ECC-based lightweight authentication and access control scheme for IoT E-healthcare

Hailong Yao · Qiao Yan · Xingbing Fu · Zhibin Zhang · Caihui Lan

Abstract
The E-healthcare system has a complex architecture, diverse business types, and sensitive data security. To meet the secure communication and access control requirements in the user–medical server, user–patient, patient–medical server, and other scenarios in the E-healthcare system, secure and efficient authenticated key agreement and access authorization scheme need to be studied. However, the existing multi-server solutions do not consider the authentication requirements of the Wireless Body Area Network (WBAN) and are not suitable for user–patient, patient–medical server scenarios; most of the existing WBAN authentication schemes are single-server type, which are difficult to meet the requirements of multi-server applications, and the study of user–patient real-time scenarios has not received due attention. This work first reveals the structural flaws and security vulnerabilities of the existing typical schemes and then proposes an authentication and access control architecture suitable for multiple scenarios of the E-healthcare system with separate management and business and designs a novel ECC-based multi-factor remote authentication and access control scheme for E-healthcare using physically unclonable function (PUF) and hash. Security analysis and efficiency analysis show that the new scheme has achieved improved functionality and higher security while maintaining low computational and communication overhead.

Keywords Authentication · Key agreement · Access control · Healthcare

1 Introduction
In the near future, the medical industry will incorporate more artificial intelligence, sensor technology, and other high technologies to create smart hospital systems, regional health systems, and home health systems. They will use advanced Internet of things technology, cloud computing technology, big data technology, and artificial intelligence technology to achieve seamless interaction between patients and medical staff, medical institutions, medical equipment and make medical services truly digital and intelligent. Through the wireless network, the portable personal digital assistant (PDA) is used to easily connect various diagnostic and therapeutic instruments, so that medical staff can grasp the patient’s medical record information and the latest diagnostic report at anytime and quickly formulate a diagnostic program anytime, anywhere; authorized medical staff and family members of patients can access the telemedicine server at anytime and any place to query medical image data and medical orders; the patient’s referral information and medical records can be accessed through medical networking at any hospital, and special groups such as chronic diseases, old and young patients, mental retardation, disability, and infectious diseases can be monitored and taken care of through the telemedicine system.

For secure communication and access control among all these entities, we need a secure mutual authenticated key agreement and access authorization mechanism which can provide authentication among body sensors and personal gateways, personal gateways and health servers, personal gateways and users (i.e., medical staff and family members of patients), and health servers and users and can provide authorization for users and patients to access medical servers, and users to access patient sensors. However, due to the complex network structure of E-healthcare system (the server
side is mostly secure and stable Ethernet, the user side is mostly WLAN or cellular mobile communication network, and the patient side is wireless sensor network, some nodes are resource-constrained devices (most medical servers are high-performance server cluster or cloud server, the user-side devices are mostly personal computers or mobile smart terminals, except for the relatively rich gateway on the patient side, the remaining sensors and other devices are cheap terminals with limited batteries, storage, and computing power), the interaction data involve individuals privacy (such as patient’s name, home address, medical records, blood test results, DNA sequence, and other sensitive data) and other characteristics, so existing authentication and authorization scheme cannot be directly applied to E-healthcare system.

1.1 Motivation

The drawbacks of existing scheme include two aspects: architecture flaws and security vulnerability.

(1) Session key initialization between users and patients requires the assistance of a particular medical server, which is not in line with the design concept of separation of management and application.

(2) Single server mode cannot meet the application needs of a multi-server environment (Kumari et al. 2020). Common multi-server authentication schemes (Amin et al. 2019; Barman et al. 2019; Feng et al. 2018; He and Wang 2015; Lwamo et al. 2019; Odelu et al. 2015; Qi et al. 2018; Roy et al. 2019; Yao et al. 2019a, b) can meet the authentication or authorization requirements of the user–server scenario, but no multiple solution is proposed, and many schemes (Amin et al. 2019; Lwamo et al. 2019; Roy et al. 2019) that do not use the public key system suffer from the vulnerability of anonymity (Wang et al. 2016, 2020b).

(3) There are fewer schemes for the patient–server scenario, and most existing schemes (Aghili et al. 2019; Amin et al. 2018; He et al. 2015; Kirsal Ever 2018; Wazid et al. 2018; Zhang et al. 2018) are in the WBAN-server mode.

(4) There are few schemes for the patient–user scenario. The only few schemes also adopt the patient–server–user mode, which does not meet the requirements of separation of management and business (Wazid et al. 2018; Fu et al. 2018).

(5) There are still some general security flaws in the existing schemes. Most schemes (Amin et al. 2019; Lwamo et al. 2019) that do not use public key cryptography suffer from the vulnerability of anonymity (Roy et al. 2019; Wang et al. 2020b). Some schemes (Feng et al. 2018; Lwamo et al. 2019) have lost their forward security due to ephemeral secrets being acquired by adversary (Yao et al. 2019b). Some schemes (Feng et al. 2018; Lwamo et al. 2019; Banerjee et al. 2019) are vulnerable to smart card loss attacks due to poor secret packaging in smart card, which can lead to offline dictionary attack, causing the schemes cannot resist user impersonation attack or device impersonation attack (Wang et al. 2020a; Yao et al. 2019b, 2020).

To overcome the above challenges, this work uses PUF and biohash based on ECC cryptography to propose a secure and efficient multi-server authentication and access control scheme for E-healthcare. This proposal can provide mutual authentication and access authorization for entities in the E-healthcare systems.

1.2 Our contributions

The contributions of this article are summarized below.

(1) We cryptanalyze existing authentication schemes such as LACO (Aghili et al. 2019), revealing the reasons why their anonymity and forward security are vulnerable and cannot resist user impersonation or device impersonation attacks.

(2) We first proposed a multiple solution architecture for authentication and authorization in user–server, patient–server, user–patient, and other scenarios in E-healthcare.

(3) Based on the above architecture, we combine PUF-based patient WBAN authentication with ECC-based remote multi-server authentication and use a hash function to design a remote authentication and access control scheme that integrates three factors of identity, password, and biometric, named SEMAS.

(4) Formal security proof, non-formal security analysis, comparative analysis of functional and security properties, comparative analysis of computing efficiency and communication efficiency are given.

1.3 Paper outline

The rest of this work is organized as follows. In Sect. 2, we briefly discuss the related work. Basic notations, ECC security assumptions, physically unclonable function, communication model, and threat model definition will be described in Sect. 3. The LACO is reviewed, and its weaknesses are analyzed in Sects. 4 and 5, respectively. We describe the details of our scheme in Sect. 6. The security analysis and performance evaluation will be given in Sects. 7 and 8, respectively. Finally, we present our conclusions in Sect. 9.

2 Literature review

Authentication and access control schemes can be classified into symmetric cryptography-based schemes and public...
key cryptography-based schemes according to the cryptography they rely on. Although symmetric cryptography-based schemes are generally computationally efficient, it is almost difficult to effectively achieve strong anonymity (Wang et al. 2020b; Yao et al. 2020). Therefore, authentication and access control schemes with privacy protection are usually designed based on public key cryptography. However, most public key cryptography-based schemes are difficult to apply to the IoT environment due to high overhead, such as RSA-based schemes (Lwamo et al. 2019; Dharminder et al. 2020), bilinear-pairing-based schemes (Amin and Biswas 2015; Nikravan and Reza 2020) and chaotic-maps-based schemes (Chatterjee et al. 2016; Roy et al. 2018). In the IoT scenario, the short key feature of ECC cryptography gives it an advantage in balancing resources and efficiency.

In 2010, Yang and Yang (2010) propose the first three-factor EDLP-based authenticated key exchange scheme. In the same year, Yoon and Yoo (2013) propose another EDLP-based three-factor authenticated key exchange scheme. However, He (2011) show that Yoon and Yoo’s scheme cannot resist insider attack and hardware factor loss attack and give an improvement (He and Wang 2015). In 2015, Odelu et al. (2015) show that He et al. scheme’s anonymity is vulnerable and cannot resist replay attack and user impersonation attack. Chuang and Chen (2014) also show the anonymity problem of Yoon–Yoo’s scheme and use a random number and hash function to construct a lightweight improvement scheme. In 2017, Kumari et al. show that Chuang et al.’s scheme cannot resist intermediate data attacks, user impersonation attack, and forward security attack and propose an improvement using digital signature (Kumari and Om 2017; Kumari et al. 2017). In 2018, Feng et al. (2018) show that Kumari et al.’s scheme (Kumari et al. 2017) is vulnerable to user anonymity and impersonation attacks and an improvement is given. However, Yao et al. show that Feng et al.’s scheme is vulnerable to anonymity and cannot resist ephemeral secrets leak attacks, and causing replay attacks and session key security attacks (Yao et al. 2019a). In 2018, Lwamo et al. (2019) find that Kumari-Om’s scheme (Kumari et al. 2017) used too many exponential operations, resulting in excessive computational overhead. They propose a new RSA-based remote authentication scheme for the single- and multi-server environments to achieve lower computational overhead and higher security. However, Yao et al. show that the anonymity of Lwamo et al.’s scheme is vulnerable and cannot resist hardware loss attack, so incurred offline dictionary attack and user impersonation attack (Yao et al. 2019b). In 2018, Zhang et al. (2018) propose a lightweight three-factor authenticated key agreement scheme for E-health systems to protect user privacy through the use of a dynamic authentication mechanism. In 2019, Aghili et al. (2019) show that Zhang et al.’s scheme suffers from several attacks including desynchronization attack, denial of service attack, and insider attacks and propose an improvement scheme named LACO.

