$ and $q$ are prime numbers and their sizes are 160 bits each. For bilinear pairing-based schemes, we use a bilinear pairing $e : G_1 \times G_1 \rightarrow G_2$. The additive group $G_1$ has order $q$. It is generated by point $P$ over the supersingular elliptic curve $y^2 = x^3 + 1 \mod p$. Parameters $p$ and $q$ are prime numbers. Their sizes are 512 bits and 160 bits, respectively [65], [70], [71].

The operations’ execution times were taken from the experimental results presented in [65], [70]. The experiment’s environment setup was a computer with an Intel i7-4770 processor, 3.40 GHz of clock frequency, 4 gigabytes of memory, and the Windows 7 operating system. The operations’ execution times were obtained from the cryptographic library “Multiprecision Integer and Rational Arithmetic Cryptographic Library (MIRACL)” [72].

We following describe the operations included in the evaluation. Let $T_{bp}$ be the running time of a bilinear pairing operation $e$. Let $T_{sm-bp}$ be the execution time of scalar multiplication $\tau P$, where $\tau \in Z_q^*$ and $P \in G_1$, related to bilinear pairing. Let $T_{pa-bp}$ be the running time of point addition $Q + R$, where $Q, R \in G_1$, related to bilinear pairing. Let $T_b$ be the execution time of a conventional hash function. Let $T_{s/d}$ be the running time of a symmetric key encryption or decryption function. Let $T_{mp}$ be the time to execute a hash-to-point function that maps strings to points in $G_1$. Let $T_{mul}$ be the execution time of a multiplication operation. Let $T_{sm-ecc}$ be the running time of scalar point multiplication $rP$, where $r \in Z_q^*$ and $P \in G$. Let $T_{pa-ecc}$ be the time to execute a point addition $Q + R$, where $Q, R \in G$.

In Table IV are listed the operations and their execution times. The running times of the xor and concatenation operations are not considered in the analysis because their
TABLE IV. Cryptographic operations’ execution times [65], [70].

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Operation</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>( T_{bp} )</td>
<td>Bilinear pairing.</td>
<td>4.2100 ms</td>
</tr>
<tr>
<td>( T_{sm-bp} )</td>
<td>Scalar multiplication, related to the bilinear pairing.</td>
<td>1.7090 ms</td>
</tr>
<tr>
<td>( T_{pa-bp} )</td>
<td>Point addition, related to the bilinear pairing.</td>
<td>0.0073 ms</td>
</tr>
<tr>
<td>( T_{h} )</td>
<td>Conventional hash function.</td>
<td>0.00010 ms</td>
</tr>
<tr>
<td>( T_{e/d} )</td>
<td>Symmetric key encryption or decryption function.</td>
<td>0.00020 ms</td>
</tr>
<tr>
<td>( T_{mt} )</td>
<td>Hash-to-point function.</td>
<td>4.30200 ms</td>
</tr>
<tr>
<td>( T_{mul} )</td>
<td>Point multiplication.</td>
<td>0.00200 ms</td>
</tr>
<tr>
<td>( T_{sm-ecc} )</td>
<td>Scalar point multiplication.</td>
<td>0.44200 ms</td>
</tr>
<tr>
<td>( T_{pa-ecc} )</td>
<td>Point addition.</td>
<td>0.00180 ms</td>
</tr>
</tbody>
</table>

TABLE V. Execution times for TLAP and related protocols.