Recently, we find that although LACO solves some of the security problems of Zhang et al.’s scheme and also considers the ownership transfer in access control, there are security vulnerability and algorithm errors. In 2020, Wazid et al. (2020) proposed a new lightweight authentication and real-time data access scheme for cloud-based IoT environments. In 2020, Alsahlani and Popa (2020) show that the Wazid et al. (2020) scheme lacks flexibility and has no access control mechanism. In 2021, Chaudhry et al. (2021) reveal that Wazid et al. (2020) cannot provide mutual authentication between system entities in the case of multiple registered users. To the best of our knowledge, most of the existing symmetric-cryptography-based lightweight authentication schemes are vulnerable to security (anonymity and forward security are fragile), and the structure is incomplete (lack of access control mechanism, poor support for multiple business scenarios) and cannot meet the requirements of identity authentication, authorization, and key exchange in IoT E-healthcare environments.

3 Preliminaries and background

In this section, we describe the preliminaries which is necessary to understand the rest of this work.

3.1 Notation

Notations used in this paper and their descriptions are shown in Table 1.

3.2 EDLP and ECDH

The elliptic curve over the finite field \( F_p \) is a finite cyclic group \( G \) satisfying \( y^2 = x^3 + ax + b \mod p \) and containing the infinity point \( O \), where \( a, b \in F_p \) and \( 4a^3 + 27b^2 \neq 0 \mod (p) \) (Koblitz 1987). There are two operations of addition and scalar multiplication on \( G \), and the scalar multiplication is defined as the same point accumulation. The cryptosystem constructed using the elliptic curve discrete logarithm problem (EDLP) and the elliptic curve Diffie–Hellman problem (ECDH) is widely used in security protocols. The security assumptions of the EDLP and ECDH are given by the following two lemmas, for any probability polynomial time (PPT) adversary \( \mathcal{A} \):

**Definition 3.1** EDLP Security Assumption: Given \( k \in Z_p \) and \( P \in G \), it is easy to calculate \( Q = kP \in G \), but given \( P, Q \in G \), the advantage \( Adv_{\text{EDLP}}(\mathcal{A}) \) for solving \( k \in Z_p \) is bounded by the negligible probability \( negl(\lambda) \).

**Definition 3.2** ECDH Security Assumption: Given \( P, yP \) and \( xP \in G \), but unknown \( x \) or \( y \in Z_p \), the advantage
Table 1 Notations used in this paper

| Notations                        | Descriptions                                      |
|----------------------------------|---------------------------------------------------|
| ID<sub>i</sub> / PW<sub>i</sub> / Bi | i<sup>th</sup> user’s ID/Password/Biometric       |
| Cr<sub>i</sub> / UD<sub>i</sub>    | i<sup>th</sup> user’s Credential/Device           |
| ID<sub>j</sub> / Cr<sub>j</sub>    | j<sup>th</sup> server’s ID/Credential              |
| ID<sub>k</sub> / PW<sub>k</sub>    | k<sup>th</sup> patient’s ID/Password               |
| B<sub>k</sub> / Cr<sub>k</sub>     | k<sup>th</sup> patient’s Biometric/Credential      |
| ID<sub>l</sub>                   | l<sup>th</sup> device of k<sup>th</sup> patient   |
| R = PUF<sub>i</sub> (C)          | Device’s physically unclonable function           |
| RA / pk / sk                     | Registration center/Public and secret key          |
| L<sub>P</sub> / L<sub>U</sub> / L<sub>S</sub> | Registry of patient, user, and server            |
| h (·)                            | Cryptographic hash algorithm                       |
| h<sub>b</sub> (·)                | Biohash algorithm                                  |
| HD (·) ≤ δ                       | Hamming distance and threshold                     |
| Pr [·]                           | Probability function                               |
| negl (λ)                         | Negl-polynomial of security parameter λ            |
| D<sub>pwd</sub> / D<sub>H</sub>   | Distribution of password and hash value           |
| T<sub>i</sub> / Δ<sub>T</sub> / ΔL | i<sup>th</sup> Timestamp/Time threshold/TTL        |
| ⊕ / ⊥                            | XOR operator/concatenation operator                |
| ← / ⊥                            | Normal output/Abnormal output                      |
| $\$                          | Random sampling from the distribution              |

Adv<sub>EC</sub>DH (A) for solving $xyP \in G$ is bounded by the negligible probability $\text{negl} (\lambda)$.

3.3 Physically unclonable function

A physically unclonable function is a physical circuit that maps unique challenge $C$ to unique response $R$ based on the random variations introduced by the chip manufacturing process (Böhnm and Hofer 2012). The $R = \text{PUF}_l (C)$ of device $l$ is correct if:

1. $\Pr \left[ \text{HD} \left( \text{PUF}_l (C_i), \text{PUF}_l (C_j) \right) \leq i, j \leq n, i \neq j > \delta \right] < \text{negl} (\lambda)$;
2. For any $\text{PUF}_l (\cdot)$ and $\text{PUF}_l (\cdot)$,$\Pr \left[ \text{HD} \left( \text{PUF}_l (C_i), \text{PUF}_l (C_j) \right) \leq i, j \leq n, i \neq j \leq \delta \right] \geq 1 - \text{negl} (\lambda)$.

The $R = \text{PUF}_l (C)$ of device $l$ is secure if:

1. $\Pr \left[ \text{HD} \left( \text{PUF}_l (C_i), \text{PUF}_l (C_j) \right) \leq i, j \leq n, i \neq j \leq \epsilon \right] > 1 - \text{negl} (\lambda)$, which means that the min-entropy of the $\text{PUF}_l (\cdot)$ output is always larger than $\epsilon$ with high probability.

3.4 Communication model

In a multi-server scenario, medical servers, patients, and users need to register with a registration authority (RA). The local RA is responsible for the management and access authorization of servers, users, and patients in the region, and the central RA (CRA) is responsible for the management of region RA and the authentication and authorization between regions. Medical servers such as Electronic Medical Records (EMR) and Hospital Information Systems (HIS) work in central computer rooms with relatively high security. Users such as medical staff, academics, and patients in wards, homes, jobs, and streets need to access the medical server or access each other through the Internet. As shown in Fig. 1, patients and users can access the medical servers after the authentication and authorization obtained by the RA, and users can access the patient’s sensors after the authentication and authorization obtained by the RA.

3.5 Threat model

According to the widely accepted Dolev–Yao threat model (Dolev and Yao 1983) and the Canetti–Krawczyk adversary model (Canetti and Krawczyk 2002), the adversary $A$ attacking E-healthcare multi-server scheme has the ability to fully control the channel and get ephemeral secrets of the session. Adversary capabilities include:

1. $A$ can interfere with communication between entities by means of interception, modification, deletion, insertion, etc.
2. Medical server, patient gateway, and sensor are unreliable, and $A$ can learn long-term secrets from the captured devices.
3. $A$ has the ability to obtain ephemeral secrets of the incompletely corrupted object.
4. All servers are honest and curious.
Fig. 2 Registration phase of LACO scheme

| Patient’s sensor[NULL] | Medical server[sk] |
|------------------------|-------------------|
| registration request   |                   |
| selects device identity IDi, \( C_{r1} \leftarrow h(ID_i||sk) \) | \( ID_i \) into IDi’s memory and issue it |
| writes \{ IDi, C_{r1} \} into IDi’s memory and issue it | |

| User(IDi, PWi, Bi) | Medical server[sk] |
|-------------------|-------------------|
| selects identity IDi |                   |
| sends IDi to IDj    |                   |
| if IDi is valid     |                   |
| generates \( r_s \) |                   |
| \( X_{0i}, Z_{0i} \leftarrow NULL \) |                   |
| \( X_{1i} \leftarrow h(UD_i||ID_i||r_s) \) |                   |
| \( Y_{1i} \leftarrow h(X_{0i}||sk) \) |                   |
| \( Z_{1i} \leftarrow h(X_{0i}||Y_{0i}||A_i) \) |                   |
| writes \{ X_{0i}, Z_{0i}, X_{1i}, Z_{1i} \} into \( R_i \) |                   |
| writes \{ X_{1i}, Y_{1i}, Z_{1i}, h(\cdot) \} into smart card \( UD_i \) and issues it to IDi | |

\[
A_{1i} \leftarrow h_{b}(B_{i}) \oplus h(PW_{i}||ID_{i}) \]
\[
B_{1i} \leftarrow Y_{1i} \oplus h(ID_{i}||PW_{i}||h_{b}(B_{i})) \]
\[
flag \leftarrow 0 \]
\[
\mathrm{writes \{ A_{1i}, B_{1i}, \text{flag} \} \mathrm{\ into \ UD_{i} \ and \ deletes \ Y_{1i}} \]

4 Review of LACO scheme

In order to facilitate the understanding of the subsequent cryptanalysis of LACO, in this section we briefly review the registration and authentication process of it Aghili et al. (2019).

4.1 Registration

As shown in Fig. 2, when the LACO system is initialized, the server generates system parameters and issues written secrets sensor to the patient. When the user registers, the server issues the smart card for subsequent authentication.

4.2 Authentication and session key agreement

As shown in Fig. 3, LACO needs to perform smart card login authentication locally before initiating remote authentication and then sends an authentication request to the server after login. If the authentication passes, the server forwards the relevant information to the sensor. If the authentication passes, the sensor calculates the session key and directly sends an authenticated key agreement request to the user. If the authentication passes, the user calculates the session key.

5 Cryptanalysis of LACO scheme

The drawbacks of LACO scheme include three aspects: architecture flaws, fatal algorithm error, and security vulnerability.

5.1 The architecture flaws of LACO

A lame system architecture does not meet the needs of future E-healthcare applications.

(1) Session key initialization between medical staff and patients requires the assistance of a particular medical server, which is not in line with the design concept of separation of management and application.

(2) Single-server mode cannot meet the application needs of a multi-server environment. And dynamic updates and revocations of medical staff, patients, and medical servers are not considered.

5.2 Fatal algorithm error in LACO

There is fatal algorithm error in the LACO scheme, causing the protocol to fail to run as expected. In Step2, the server needs to find the \( \{ X_{n_i}, Z_{n_i} \} \) that satisfies \( h_3 = X_{n_i} || Z_{n_i} \) or \( h_3 = h(r_i || X_{n_i} || Y_{n_i}) || h(r_i || Y_{n_i} || Z_{n_i}) \) from the user registration information, calculates \( K_u' = h_1 \oplus h(X_{n_i} || Y_{n_i} || T_i) \) and \( ID_j = h_2 \oplus h(X_{n_i} || Y_{n_i} || Z_{n_i} || T_i) \) to verify whether \( h_4 \leftarrow h(h_1 || h_2 || h_3 || K_u || ID_j || T_i || r_i) \) is true and then achieves authentication of IDj. However, IDj and IDi updated

\[
X_{(n+1)i} = h \left( r_i || X_{ni} \right) \oplus r_i \oplus Y_{ni}' \]
\[
Z_{(n+1)i} = h \left( Y_{ni}' || X_{ni} \right) \oplus A_i \]

in Step4 and Step5, respectively. However, \( B_{ni} \) has not been updated in the user’s smart card, which means that \( ID_i \) calculates

\[
h_3^{n+1} = h \left( r_i || X_{(n+1)i} || Y_{ni}' \right) || h \left( r_i || Y_{ni}' || Z_{(n+1)i} \right) \]
in $n+1$ rounds because $Y'_{ni} = B_{ni} \oplus h\left(ID_i^r \|| PW_i^r \|| h_b\left(B_i^r\right)\right)$ and $B_{ni}$ is still the old one. However, the server calculates

$$h_3^{n+1} = h\left(r_i \|| X_{(n+1)i} \|| Y'_{(n+1)i}\right) \| h\left(r_i \|| Y'_{(n+1)i} \|| Z_{(n+1)i}\right)$$

because it calculates

$$Y'_{(n+1)i} = h\left(X_{(n+1)i}\|| sk\right).$$

Obviously,

$$h\left(r_i \|| X_{(n+1)i} \|| Y_{ni}\right) \| h\left(r_i \|| Y'_{ni} \|| Z_{(n+1)i}\right) \neq h\left(r_i \|| X_{(n+1)i} \|| Y'_{(n+1)i}\right) \| h\left(r_i \|| Y'_{(n+1)i} \|| Z_{(n+1)i}\right)$$

so the protocol is aborted here.

### 5.3 The security drawbacks of LACO

In addition to architectural flaws and algorithm error, LACO also has security flaws such as lack of session key privacy, cannot resist user impersonation attack and multi-factor security, and forward security vulnerability.

1. **Lack of session key privacy:** During the authentication and key agreement phase of LACO, the server is able to calculate the session key $ss_s = h\left(A_i \|| ID_i \|| K'_u \|| K'_p\right)$ between the user and the patient.

2. **Cannot resist user impersonation attack:** If the adversary $A$ obtains the secret $(ID_i, Cr_i)$ in the sensor’s
memory, she/he can bypass the server authentication, impersonating the server to forge $M_2^*$ to pass the ID$_i$ authentication and establish a session with it. Details are as follows:

**Step1:** $A$ generates $A_i^*$ and $K_u^*$;

**Step2:** $A$ calculates

$$h_5^* = A_i^* \oplus h (C_r || T_2),$$

$$h_6^* = A_i^* \oplus K_u^*,$$

$$h_7^* = h (A_i^* || ID_l || K_u^* || T_2),$$

$$M_2^* = \{h_5^*, h_6^*, h_7^*, T_2\},$$

and sends $M_2^*$ to ID$_l$;

**Step3:** If $T_3 - T_2 \leq \Delta T$, ID$_l$ calculates

$$A_i^* = h_5^* \oplus h (C_r || T_2),$$

$$K_u^* = A_i^* \oplus h_6^*,$$

and if $h_7^* = h (A_i^* || ID_l || K_u^* || T_2)$ is true, then generates $K_p$ and calculates

$$s_{s_p} = h (A_i^* || ID_l || K_u^* || K_p),$$

$$h_8 = h (s_{s_p} || C_r || T_3),$$

$$h_9 = K_u^* \oplus P,,$$

$$M_3 \leftarrow \{h_8, h_9, T_3\},$$

and sends $M_3$ to $A$;

**Step4:** $A$ calculates $K_p = h_9 \oplus K_u^*$ and $s_{s_A} = h (A_i^* || ID_l || K_u^* || K_p)$ after receiving $M_3$; A session between $A$ and the patient is established.

(3) **Multi-factor security vulnerability:** When $A$ knows the biometric $B_i$ and the smart card secret $\{A_{i1}, B_{i1}, X_{ni}, Y_{ni}\}$, although LACO has anonymity, since the user ID and password are low-entropy short strings, the probability that $A$ guesses the user password 100 times is $32\% - 73\%$ (Wang et al. 2016). When $A$ knows the user password $PW_i$ and the smart card secrets $\{A_{i1}, B_{i1}, X_{ni}, Y_{ni}\}$, $A$ can use a center search attack to derive the user’s biometric information (Simoen et al. 2012).

(4) **Forward security vulnerability:** Once the sensor’s secret information is leaked, $A$ will be able to derive the session key between the user and the patient from the captured $M_3$ and $M_4$. Details are as follows:

**Step1:** $A$ calculates $A_i = h_5 \oplus h (C_r || T_2)$, $K_u = A_i \oplus h_6$, $K_p = K_u \oplus h_9$; and $M_2^* = \{h_5^*, h_6^*, h_7^*, T_2\}$, and sends $M_2^*$ to ID$_l$;

**Step2:** If $h_7^* = h (A_i || ID_l || K_u || T_2)$ and $h_8 = h (s_{s_p} || C_r || T_3)$, there must be $s_{s_A} = h (A_i || ID_l || K_u || K_p)$.

### 6 Proposed scheme

To overcome the security architecture flaws and security drawbacks of previous authentication protocols such as the LACO (Aghili et al. 2019) adopted for E-health systems, we propose a secure and efficient protocol called SEMAS. In addition to providing preserving-privacy mutual authentication, key agreement, and access control, resisting known Internet attacks, the proposal also meets the authentication and access control requirements of the E-healthcare multi-server scenario.

The proposed scheme consists of six important phases: Initialization, Registration, Authentication and Key Agreement, Password Update and Ownership Transfer.

#### 6.1 Initialization

$RA$ initializes the system parameters, it selects a finite field $F_p$ with a large prime $p$ as the order and defines an elliptic curve $E_p$ over it, then selects an additive group $G$ with order $q$ and generator $P$ over $E_p$, and then selects the system private key $sk \in F_p$ and computes the public key $PK = skP$, and finally, $RA$ selects the secure hash algorithm $h()$, the biohash algorithm $h_b()$ and physically unclonable function algorithm $PUF()$ and publishes the public parameters {$P$, $PK$, $E_p$, $h()$, $h_b()$}.

#### 6.2 Registration

As shown in Fig. 4, during the registration phase, medical servers, users, and patients need to register with the RA in a secure manner. Details are as follows:

##### 6.2.1 Medical server registration

1. Server selects ID $ID_j$, and sends tuple {ID$_j$} to RA.
2. After $RA$ verifies that ID$_j$ is valid, it selects random number $r_j$, calculates credential $Cr_j$ and sends tuple {Cr$_j$} to ID$_j$, and writes {ID$_j$, $r_j$} to the server registration list $L_S$.
3. ID$_j$ writes {ID$_j$, Cr$_j$} to its memory.

##### 6.2.2 User registration

1. User selects ID $ID_i$ and password $PW_i$, generates biometric $B_i$, calculates $\alpha_i$, $\beta_i$ and sends {ID$_i$, $\alpha_i$, $\beta_i$} to RA.
2. After $RA$ verifies that ID$_i$ is valid, it selects random number $r_i$, calculates credential $Cr_i$, $\eta_i$ and $\gamma_i$, returns the message of successful registration, and writes {ID$_i$, $r_i$, $\eta_i$, $\gamma_i$} to the user registration list $L_U$.
6.2.3 Patient registration

(1) Patient selects ID \( ID_k \) and password \( PW_k \), generates biometric \( B_k \), calculates \( \alpha_k \), \( \beta_k \) and sends tuple \( \{ID_k, \alpha_k, \beta_k\} \) to RA.

(2) After RA verifies that \( ID_k \) is valid, it selects random number \( r_k \) and calculates credential \( Cr_k \), \( \eta_k \) and \( \gamma_k \); RA selects sensor \( ID_l \) according to the needs of \( ID_k \), generates random number \( C_l \) and writes \( \{h_b(\cdot), PUF(\cdot)\} \) to \( ID_l \)'s memory.

(3) \( ID_l \) calculates \( R_l = h_b(PUF(C_l)) \) and \( \alpha_l = R_l \oplus ID_l \) and inserts \( \alpha_l \) into \( ID_l \)'s memory and issues it to \( ID_k \).

(4) RA writes \( \{ID_k, r_k, \eta_k, \gamma_k, [ID_l]\} \) to the patient registration list \( L_P \) and sends tuple \( \{Cr_k, [ID_l, R_l, C_l]\} \) to \( ID_k \).

(5) Patient gateway \( ID_k \) calculates \( \kappa_k \), \( \beta_l \) and \( \gamma_l \) and writes \( \{\kappa_k, \{ID_l, \beta_l, \gamma_l\}\} \) to its memory.

6.3 Authentication and session key agreement

As shown in Figs. 5 and 6, during the authentication and key agreement phase, users and servers, users and patients can achieve authenticated key agreement and access authorization under RA authentication and authorization. The patient–server authentication is similar to the user–server and will not be repeated here. The process of user \( ID_i \) and patient \( ID_k \)'s sensor \( ID_l \) mutual authentication and establishing a secure session is as follows:

(1) User inputs \( ID_i \) and password \( PW_i' \), generates biometric \( B_i' \) and calculates \( \beta_i' \); user selects random number \( r_i \) and calculates \( A_i', A_i'^* \), \( h_{14}, h_{15}, h_{16}, h_{17} \) and \( h_{18} \) and sends tuple \( \{h_{14}, h_{15}, h_{16}, h_{17}, h_{18}, T_{U}\} \) to patient ID \( ID_k \).