<table>
<thead>
<tr>
<th>Protocol proposed by</th>
<th>Principal</th>
<th>Operations</th>
<th>Execution time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ying et al. [65]</td>
<td>User</td>
<td>( 4T_{sm-ecc} + 3T_{pa-ecc} + T_h )</td>
<td>1.7773 ms</td>
</tr>
<tr>
<td></td>
<td>Server</td>
<td>( 4T_{sm-ecc} + 3T_{pa-ecc} + 3T_h )</td>
<td>1.7719 ms</td>
</tr>
<tr>
<td>Hsieh et al. [51]</td>
<td>User</td>
<td>( 4T _{mt} + T_{sm-bp} + 5T_{pa-bp} + 8T_h )</td>
<td>16.2728 ms</td>
</tr>
<tr>
<td></td>
<td>Server</td>
<td>( 4T_{sm-bp} + 3T_{mul} )</td>
<td>21.2673 ms</td>
</tr>
<tr>
<td>Islam et al. [66]</td>
<td>User A</td>
<td>( 4T_{sm-ecc} + 2T_{pa-ecc} + 4T_h + 3T_{mul} )</td>
<td>1.7720 ms</td>
</tr>
<tr>
<td></td>
<td>User B</td>
<td>( 4T_{sm-ecc} + 2T_{pa-ecc} + 4T_h + 3T_{mul} )</td>
<td>1.7720 ms</td>
</tr>
<tr>
<td>He et al. [67]</td>
<td>Entity A</td>
<td>( 5T_{sm-bp} + 5T_{pa-bp} + 2T_h )</td>
<td>2.2156 ms</td>
</tr>
<tr>
<td></td>
<td>Entity B</td>
<td>( 6T_{sm-bp} + 5T_{pa-bp} + 3T_h )</td>
<td>2.2156 ms</td>
</tr>
<tr>
<td>Kim et al. [68]</td>
<td>Entity A</td>
<td>( 4T_{sm-bp} + 3T_{pa-bp} + 3T_h )</td>
<td>1.7736 ms</td>
</tr>
<tr>
<td></td>
<td>Entity B</td>
<td>( 4T_{sm-bp} + 3T_{pa-bp} + 3T_h )</td>
<td>1.7736 ms</td>
</tr>
<tr>
<td>Das et al. [50]</td>
<td>Device i</td>
<td>( T_{sm-ecc} + 5T_{pa-ecc} + 6T_h + 6T_{mul} )</td>
<td>3.1000 ms</td>
</tr>
<tr>
<td></td>
<td>Device j</td>
<td>( T_{sm-ecc} + 7T_{pa-ecc} + 6T_h + 6T_{mul} )</td>
<td>3.1000 ms</td>
</tr>
<tr>
<td>Us (TLAP)</td>
<td>Client a</td>
<td>( 3T_{sm-bp} + 5T_h + T_{e/d} )</td>
<td>1.3267 ms</td>
</tr>
<tr>
<td></td>
<td>AP b</td>
<td>( 3T_{sm-bp} + 5T_h + T_{e/d} )</td>
<td>1.3267 ms</td>
</tr>
</tbody>
</table>

The execution times for TLAP and related protocols are presented in Table V. As can be seen, TLAP has the lowest execution time. The proposal of Hsieh et al. has the highest. This is due to the use of bilinear pairing operations and hash-to-point functions. The protocols of Ying et al., Islam et al., and Kim et al. have very similar execution times.

B. COMMUNICATION-COST EVALUATION

For the analysis of the protocols’ communication cost, we followed the criteria described in [50], [73]. We added the size in bits of the messages transmitted by the entities during the authentication phase of the protocols. We do not consider the messages in the registration phase because the entities execute this phase only once, and the communication is performed in a secure channel.

Elements in \( G \) and \( G_1 \) have the following sizes based on the lengths of \( p \) and \( p \) specified in the subsection Execution-time evaluation. Points on an elliptic curve have the form \( P = (x_P, y_P) \), where \( x_P \) and \( y_P \) are the x and y coordinates, respectively. Thus, the size of an element in \( G \) is 160 x 2 = 320 bits. The security of a 160-bit ECC cryptosystem is the same as a 1024-bit RSA cryptosystem [71]. Finally, the size of an element in \( G_1 \) is 512 x 2 = 1024 bits.

Furthermore, the size of a hash function output is 160 bits when SHA-1 is used. Identities and random numbers are 160 bits long. Timestamps have a length of 32 bits [50], [73].

In Table VI the sizes of the different types of cryptographic data transmitted in the protocols’ messages are listed. In the proposal of Ying et al., the messages transmitted between user and server are \( \sigma_{ij}, D_{IDU_i}, A_{ij}^t, F_{u ij}, \sigma_{sj}, B_{sj}, ID_{S_j}, \) and \( F_{S_j} \). Where \( \sigma_{U_i}, \sigma_{S_j} \in Z_q^*, A_{ij}^t, F_{U_i}, B_{S_j}, F_{S_j} \in G, D_{IDU_i} \) is a hash-function output, and \( ID_{S_j} \) is an identity. Therefore, the communication cost of Ying et al.’s proposal is 1920 bits.

In the protocol of Hsieh et al., the messages transmitted between user and server are \( x_{Auth_i}, C_m, M_i, B_{ij}, R_i, Auth_{ij}, K_i, R_j, \) and \( Auth_{ij} \). Where \( x_{Auth_i}, C_m, M_i, B_{ij}, R_i, K_i, R_j \in G_1 \), and \( Auth_{ij}, Auth_{ij} \in Z^*_q \). Therefore, the communication cost of Hsieh et al.’s protocol is 7488 bits.