(2) After \( ID_k \) verifies that timestamp is valid, she/he inputs \( ID_k \) and password \( PW_k \), generates biometric \( B_k' \), and calculates \( \beta_k' \); \( ID_k \) selects random number \( r_k \), and calculates credential \( A_k, A_k'^* \), \( h_{19}, h_{20}, h_{21}, h_{22} \) and sends tuple \( \{h_{14}, h_{15}, h_{16}, h_{17}, h_{18}, h_{20}, h_{21}, h_{22}, T_{U}\} \) to RA to request authentication.

(3) After RA verifies that timestamp is valid, it calculates \( A_k^* = skh_{19} \) and \( ID_k' = h_{20} \oplus h\left(\left(A_k^*\right)_x\right)\), and if searching for \( ID_k \) in patient registration list \( L_P \) is false, abort the protocol, else if \( h_{22} = h\left(ID_k||h_{18}||h_{19}||h_{20}||h_{21}||T_U\right) \) is false, abort the protocol, else calculates \( \beta_k = h_{21} \oplus h\left(\left(A_k^*\right)_x\right)\), \( Cr_k \) and \( \beta_k = \gamma_k \); if \( H\left(\beta_k, \beta_k'\right) \leq \delta \) is false, abort the protocol, else calculates \( A_i^* = skh_{14} \) and \( ID_i' = h_{15} \oplus h\left(\left(A_i^*\right)_x\right)\), and if searching for \( ID_i \) in user registration list \( L_U \) is false, abort the protocol, else if \( h_{18} = h\left(ID_i||ID_k||h_{14}||h_{15}||h_{16}||h_{17}||T_U\right) \) is false,
**Fig. 5** User–server authentication phase of our scheme

| Step | Description |
|------|-------------|
| **Step1** | User (ID$_{u}$, PW$_{u}$, B$_{u}$, γ$_{u}$) starts the authentication process. |
| | inputs ID$_{u}$, PW$_{u}$, B$_{u}$ |
| | β$_{u}$ ← h$_{b}$ (B$_{u}$) + h (ID$_{u}$) |
| | r$_{1}$ ← Z$_{p}$, A$_{u}$ ← r$_{1}$P, h$_{0}$ ← A$_{u}$, A*$_{u}$ ← r$_{1}$PK = (A*$_{u}$)$_{x}$, (A*$_{u}$)$_{y}$ |
| | h$_{1}$ ← ID$_{u}$ ⊕ h ([A*$_{u}$]$_{x}$) |
| | h$_{2}$ ← β$_{u}$ ⊕ (A*$_{u}$)$_{y}$ |
| | h$_{3}$ ← h (ID$_{u}$) h$_{0}$ h$_{1}$ T$_{1}$ |
| | M$_{1}$ = {h$_{0}$, h$_{1}$, h$_{2}$, h$_{3}$, T$_{1}$} |
| | sends M$_{1}$ to ID$_{d}$ |
| **Step2** | If T$_{2}$ - T$_{1}$ ≤ ΔT |
| | r$_{2}$ ← Z$_{p}$, A$_{d}$ ← r$_{2}$P, h$_{4}$ ← A$_{d}$ |
| | A*$_{d}$ ← r$_{2}$PK = (A*$_{d}$)$_{y}$ |
| | h$_{5}$ ← ID$_{d}$ h$_{3}$ h$_{4}$ T$_{2}$ |
| | h$_{6}$ ← h (ID$_{d}$) h$_{0}$ h$_{1}$ T$_{2}$ |
| | M$_{2}$ = {h$_{0}$, h$_{1}$, h$_{2}$, h$_{4}$, h$_{5}$, T$_{2}$} |
| | sends M$_{2}$ to RA |
| **Step3** | If T$_{3}$ - T$_{1}$ ≤ ΔT |
| | A*$_{d}$ ← skA$_{d}$ = (A*$_{d}$)$_{x}$, (A*$_{d}$)$_{y}$ |
| | ID$_{d}$' = h$_{b}$ h (A*$_{d}$)$_{y}$ |
| | search for ID$_{d}$' in LG |
| | if ID$_{d}$' = ID$_{d}$ |
| | Cr$_{d}$ ← h (ID$_{d}$) r$_{k}$ |
| | h$_{6}$ ← h (ID$_{d}$) h$_{0}$ h$_{1}$ T$_{2}$ |
| | A*$_{d}$ ← skA$_{d}$ = (A*$_{d}$)$_{x}$, (A*$_{d}$)$_{y}$ |
| | ID$_{d}$' = h$_{b}$ h (A*$_{d}$)$_{y}$ |
| | search for ID$_{d}$' in LG |
| | if ID$_{d}$' = ID$_{d}$ |
| | Cr$_{d}$ ← h (ID$_{d}$) r$_{k}$ |
| | β$_{d}$ ← γ$_{d}$ ⊕ Cr$_{d}$ |
| | if HD (β$_{d}$, β$_{d}'$) ≤ δ and |
| | if ID$_{d}$' in AL$_{d}$ |
| | r$_{3}$ ← Z$_{p}$, α$_{d}$ ← r$_{3}$P, h$_{0}$ |
| | h$_{7}$ ← h (ID$_{d}$) h$_{0}$ h$_{1}$ T$_{2}$ |
| | h$_{8}$ ← r$_{d}$ ⊕ h (A*$_{d}$)$_{y}$ |
| | h$_{9}$ ← h (h$_{0}$ h$_{9}$) (A*$_{d}$)$_{y}$ |
| | s$_{d}$ ← h (ID$_{d}$) r$_{d}$ h$_{0}$ |
| | s$_{d}$使之 = h (ID$_{d}$) r$_{d}$ h$_{0}$ |
| | sets T$_{d}$ = T$_{4}$ and writes |
| | {A$_{d}$, ID$_{d}$, T$_{1}$, r$_{1}$} to cache |
| | h$_{10}$ ← h (h$_{0}$ h$_{9}$) (A*$_{d}$)$_{y}$ |
| | h$_{11}$ ← h (h$_{0}$ h$_{9}$) (A*$_{d}$)$_{y}$ |
| | M$_{3}$ = {h$_{9}$, h$_{10}$, h$_{11}$, T$_{3}$} |
| | sends M$_{3}$ to ID$_{d}$ |
| **Step4** | If T$_{4}$ - T$_{3}$ ≤ ΔT |
| | ID$_{d}$' ← h$_{b}$ h (A*$_{d}$)$_{y}$ |
| | r$_{d}$ ← h$_{b}$ h (A*$_{d}$)$_{y}$ |
| | h$_{7}$ ← h (ID$_{d}$) h$_{0}$ h$_{1}$ T$_{2}$ |
| | h$_{8}$ ← r$_{d}$ ⊕ h (A*$_{d}$)$_{y}$ |
| | h$_{9}$ ← h (h$_{0}$ h$_{9}$) (A*$_{d}$)$_{y}$ |
| | s$_{d}$使之 = h (ID$_{d}$) r$_{d}$ h$_{0}$ |
| | sets T$_{d}$ = T$_{4}$ and writes |
| | {A$_{d}$, ID$_{d}$, T$_{1}$, r$_{1}$} to cache |
| | h$_{10}$ ← h (h$_{0}$ h$_{9}$) (A*$_{d}$)$_{y}$ |
| | h$_{11}$ ← h (h$_{0}$ h$_{9}$) (A*$_{d}$)$_{y}$ |
| | M$_{4}$ = {h$_{4}$, h$_{9}$, h$_{11}$, T$_{2}$} |
| | sends M$_{4}$ to ID$_{d}$ |
| **Step5** | If T$_{5}$ - T$_{4}$ ≤ ΔT |
| | ID$_{d}$' ← h$_{b}$ h (A*$_{d}$)$_{y}$ |
| | h$_{7}$ ← h (ID$_{d}$) h$_{0}$ h$_{1}$ T$_{2}$ |
| | h$_{8}$ ← r$_{d}$ ⊕ h (A*$_{d}$)$_{y}$ |
| | h$_{9}$ ← h (h$_{0}$ h$_{9}$) (A*$_{d}$)$_{y}$ |
| | s$_{d}$使之 = h (ID$_{d}$) r$_{d}$ h$_{0}$ |
| | sets T$_{d}$ = T$_{4}$ and writes |
| | {A$_{d}$, ID$_{d}$, T$_{1}$, r$_{1}$} to cache |
| | h$_{10}$ ← h (h$_{0}$ h$_{9}$) (A*$_{d}$)$_{y}$ |
| | h$_{11}$ ← h (h$_{0}$ h$_{9}$) (A*$_{d}$)$_{y}$ |
| | M$_{4}$ = {h$_{4}$, h$_{9}$, h$_{11}$, T$_{2}$} |
| | sends M$_{4}$ to ID$_{d}$ |
| **Step6** | If T$_{6}$ - T$_{5}$ ≤ ΔT |
| | if h$_{13}$ = h (r$_{13}$) ss$_{ij}$ T$_{5}$ |
| | session key is established |
| | ID$_{K}$ calculates h$_{29}$ = h (ID$_{k}$) (C$_{l}$ || R$_{l}$ || T$_{9}$) and sends tuple (C$_{l}$, h$_{29}$, T$_{9}$) to sensor ID$_{K}$ |

(4) After patient gateway ID$_{K}$ verifies that timestamp is valid, it derives ID$_{d}$ and ID$_{d}$ from h$_{23}$ and h$_{24}$ and calculates Cr$_{d}$ and derives R$_{l}$ and C$_{l}$ from β$_{d}$ and γ$_{d}$, respectively; ID$_{K}$ calculates h$_{29}$ = h (ID$_{d}$) (C$_{l}$ || R$_{l}$ || T$_{9}$) and sends tuple (C$_{l}$, h$_{29}$, T$_{9}$) to sensor ID$_{K}$.

(5) After ID$_{d}$ verifies that timestamp is valid, it calculates R$_{d}$ = h$_{b}$ (PUF$_{d}$ (C$_{l}$)), if h$_{29}$ = h (R$_{d}$ || α$_{l}$) (C$_{l}$ || R$_{l}$ || T$_{9}$) is false, abort the protocol, else calculates session key s$_{d}$ = h (R$_{d}$ || T$_{10}$) between ID$_{d}$ and ID$_{K}$ and calculates R$_{d}$ = h$_{b}$ (PUF$_{d}$ (C$_{l}$ || T$_{10}$)) and updates α$_{l}$ = α$_{l}$ || R$_{d}$ || α$_{l}$ and calculates s$_{d}$ and sends tuple (R$_{d}$ || R$_{d}$ || C$_{l}$ || α$_{l}$ || R$_{d}$ || T$_{10}$) to ID$_{K}$.

(6) After verifying that timestamp is valid, ID$_{K}$ calculates session key s$_{d}$ = h (R$_{d}$ || T$_{10}$) between ID$_{K}$ and ID$_{d}$ and updates β$_{d}$ = R$_{d}$ || R$_{d}$ || C$_{l}$ || R$_{d}$ || γ$_{d}$ = h (C$_{l}$ || T$_{9}$) || Cr$_{d}$.
### Step 1

\[
\begin{align*}
\beta_1 &\leftarrow h_5 \left( B_i^t \right) \oplus h \left( ID_i^t \| PW_i \right) \\
\delta &\leftarrow h_6 \left( B_k^t \right) \oplus h \left( ID_k^t \| PW_k \right) \\
A'_{ik} &\leftarrow r_{ik} \cdot PK = \left( A'_{ik} \right)_x \oplus \left( A'_{ik} \right)_y \\
h_{15} &\leftarrow ID_i^t \oplus h \left( \left( A'_{ik} \right)_x \left( A'_{ik} \right)_y \right)
\end{align*}
\]

### Step 2 if \( T_r - T_b \leq \Delta T \)

\[
\begin{align*}
\beta_2 &\leftarrow h_5 \left( B_k^t \right) \oplus h \left( ID_k^t \| PW_k \right) \\
\delta &\leftarrow h_6 \left( ID_i^t \| PW_i \right) \\
A'_{lk} &\leftarrow r_{lk} \cdot PK = \left( A'_{lk} \right)_x \oplus \left( A'_{lk} \right)_y \\
h_{22} &\leftarrow ID_k \oplus h \left( \left( A'_{lk} \right)_x \left( A'_{lk} \right)_y \right) \\
\beta_5 &\leftarrow h_5 \left( ID_i^t \| PW_i \right) \\
h_{23} &\leftarrow h \left( ID_i^t \| PW_i \right) \oplus h \left( ID_k^t \| PW_k \right)
\end{align*}
\]

### Step 3 if \( T_b - T_r \leq \Delta T \)

\[
\begin{align*}
A'_{ik} &\leftarrow s_k A_i = \left( \left( A'_{ik} \right)_x , \left( A'_{ik} \right)_y \right) \\
ID_k^t &\leftarrow h_6 \left( ID_k^t \right) \\
\end{align*}
\]

### Step 4

\[
\begin{align*}
M_r &\leftarrow \left\{ h_{14}, h_{15}, h_{16}, h_{17}, h_{18}, h_{19}, h_{20}, h_{21}, h_{22}, T_r \right\} \\
\end{align*}
\]

### Step 5

### Step 6

### Step 7

\[
\begin{align*}
M_6 &\leftarrow \left\{ h_{23}, h_{12}, T_r \right\} \\
\end{align*}
\]

---

**Fig. 6 User–patient authentication phase of our scheme**

(7) If \( h_{30} = h \left( ID_i^t \| C_l \| R_l^t \| s_{skl}^t \| T_{10} \right) \) is false, \( ID_k^t \) aborts the protocol and returns \( \bot \), else derives access control string \( r_{ik}^t \) and if \( h_{27} = h \left( h_{23} \| h_{24} \| h_{25} \| r_{ik}^t \| \beta_k^t \right) \) is false, abort the protocol, else calculates session key \( s_{skl}^t = h \left( ID_i^t \| r_{ik}^t \| h_{14} \| h_{15} \right) \) and digest \( h_{31} \), and sends tuple \( \left\{ h_{19}, h_{26}, h_{28}, h_{31}, T_{11} \right\} \) to \( ID_i^t \) request authentication and initializes the value of the time to live of access control string \( r_{ik}^t \) to \( T_{11} \), and calculates access control label \( A_{ikl} = h \left( ID_i^t ; ID_i^t \| r_{ik}^t \right) \) and writes tuple \( \left\{ A_{ikl}, ID_i^t, T_r, r_{ik}^t \right\} \) to cache.

(8) After \( ID_k^t \) verifies that timestamp is valid, \( ID_k^t \) derives \( r_{ik}^t \) from \( h_{26} \), and if \( h_{28} = h \left( h_{26} \| r_{ik}^t \| \beta_k^t \right) \) is false, abort the protocol, else calculates session key \( s_{skl}^t = h \left( ID_i^t \| r_{ik}^t \| h_{19} \right) \), and if \( h_{31} = h \left( r_{ik}^t \| h_{19} \right) h_{26} \| h_{28} \| s_{skl}^t \) is false, abort the protocol, else initializes the value of the time to live of access control string \( r_{ik}^t \) to \( T_{11} \), and calculates access control label \( A_{ikl} = h \left( ID_i^t \| ID_i^t \| r_{ik}^t \right) \) and writes tuple \( \left\{ A_{ikl}, ID_i^t, T_r, r_{ik}^t \right\} \) to cache.

In fact, the user checks the validity of the relevant access control authorization before initiating a authentication request; that is, if \( T_{current} - T_{ikl} \leq \Delta L \) is true, the session key is negotiated directly by \( r_{ik}^t \); otherwise, the authentication and authorization requests are initiated according to the algorithm shown in Fig. 6, and \( ID_k^t \) and \( ID_i^t \) still use the dynamic shared secret \( R_l^t \) to achieve authenticated key agreement. Details are as follows:
(1) If $T_6 - T_{ikl} \leq \Delta L$ is true, ID$_k$ selects random number $r_4$ and inputs ID$_i$, and calculates $h_{14} = r_4 P$, $h_{15} = ID_i \oplus h(r_{ik})$, $h_{16} = ID_i \oplus h(r_{ik})$, $h_{18} = h(ID_i \| ID_i \| h_{14} \| h_{15} \| h_{16} \| A_{ikl} \| T_6)$, and sends tuple $(h_{14}, h_{15}, h_{16}, h_{18}, A_{ikl}, T_6)$ to patient ID$_k$.

(2) After verifying that the timestamp is valid, if searching for $A_{ikl}$ in cache is false, ID$_k$ aborts the protocol and returns ⊥, else derives ID$_i$ and ID$_f$ from $h_{15}$ and $h_{16}$, and if verifying that $T_{11} - T_{ikl} \leq \Delta L$ or $h_{18}$ is false, ID$_k$ aborts the protocol and returns ⊥, else ID$_k$ selects random number $r_5$ and calculates $h_{19} = r_5 P$, and calculates session key $ssik = h(ID_i \| r_{ik} \| r_5 \| h_{19})$, and sets the value of the time to live of access control string $r_{ik}$ to $T_{ikl} = T_{11}$, and updates access control label $A_{ikl} = h(A_{ikl})$ and updates tuple $(A_{ikl}, ID_i, ID_i, T_{ikl}, r_{ik})$ in cache; ID$_k$ calculates $h_{31} = h(r_{ik} \| h_{19} \| A_{ikl} \| ID_i \| SSik \| T_{11})$ and sends tuple $(h_{19}, h_{31}, T_{11})$ to ID$_i$.

(3) After verifying that the timestamp is valid, ID$_i$ calculates session key $ssik = h(ID_i \| r_{ik} \| r_5 \| h_{19})$, and sets the value of the time to live of $r_{ik}$ to $T_{ikl} = T_{12}$, and updates $A_{ikl} = h(A_{ikl})$ and updates tuple $(A_{ikl}, ID_i, ID_i, T_{ikl}, r_{ik})$ in cache; if $h_{31} = h(r_{ik})$ is true, ID$_i$ selects random number $r_7$, and calculates $h_{32} = r_7 P$, $A_i^* = r_7 P K$, $h_{34} = ID_i \oplus h((A_i^*)_x \| 1)$, $h_{35} = h((A_i^*)_x \| 2)$, $h_{36} = h_i \oplus h_i \oplus h_i$, and sends tuple $(h_{33}, h_{34}, h_{35}, h_{36}, T_{13})$ to RA.

(4) If ID$_k$ verifies that timestamp and $h_{32}$ are valid, then $ssik$ is accepted.