In the scheme proposed by Islam et al., the messages transmitted between the users are \( ID_A, T_A, R_A, S_A, ID_B, T_B, R_B, \) and \( S_B \). Where \( ID_A, ID_B, S_A, S_B \in Z^*_q \), \( TA, TB, RA, RB \in G \). Therefore, the communication cost of Islam et al.’s protocol is 1920 bits.

In the proposal of He et al., the messages transmitted between the entities are \( ID_A, R_A, ID_B, R_B, \) and \( T_B \). Where \( R_A, T_A, R_B, T_B \in G \), and \( ID_A \) and \( ID_B \) are identities. Therefore, the communication cost of He et al.’s protocol is 1600 bits.

In the protocol of Kim et al., the messages transmitted between the entities are \( ID_A, R_A, ID_B, R_B, T_B, \) and \( T_B \). Where \( R_A, T_A, R_B, T_B \in G \), and \( ID_A \) and \( ID_B \) are identities. Therefore, the communication cost of Kim et al.’s scheme is 1600 bits.

In the scheme proposed by Das et al., the messages transmitted between the smart devices are \( ID_i, A_i, c_i, T_i, z_i, R_i, \) and \( Q_i, ID_j, A_j, c_j, T_j, z_j, R_j, SKV_{ij}, Q_j, SKV_{ij}, \) and \( T_j^i \). Where \( ID_i \) and \( ID_j \) are identities, \( T_i, T_j, T_j^i \) are timestamps, \( c_i \) and \( c_j \) are certificates, \( z_i \) and \( z_j \) are signatures, \( SKV_{ij} \) and \( SKV_{ij}^* \) are hash-function outputs, and \( A_i, R_i, \) \( A_j, R_j, Q_j \in G \). Therefore, the communication cost of Das et al.’s proposal is 3296 bits.

In TLAP, the messages transmitted between the client and AP are \( T_1, D_1, D_3, D_4, T_2, D_5, D_7, T_3, \) and \( D_8 \). Where \( T_1, T_2, \) and \( T_3 \) are timestamps, \( D_1, D_3, D_4, D_7, \) and \( D_8 \) are hash-function outputs. \( D_3 \) is the result of the
The communication cost for TLAP and related protocols is presented in Table VII. As can be seen, TLAP has the lowest communication cost. The protocol of Hsieh et al. has the highest. The proposals of Ying et al. and Islam et al. have the same cost, as well as He et al. and Kim et al. have the same cost. The communication cost of these last two protocols can be considered low.

C. SECURITY COMPARISON

The security of TLAP and related protocols were analyzed. The articles in [65], [73], [75]–[79] were reviewed during the security analysis. The protocols’ security was analyzed concerning whether the protocol achieves the security properties anonymity, un-traceability, session key agreement, and forward secrecy. And whether the protocol resists potential attacks in WBANs such as off-line identity guessing, impersonation, replay, session key disclosure, insider, and MITM attacks. Table VIII presents a comparison of the security attributes achieved by the protocols. As can be seen, only TLAP achieves all the mentioned features. The protocol of Das et al. fulfills many security attributes. However, the scheme achieves them at a higher execution-time and communication-cost than TLAP.

The security of TLAP was analyzed using formal and informal methods. The formal approaches consisted of the AVISPA tool and BAN logic, which are well-known mechanisms to assess the security of authentication protocols. AVISPA tool considers a Dolev–Yao threat model, where an adversary can capture, reassemble, and alter transmitted messages. It evaluates if the protocol running under these conditions resists MITM and replay attacks. Using the AVISPA tool, TLAP was concluded as safe. BAN logic comprises a set of postulates and rules which allow verifying the trustworthiness of the transmitted data on an authentication protocol. Using BAN logic was demonstrated that TLAP achieves mutual authentication between the principals. An informal analysis of TLAP’s security was also performed. We showed TLAP achieves the security properties confidentiality, data integrity, client anonymity, perfect forward and backward secrecy, known session key security, mutual authentication, and key agreement. Also, the proposal is secure against potential attacks in WBANs such as tracing, off-line identity guessing, impersonation, injection, MITM, privileged insider, replay, known session-specific temporary information, DoS, de-synchronization, and key disclosure attacks.

As was presented in the Security requirements subsection, a protocol for TMISs should provide anonymity and un-traceability to ensure the privacy of the patients’ health data. As medical information is sensitive and intimate, its unauthorized disclosure or its malicious modification can have catastrophic effects in the patients’ life, from creating social problems for the patients to even endangering their lives.

In TLAP, the Client a’s public key is never sent in clear-text in the public channel to prevent tracing attacks. When AP b receives message $M_1$ from Client a, AP b obtains Client a’s public key from the NM. AP b can use TLAP or a traditional cryptosystem to communicate securely with the

<table>
<thead>
<tr>
<th>Protocol proposed by</th>
<th>Communication cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ying et al. [65]</td>
<td>1920 bits</td>
</tr>
<tr>
<td>Hsieh et al. [51]</td>
<td>1920 bits</td>
</tr>
<tr>
<td>Islam et al. [66]</td>
<td>1920 bits</td>
</tr>
<tr>
<td>He et al. [67]</td>
<td>1600 bits</td>
</tr>
<tr>
<td>Kim et al. [68]</td>
<td>1600 bits</td>
</tr>
<tr>
<td>Das et al. [50]</td>
<td>3296 bits</td>
</tr>
<tr>
<td>Us (TLAP)</td>
<td>1376 bits</td>
</tr>
</tbody>
</table>

TABLE VII. Communication cost for TLAP and related protocols.

<table>
<thead>
<tr>
<th>Protocol proposed by</th>
<th>SA1</th>
<th>SA2</th>
<th>SA3</th>
<th>SA4</th>
<th>SA5</th>
<th>SA6</th>
<th>SA7</th>
<th>SA8</th>
<th>SA9</th>
<th>SA10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ying et al. [65]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>Hsieh et al. [51]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>Islam et al. [66]</td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>He et al. [67]</td>
<td></td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Kim et al. [68]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Das et al. [50]</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Us (TLAP)</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>x</td>
</tr>
</tbody>
</table>

✓: The protocol supports the security attribute. x: The protocol does not support the security attribute.

SA1: Anonymity; SA2: Un-traceability; SA3: Session key agreement; SA4: Forward secrecy; SA5: Off-line identity guessing resistance; SA6: Impersonation attack resistance; SA7: Replay attack resistance; SA8: Session key disclosure resistance; SA9: Insider attack resistance; SA10: MITM attack resistance.

IX. DISCUSSION

In this article, the TLAP authentication protocol is proposed for WBANs in health-care applications. Considering that WBAN devices are limited in computing and battery resources, TLAP does not involve operations that use many computational resources such as security mechanisms based on PKI, bilinear pairing operations, and map-to-point hash functions. Instead, TLAP is based on ECC scalar point multiplication, symmetric key encryption, and the lightweight operations xor and conventional hash function. Therefore, TLAP’s execution time and communication cost are low. Further, the proposal does not require public-key certificates. It uses instead self-certified public keys. Thus, the overhead of generating, storing, verifying, and revoking certificates is avoided. TLAP prevents the key escrow problem by making the principals create their own private keys instead of entrusting this task to a PKG. This also prevents malicious PKGs from performing MITM attacks using the principals’ keys.
NM. Furthermore, AP b can have a cache list of clients that request services frequently. In a similar manner as the cache memory presented in [80]. Therefore, AP b does not need to communicate with the NM to obtain Client a’s public key every time AP b receives a request from Client a.

Even though the Client a’s self-certified public key contains the Client a’s identity (IDa), it includes the value in a non-invertible manner. In the registration phase, IDa and the random numbers r0 and r1 are hashed to construct the public key. a cannot invert the hash function to obtain IDa. Further, a cannot guess IDa because r0 and r1 are fresh, unpredictable, and unknown to a. If the lengths of IDa and r0 and r1 are n and m bits, respectively, the probability of a guessing the values at the same time approximates \(1/2^{n+2m}\). Guessing two or more secret parameters is not computationally feasible in polynomial time [60], [61].

Moreover, IDa is Client a’s identity only in the NM’s system. IDa itself does not contain information that could reveal who the human owner of the WBAN is. However, Client a’s IDa should be confidential to prevent a from tracking the user’s behavior patterns since a can use this information for malicious objectives or marketing purposes. TLAP keeps the clients requesting services confidential by not sending their IDa in clear-text in the public channel. In the authentication phase, the IDa is sent hashed to make it non-invertible. And encrypted with an ephemeral key only known by AP b; thus, only the legitimate AP b can decrypt it and know the client that requests the service. D1, D2, and D3 have the hash of IDa, and D3 the encryption of IDa. D3 is Client a’s ephemeral pseudonym, which is only valid in the current session.