### 6.4 Password update

Fig. 7 Password update phase of our scheme

As shown in Fig. 7, users or patients can update their passwords online at anytime, anywhere. Details are as follows:

(1) User inputs ID$_i$, old password $PW_i$, and new password $PW_i^*$, generates biometric $B_i$, and calculates $\beta_i$ and $\beta_i^*$; ID$_i$ selects random number $r_7$, and calculates $h_{37} = r_7 P$, $A_i^* = r_7 P K$, $h_{34} = ID_i \oplus h((A_i^*)_x \| 1)$, and if searching for ID$_i$ in user registration list $L_U$ is false, abort the protocol, else if $h_{37} = h(ID_i \| h_{32} \| h_{34} \| h_{35} \| h_{36} \| T_{13})$ is false, returns ⊥, else calculates $\beta_i = h_{35} \oplus h((A_i^*)_x \| 2)$, $\beta_i = h_i \oplus h_i \oplus h_i$ if $HD(\beta_i, \beta_i) \leq \delta$ is false, returns ⊥, else updates $\gamma_i = \gamma_i^* = h_{36} \oplus \beta_i \oplus Cr_i$, and calculates $h_{38} = h_{39} = h(ID_i \| (A_i^*)_x \| \beta_i \| T_{14})$.

| User | RA (sk) |
|------|---------|
| $ID_i$, $PW_i$, $B_i$ | generates $B_i$, $\beta_i \leftarrow h_b(B_i) \oplus h(ID_i \| PW_i)$, $\beta_i^* \leftarrow h_b(B_i) \oplus h(ID_i \| PW_i^*)$, $r_7 \leftarrow Z_q^*$, $A_i \leftarrow r_7 P$, $h_{33} \leftarrow A_i$, $A_i^* \leftarrow r_7 P K$, $h_{34} \leftarrow ID_i \oplus h((A_i^*)_x \| 1)$, $h_{35} \leftarrow h_i \oplus h_i \oplus h_i$, $h_{36} \leftarrow h_i \oplus h_i \oplus h_i$, $h_{37} \leftarrow h(ID_i \| h_{33} \| h_{34} \| h_{35} \| h_{36} \| T_{13})$ |
| $ID_i$, $PW_i$, $B_i$ | calculates $\beta_i$, $\beta_i^*$, $r_7$, $A_i$, $A_i^*$, $h_{34}$, $h_{35}$, $h_{36}$, $h_{37}$, $T_{13}$ to RA if $T_{14} - T_{13} \leq \Delta T$ |
| $ID_i$, $PW_i$, $B_i$ | sends $h_{33}, h_{34}, h_{35}, h_{36}, h_{37}, T_{13}$ to RA |
| $ID_i$, $PW_i$, $B_i$ | if $ID_i = ID_i$ and $h_{37} = h(ID_i \| h_{33} \| h_{34} \| h_{35} \| h_{36} \| T_{13})$ |
| $ID_i$, $PW_i$, $B_i$ | $\beta_i \leftarrow h_{35} \oplus h((A_i^*)_x \| 2)$ |
| $ID_i$, $PW_i$, $B_i$ | $Cr_i = h(ID_i \| r_i \| sk)$ |
| $ID_i$, $PW_i$, $B_i$ | $\beta_i = Cr_i \oplus \gamma_i$ |
| $ID_i$, $PW_i$, $B_i$ | $\gamma_i^* = h_{36} \oplus \beta_i \oplus Cr_i$ |
| $ID_i$, $PW_i$, $B_i$ | $h_{38} = h(ID_i \| (A_i^*)_x \| \beta_i \| T_{14})$ |

If $T_{15} - T_{14} \leq \Delta T$ and $h_{38} = h(ID_i \| (A_i^*)_x \| \beta_i \| T_{14})$ accepts update
h \left( I D_i \| ( A_i^x \| \beta_i^y \| T_{14} ) \right), and sends tuple \((h_{38}, T_{14})\) to ID_j.

(3) After verifying that the timestamp and \(h_{38}\) are valid, ID_i accepts the update.

6.5 Ownership transfer

In this proposal, users can transfer ownership after passing RA authentication and authorization. Suppose ID_{i1} wants to transfer ownership of patient ID_{k} to ID_{i2}, the details are as follows:

(1) ID_{i1} generates a transfer request according to the algorithm FIG.6.Step1 and sends it to ID_{i2}.

(2) After verifying that the timestamp is valid, ID_{i2} also generates a transfer request according to the algorithm FIG.6.Step1 and sends it to RA.

(3) After verifying that the timestamp is valid, the RA verifies the identity of ID_{i1} and ID_{i2} according to the algorithm FIG.6.Step3. If it is false, abort the protocol, else if searches for ID_{i1} in the access control list AL_k of ID_{k} is false, aborts the protocol, else writes ID_{i2} to AL_k, and returns the message of successful transfer.

7 Security analysis

In this section, we analyze our proposed scheme SEMAS informally and formally. Moreover, we show that the proposed scheme is provably secure based on the threats model defined in Sect. 3.5, using ROR model (Abdalla et al. 2005), BAN-logic (Burrows et al. 1990), and AVISPA tool (Armando et al. 2005).

7.1 Informal security analysis

In this section, we will discuss how this proposal (SEMAS) provides mutual authentication, access control, session key privacy, and forward security, and how to resist known Internet attacks such as insider attacks, multi-factor security attacks, and impersonation attacks.

1. Mutual Authentication In SEMAS, the user (or patient) mutually authenticates with the RA by \(\beta_i\), the server mutually authenticates with RA by \(C r_j\), and the user and server mutually authenticate with the shared secret \(r_{ij}\) issued by the RA. ID_i encapsulates ID_{i} and \(\beta_i\) with the public key of RA. If \(h_3\) and \(\beta_i\) are valid, RA believes that ID_{i} is a legitimate user. RA encapsulates the shared secret \(r_{ij}\) with \(C r_j\). If \(h_11\) is valid, ID_j believes that RA is the holder of the private key corresponding to the system public key. ID_j encapsulates ID with the public key of RA. If \(h_6\) is valid, RA believes that ID_j is a legitimate server. RA encapsulates the shared secret \(r_{ij}\) with \(C r_j\). If \(h_{10}\) is valid, ID_j believes that RA is the holder of the private key corresponding to the system public key. On the basis of mutual authentication with RA, if \(h_{12}\) is valid, ID_j believes that ID_j is the common secret holder of RA certified; if \(h_{13}\) is valid, ID_j believes that ID_i is the common secret holder of RA certified.

2. Access Control In SEMAS, the RA manages access authorization of server (or patient’s sensor). The RA periodically generates an access control string \(r_{ij}\) for an authenticated and authorized user ID_i. A session can be established only if ID_i and ID_j hold the same access control string that meets the time limit.

3. Session Key Security In SEMAS, the user ID_i and the server ID_j independently compute the session key \(ss_{ij} = h \left( I D_i \| r_{ij} \| r_1 r_2 P \right)\), and the random numbers \(r_1, r_2,\) and \(r_{ij}\) are selected freshly each session, and the advantage of the enemy \(A\) to solve \(r_1, r_2,\) and \(r_{ij}\) is the advantage of attacking the EDLP security assumption, and it is negligible. So \(A\) needs to know all the random numbers and ID_i to calculate \(ss_{ij}\), and RA needs to know the random number \(r_1\) and \(r_2\) to calculate \(ss_{ij}\).

4. Forward Security In SEMAS, the user ID_i and the server ID_j independently compute the session key \(ss_{ij} = h \left( I D_i \| r_{ij} \| r_1 r_2 P \right)\), and the random numbers \(r_1, r_2,\) and \(r_{ij}\) are selected freshly each session, and the advantage of the enemy \(A\) to solve \(r_1, r_2,\) and \(r_{ij}\) is the advantage of attacking the EDLP security assumption, and it is negligible. In addition, the information exchanged in the protocol is ECC ciphertexts and hash values generated by fresh random numbers. Therefore, the advantage of adversary tracking session is equivalent to the advantage of attacking EDLP security assumptions, which is negligible.

5. Privacy Protection In the authentication and key agreement phase of the protocol, both ID_i and ID_j are transmitted in random pseudonym form \(h_1\) and \(h_5\), and the advantage of adversary \(A\) attacking these pseudonyms is equivalent to the advantage of attacking EDLP security assumption, which is negligible, so the advantage of \(A\) obtaining \(I D_i\) and \(I D_j\) also is negligible. In addition, the information exchanged in the protocol is ECC ciphertexts and hash values generated by fresh random numbers. Therefore, the advantage of adversary tracking session is equivalent to the advantage of attacking EDLP security assumptions, which is negligible. In SEMAS, the biometric vector in the registration phase is encapsulated in \(\gamma_i\) by the RA’s private key. According to the hash security assumption, the adversary’s advantage of getting \(\beta_i\) from RA’s \(L_U = \text{negl}(\lambda)\), which is negligible; during the authentication phase, the biometric vector is encapsulated in \(h_2\) by a random number and RA’s private key. According to the ECC security assumption, the
In this section, we follow the adversary capabilities defined in Sect. 3.5 to modify the ROR model defined in Abdalla et al. (2005) to a security model suitable for SEMAS.

The protocol participants in SEMAS are $\mathcal{P} = \mathcal{U} \cup \mathcal{S} \cup \mathcal{RA}$, where user $\mathcal{U} \in \mathcal{U}$, server $\mathcal{S} \in \mathcal{S}$, and registered authorization party $\mathcal{RA} \in \mathcal{RA}$. $\mathcal{P}^i$ represents the $i^{th}$ instance of $\mathcal{P}$.

### 7.2 Formal security proof using ROR model

In this section, we formally analyze SEMAS’s semantic security of the session key using the Real-Or-Random (ROR) (Abdalla et al. 2005) security model.

#### 7.2.1 ROR security model

In this section, we follow the adversary capabilities defined in Sect. 3.5 to modify the ROR model defined in Abdalla et al. (2005) to a security model suitable for SEMAS.

The protocol participants in SEMAS are $\mathcal{P} = \mathcal{U} \cup \mathcal{S} \cup \mathcal{RA}$, where user $\mathcal{U} \in \mathcal{U}$, server $\mathcal{S} \in \mathcal{S}$, and registered authorization party $\mathcal{RA} \in \mathcal{RA}$. $\mathcal{P}^i$ represents the $i^{th}$ instance of $\mathcal{P}$.

#### Definition 7.1 Partnering: The notion of partnering refers to the instances sharing the same non-null session identifications (sid). The sid is a partial transcript of the conversation between the participants’ instances.

#### Definition 7.2 Freshness: The instance $\mathcal{P}^i$ is considered fresh if their corresponding session key is not revealed to the adversary $\mathcal{A}$.