Furthermore, TLAP prevents tracing attacks because the parties do not send constant values in the authentication phase. Since all messages contain ephemeral random numbers, there is no relation between messages of two authentication sessions. Therefore, a cannot use the transmitted data to trace Client a’s actions [81].

A comparison of the security attributes achieved by TLAP and existing similar protocols was performed. Table VIII shows the comparison of the security attributes satisfied by each protocol. As can be seen, TLAP achieves higher security than the other schemes. The proposal of Das et al. achieves many security properties. However, it requires more than double of execution-time and more than double of communication-cost than our proposal, as can be seen in Table V and Table VII, respectively.

Authentication protocols impact the entire data communication process concerning the response time and processor cycles needed for their execution. The protocol performance is especially relevant for applications that involve frequent communications of sensitive information. Quantifying the execution time, data transmission, and security level of protocols supports deciding the appropriate protocol for an application [82]. Thus, the performance of TLAP was analyzed in terms of execution time, communication cost, and security features, and it was contrasted with similar existing proposals.

We considered a 160-bit ECC in the execution time and communication cost analysis. Thus, the system has a security level comparable to 1024-bit RSA [71], with a considerably smaller key. Further, the execution of 160-bit ECC does not significantly increase the device duty cycle if the computation complexity and the volume of data transmitted and stored are reduced [83].

The execution times of TLAP and related protocols are shown in Table V. As can be seen, TLAP has the lowest execution time. TLAP uses 25.13% less execution time than Ying et al., 92.93% less than Hsieh et al., 25.13% less than Islam et al., 40.12% less than He et al., 25.20% less than Kim et al., and 57.20% less than Das et al.

Fig. 5 shows the execution times of the protocols with the increase in the number of users. The execution times increase linearly when the number of users grows. However, the rise of TLAP’s execution-time is slower than the other proposals, which can improve the efficiency of the authentications.

The communication cost of TLAP is lower than the compared protocols, as can be seen in Table VII. TLAP transmits 28.33% fewer bytes than Ying et al., 81.62% fewer bytes than Hsieh et al., 28.33% fewer bytes than Islam et al., 14% fewer bytes than He et al., 14% fewer bytes than Kim et al., and 58.25% fewer bytes than Das et al. Fig. 6 shows the communication costs of the protocols with the growth in the number of users. As can be seen, the communication cost of TLAP increases slower than the other schemes.

The low execution-time and low communication-cost of TLAP are important features to WBAN applications because they allow decreasing the energy consumption in WBAN devices. As some medical devices are implanted in the user’s body, their batteries are difficult to recharge or replace; thus, it is necessary to prolong their batteries’ life.
Communication cost (bytes)

<table>
<thead>
<tr>
<th>Scheme</th>
<th>Communication cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hsieh et al.</td>
<td>25000</td>
</tr>
<tr>
<td>Islam et al.</td>
<td>35000</td>
</tr>
<tr>
<td>He et al.</td>
<td>50000</td>
</tr>
<tr>
<td>Kim et al.</td>
<td>10000</td>
</tr>
<tr>
<td>Das et al.</td>
<td>20000</td>
</tr>
<tr>
<td>TLAP</td>
<td>50000</td>
</tr>
</tbody>
</table>

Health-care professionals to be confident that the transmitted time and communication cost than the related schemes. The achieves more security features, and it has lower execution and it resists potential attacks in WBANs. The security and was presented showing TLAP satisfies security requirements, Therefore, the key escrow problem is prevented. The security of TLAP was analyzed using the well-known formal methods AVISPA tool and BAN logic, which demonstrated that TLAP achieves mutual authentication and is secure against MITM and replay attacks. Additionally, an informal analysis was presented showing TLAP satisfies security requirements, and it resists potential attacks in WBANs. The security and performance of TLAP and similar existing protocols were analyzed and compared. The analysis showed that TLAP achieves more security features, and it has lower execution time and communication cost than the related schemes. The high security and performance of TLAP allow patients and health-care professionals to be confident that the transmitted medical information is disclosed only to authorized entities and is not maliciously altered during transmission.

REFERENCES