#### Definition 7.3 Adversary: The adversary $\mathcal{A}$ is assumed to have a full control over the communications as mentioned in Sect. 3.5. During the running of the protocol, $\mathcal{A}$ can initiate multiple parallel sessions to $\mathcal{P}$. The ability of $\mathcal{A}$ is simulated by the following oracle queries:

- $\text{Execute}(\mathcal{P}^1, \mathcal{P}^2)$: This oracle models the passive attack ability of $\mathcal{A}$. The input of this oracle is empty, and the output is a copy of the interactive information of the protocol instance.
- $\text{Send}(\mathcal{P}^i, M)$: This oracle models an active attack. This query allows $\mathcal{A}$ to send a message $M$ to a participant instance $\mathcal{P}^i$ and to receive a response message.
- $\text{Corrupt}(\mathcal{P}^i, \theta)$: This oracle models the ability of $\mathcal{A}$ to completely corrupt $\mathcal{P}^i$. The input of this oracle is the long-term secret of $\mathcal{P}^i$ except $\theta$, and the output is the long-term secret factor $\theta$ of $\mathcal{P}^i$ leakage.
- $\text{Test}(\mathcal{P}^i)$: This oracle is used to test the semantic security of the session key. $\mathcal{A}$ is allowed to execute several $\text{Execute}(\Delta)$, $\text{Send}(\Delta)$ and $\text{Corrupt}(\Delta)$ queries and then execute a $\text{Test}(\Delta)$ query. Return $\perp$ if the session key of instance has not been established; otherwise, return the session key only if $\mathcal{A}$ wins the coin flipping game; otherwise returns a random string of the same length as the session key.
- $\text{Hash}(a, h(a))$: This oracle maintains a mapping table $HT$ between $a$ and $h(a)$. If the input of this query is $a$, if and only if there is $(a, h(a))$ in $HT$, return $h(a)$; otherwise, return the randomly selected hash value $h(a)$ and write $(a, h(a))$ into $HT$.

#### Definition 7.4 Adversary advantage: The advantage of an adversary $\mathcal{A}$ in destroying the AKE semantic security of the protocol $\mathcal{P}$ is defined as

$$\text{Adv}_{\mathcal{P}}^{\text{AKE}}(\mathcal{A}) = |2 \cdot \Pr[\text{Succ}] - 1|.$$
$D_{PW}$, $D_{Bio}$, and $D_H$ denote the uniformly distributed dictionary of identity, password, biometric, and hash, respectively. $Adv_A^{SEMAS}$ is the advantage of the $A$ attacking EDLP security assumption defined in Sect. 3.2. The advantage of $A$ to break the semantic security of SEMAS is expressed as

$$Adv_A^{SEMAS} = 2|Adv_A^{G_0} - Adv_A^{G_1}|$$

$$\leq q_h/|D_h| + max \left( q_h^2/|D_H|, 2q_h/|D_{PW}| \right) + negl(\lambda).$$

where $q_h$, $q_s$, $|D_{ID}|$, $|D_{PW}|$, $|D_{Bio}|$, and $|D_H|$ denote the number of Hash queries, the number of Send queries, the range space of $D_{ID}$, $D_{PW}$, $D_{Bio}$, $D_H$, respectively.

**Proof**

Let $Succ_i$ be the event that $A$ wins game $G_i$. These games begin from the real attack scenario. We gradually change the simulation rules of each game. In the final game, $A$ will have no other advantage except to flip the coin.

$G_0$: $G_0$ is the real attack scenario; according to the definition 7.4, we have

$$Adv_A^{SEMAS} = |2 \cdot Pr [Succ_0] - 1|. \quad (1)$$

$G_1$: Based on $G_0$, Execute($P_r^x$, $P_r^z$), Execute($P_j^x$, $P_j^z$), Execute($P_r^x$, $P_j^z$), and Execute($P_j^x$, $P_r^z$) are added to simulate the passive attack ability of $A$. However, $A$ has to execute Test($ID_j^x$) and Test($ID_j^z$). Because both user ID$_j$ and server ID$_j$ calculate the session key $ss_{ij} = h(ID_j||r_i||r_j)P$ independently, and the random numbers $r_1$, $r_2$ and $r_{ij}$ are selected freshly each session, and the advantage of $A$ to solve $r_1$, $r_2$, and $r_{ij}$ is the advantage of attacking the EDLP security assumption, it is negligible. $A$ cannot gain the advantage of calculating the session key by eavesdropping on M1, M2, M3, M4, and M5. Thus, we have

$$|Adv_A^{G_1} - Adv_A^{G_0}| \leq Adv_A^{EDLP} \quad (2)$$

$G_2$: In this game, $A$ executes Send ($ID_j^x$, $M_i^x$), Send ($ID_j^z$, $M_i^z$), Send ($ID_j^y$, $M_i^y$), Send ($ID_j^z$, $M_i^z$), and Send ($ID_j^y$, $M_i^y$) with the help of Hash query to implement active attacks. Since random numbers are introduced in M1, M2, M3, M4, and M5, the Hash query will not cause conflicts. According to the birthday attack, we have

$$|Adv_A^{G_2} - Adv_A^{G_1}| \leq q_h^2/2|D_h| \quad (3)$$

$G_3$: We transfer $G_2$ to this game by adding the Correct ($ID_i^x$, $PW_i$) or Correct ($ID_i^y$, $Bio_i$) oracle to enhance attack ability of $A$. We assume that $A$ has acquired the $ID_i$, and then, there are three strategies to attack two-factor security, and he/she chooses the one with the highest probability.

**Case1**: $A$ breaks the $\beta_i^j$ with the result of the Hash query when the $PW_i$ has been compromised. According to the birthday attack, we have

$$|Adv_A^{G_3-Case1} - Adv_A^{G_2}| \leq q_h^2/2|D_H| \quad (4)$$

**Case2**: $A$ also breaks the $\beta_i^j$ by guessing result $Bio_i^j$ when the $PW_i$ has been compromised; we have

$$|Adv_A^{G_3-Case2} - Adv_A^{G_2}| \leq q_s/|D_{Bio}|. \quad (5)$$

**Case3**: $A$ breaks the $h_2^i$ by guessing result $PW_i^*$ when the $Bio_i$ has been compromised; we have

$$|Adv_A^{G_3-Case3} - Adv_A^{G_2}| \leq q_s/|D_{PW}|. \quad (6)$$

Combining the Case1-Case3 of $G_3$ and comparing to $G_2$, we have

$$|Adv_A^{G_3} - Adv_A^{G_2}| \leq \max \left( q_h^2/|D_H|, q_s/|D_{PW}| \right). \quad (7)$$

After using all the attack capabilities simulated by the oracle, $A$ can only use test query to find the session key, so

$$Adv_A^{G_4} = 1/2. \quad (8)$$

Finally, from Eqs. (1) and (8) we have

$$\frac{1}{2} Adv_A^{SEMAS} = |Adv_A^{G_0} - \frac{1}{2}| = |Adv_A^{G_0} - Adv_A^{G_1}| \quad (9)$$

According to the triangle inequality theorem, combining equations (2), (3), (7), (9), and Definition 3.1, we have

$$Adv_A^{SEMAS} = 2|Adv_A^{G_0} - Adv_A^{G_1}| \leq q_h^2/|D_h| + \max \left( q_h^2/|D_H|, 2q_s/|D_{PW}| \right) + negl(\lambda).$$

□

### 7.3 Formal security proof with BAN-logic

We utilize the widely recognized BAN-logic (Burrows et al. 1990) to prove that in the proposed scheme, the mutual authentication between a registered legitimate user ID$_j$ and medical server ID$_j$ is achieved with the help of RA. Notations used in the BAN-logic are shown in Table 2.

#### 7.3.1 Rules

In this section, we present some of the main BAN-logic rules for security proof.

*Rule 1.* Message meaning rule:
According to the BAN-logic, our scheme need to achieve the following five main goals:

\[ \text{Goa}_1: |ID_i| \equiv |ID_j| \xrightarrow{X} |ID_i| \]
\[ \text{Goa}_2: |ID_i| \equiv |ID_j| \xrightarrow{X} |ID_j| \]
\[ \text{Goa}_3: |ID_j| \equiv |ID_i| \xrightarrow{K} |ID_j| \]
\[ \text{Goa}_4: |ID_j| \equiv |ID_i| \xrightarrow{K} |ID_i| \]

7.3.3 Hypotheses

According to the BAN-logic, the initialization conditions of our scheme are assumed as follows:

\[ \text{Hyp}_1: |ID_i| \equiv |ID_j| \xrightarrow{\#(r_1)} |ID_i| \]
\[ \text{Hyp}_2: |ID_i| \equiv |ID_j| \xrightarrow{\#(r_2)} |ID_j| \]
\[ \text{Hyp}_3: |ID_i| \equiv |ID_i| \xrightarrow{\beta_i} |RA| \]
\[ \text{Hyp}_4: |RA| \equiv |ID_i| \xrightarrow{\beta_i} |RA| \]
\[ \text{Hyp}_5: |ID_i| \equiv |ID_i| \xrightarrow{r_{ij}} |RA| \]
\[ \text{Hyp}_6: |ID_j| \equiv |ID_j| \xrightarrow{Cr_{ij}} |RA| \]
\[ \text{Hyp}_7: |RA| \equiv |ID_j| \xrightarrow{Cr_{ij}} |RA| \]
\[ \text{Hyp}_8: |ID_j| \equiv |ID_j| \xrightarrow{f_{ij}} |RA| \]
\[ \text{Hyp}_9: |ID_i| \equiv |RA| \Rightarrow |ID_i| \xrightarrow{f_{ij}} |ID_j| \]
\[ \text{Hyp}_{10}: |ID_j| \equiv |RA| \Rightarrow |ID_j| \xrightarrow{f_{ij}} |ID_i| \]
\[ \text{Hyp}_{11}: |ID_i| \equiv |RA| \Rightarrow |ID_j| \sim r_2 P. \]

7.3.4 The idealized form of messages

In this section, we transform the general form of messages in our scheme into idealized ones.

**M_1:** From

\[ h_0 = A_i, \]
\[ h_1 = ID_j \oplus h \left( \left( A_j^* \right)_y \|1 \right). \]
\[ h_2 = \beta_i \oplus h \left( \left( A_i^* \right)_x \right), \]
\[ h_3 = h \left( ID_i \| ID_j \| h_0 \| h_1 \| h_2 \| T_j \right) \]
\[ \left( ID_i \right)_{ID_i \xrightarrow{\#(A_i^*)y} RA} \] and \( \left( ID_j, A_i, \left( A_i^* \right)_x \right)_{ID_j \xrightarrow{\rho_j} RA}. \)  \hspace{1cm} (10)

**M_2:** From

\[ h_0, h_1, h_2, h_3, \]
\[ h_4 = A_j, \]
\[ h_5 = ID_j \oplus h \left( \left( A_j^* \right)_x \right), \]
\[ h_6 = h \left( ID_j \| h_3 \| h_4 \| h_5 \| T_j \| Cr_j \right) \]
\[ \left( ID_j \right)_{ID_j \xrightarrow{\#(A_j^*)x} RA} \] and \( \left( ID_i, A_j, \left( A_j^* \right)_y \right)_{ID_j \xrightarrow{\rho_j} RA}. \)  \hspace{1cm} (11)

**M_3:** From

\[ h_8 = r_{ij} \oplus h \left( \left( A_j^* \right)_y \| Cr_j \right), \]
\[ h_9 = r_{ij} \oplus h \left( \left( A_i^* \right)_y \| A_i \| \beta_i \right), \]
\[ h_{10} = h \left( h_8 \| h_9 \| r_{ij} \| \left( A_j^* \right)_y \| Cr_j \right), \]
\[ h_{11} = h \left( h_9 \| r_{ij} \| \left( A_j^* \right)_y \| \beta_i \right) \]
\[ \left( r_{ij} \right)_{ID_j \xrightarrow{\rho_j} RA} \] and \( \left( ID_j, \left( A_j^* \right)_y \right)_{ID_j \xrightarrow{\rho_j} RA}. \)  \hspace{1cm} (12)

**M_4:** From

\[ h_0, h_9, h_{11}, \]
\[ h_{12} = h \left( r_{ij} \| h_5 \| h_9 \| h_{11} \| ss_{ij} \right) \]
\[ \left( r_{ij} \right)_{ID_i \xrightarrow{\rho_i} RA} \] and \( \left( ID_i, \left( A_i^* \right)_y, A_j \right)_{ID_i \xrightarrow{\rho_i} RA}. \)  \hspace{1cm} (13)
Based on the idealized message, BAN-logic rules, and initial condition hypotheses, the security analysis of our scheme is as follows:

According to the message $M_1$, hypothesis $Hyp_4$, and rule $Rul_1$, we have

$$RA \triangleleft \left( (ID_1) \left( (A_i^+)^{s_i}_i \rightarrow RA \right), A_i \right)_{ID_1} \triangleright \equiv RA,$$  
$$RA \equiv ID_1 \triangleright \equiv (ID_1, A_i).$$  

According to the message $M_2$, hypothesis $Hyp_7$, and rule $Rul_1$, we have

$$RA \triangleleft \left( (ID_j) \left( (A_j^+)^{s_j}_j \rightarrow RA \right), A_j \right)_{ID_1} \triangleright \equiv RA,$$  
$$RA \equiv ID_j \triangleright \equiv (ID_j, A_j).$$  

According to the message $M_3$, hypothesis $Hyp_6$, and rule $Rul_1$, we have

$$ID_j \triangleright \left( (r_{ij}) \left( (A_j^+)^{s_j}_j \rightarrow RA, A_i \right)_{ID_1} \triangleright \equiv RA,$$  
$$ID_j \equiv RA \triangleright \equiv (r_{ij}, A_j).$$  

According to the conclusions (16) and (17), hypothesis $Hyp_6$, rules $Rul_2$, and $Rul_4$, we have

$$ID_j \equiv RA \triangleright \equiv ID_j \triangleright \equiv RA.$$  

According to the conclusions (18), hypothesis $Hyp_{10}$, rules $Rul_2$, and $Rul_4$, we have

$$ID_j \equiv ID_i \triangleright \equiv ID_j.$$  

According to the conclusion (18) and (19), hypothesis $Hyp_6$, rules $Rul_3$, we achieve goal $Goa_2$

$$ID_j \equiv ID_i \triangleright \equiv ID_j.$$  

According to the message $M_4$, hypothesis $Hyp_3$, and rule $Rul_1$, we have

$$ID_i \triangleright \left( (r_{ij}) \left( (A_j^+)^{s_j}_j \rightarrow RA \right), A_j \right)_{ID_1} \triangleright \equiv RA,$$  
$$ID_i \equiv RA \triangleright \equiv (r_{ij}, A_j).$$  

According to the conclusion (21), hypothesis $Hyp_3$, rules $Rul_2$, and $Rul_4$, we have

$$ID_i \equiv RA \equiv ID_i \triangleright \equiv RA.$$  

According to the conclusion (21) and (22), hypothesis $Hyp_5$, rules $Rul_2$, and $Rul_4$, we have

$$ID_i \equiv ID_j \triangleright \equiv ID_i.$$  

According to the conclusion (22) and (23), hypothesis $Hyp_3$, rules $Rul_3$, we achieve goal $Goa_3$

$$ID_i \equiv ID_j \triangleright \equiv ID_i.$$  

According to the conclusion (21), (22), (23) and (15), hypothesis $Hyp_9$ and $Hyp_{11}$, rules $Rul_3$, we achieve goal $Goa_3$

$$ID_i \equiv ID_j \triangleright \equiv ID_i.$$  

According to the message $Mess_5$, conclusion (20), and rule $Rul_1$, we have

$$ID_j \triangleright \left( (T_5) \left( (s_{ij}) \rightarrow ID_i \right), ID_i \triangleright \equiv ID_i,$$  
$$ID_j \equiv ID_i \triangleright \equiv (T_5).$$  

According to the conclusion (20) and (26), hypothesis $Hyp_{12}$, rules $Rul_3$, we achieve goal $Goa_4$

$$ID_j \equiv ID_i \triangleright \equiv ID_j \triangleright \equiv ID_i.$$  

### 7.4 Formal security verification using AVISPA

In this section, we use the widely adopted security protocol formal verification tool AVISPA Armando et al. (2005) to verify the security of the proposed scheme. AVISPA supports the High-Level Protocol Specification Language (HLPSL) to implement a security protocol to check whether the designed protocol is safe or unsafe. In AVISPA, there are four back-ends: On-the-fly-Model-Checker (OFMA), CL-based Attack Searcher (CL-AtSe), SAT-based Model-Checker (SATMC), and Tree-Automata-based Protocol Analyzer (TA4SP).
Table 2 Notations used in the BAN-logic

| Notations | Descriptions |
|-----------|--------------|
| $P \equiv X$ | $P$ believes a statement $X$ |
| #($X$) | The statement $X$ is fresh |
| $P \preceq X$ | $P$ sees the statement $X$ |
| $P \sim X$ | $P$ once said the statement $X$ |
| $P \Rightarrow X$ | $P$ has jurisdiction over statement $X$ |
| $P \leftrightarrow Q$ | $K$ is a secret shared by $P$ and $Q$ |
| $P \oplus Q$ | $X$ is a secret shared by $P$ and $Q$ and TTP |
| $(X, Y)$ | $X$ or $Y$ is one part of $(X, Y)$ |
| $(X)_K$ | $X$ is encrypted under the key $K$ |
| $(X)_K$ | $X$ is hashed with the key $K$ |
| $(X)_K$ | $X$ is XORed with the key $K$ |

Fig. 8 SEMAS’s verification results using AVISPA

We select the widely accepted OFMC and CL-AtSe back-ends to verify the security of SEMAS in Dolev and Yao (1983) threat model as mentioned in Sect. 3.5. As shown in Fig. 8, SEMAS is “SAFE,” which means that it is proven by AVISPA to resist known Internet attacks.

8 Performance analysis

This section demonstrates that SEMAS how to satisfy the security goals and application requirements from the security and functionality properties, computational complexity, and communication overhead.

Table 3 Security and functionality features comparison

| Features | He and Wang (2015) | Qin et al. (2018) | Feng et al. (2018) | Das et al. (2019) | Liao et al. (2021) | Shi et al. (2019) | SEMAS |
|----------|-------------------|------------------|-------------------|------------------|-------------------|-----------------|-------|
| P1       | ✓✓✓✓✓✓✓✓✓✓         |                  |                   |                   |                   |                 |       |
| P2       | ××××              |                  |                   |                   |                   |                 |       |
| P3       | ×                 | ✓✓✓✓✓✓✓✓✓✓        |                   |                   |                   |                 |       |
| P4       | ✓✓✓✓              | ×                 | ✓✓✓✓              |                   |                   |                 |       |
| P5       | ×                 | ✓                 | ××××              | ✓✓✓✓✓✓✓✓✓✓         |                   |                 |       |
| P6       | ✓✓✓✓              | ×                 | ✓✓✓✓              |                   |                   |                 |       |
| P7       | ✓✓✓✓              | ×                 | ✓✓✓✓              |                   |                   |                 |       |
| P8       | ×                 | ✓✓✓✓✓✓✓✓✓✓        |                   |                   |                   |                 |       |
| P9       | ✓✓✓✓              | ×                 | ✓✓✓✓              |                   |                   |                 |       |
| P10      | ×                 | ✓✓✓✓✓✓✓✓✓✓        |                   |                   |                   |                 |       |
| P11      | –                 | –                 | –                 | –                 | –                 |                 |       |
| P12      | ×                 | ✓✓✓✓✓✓✓✓✓✓        |                   |                   |                   |                 |       |
| P13      | ✓✓✓✓              | ×                 | ✓✓✓✓              |                   |                   |                 |       |
| P14      | ✓✓✓✓              | ×                 | ✓✓✓✓              |                   |                   |                 |       |
| P15      | ✓✓✓✓              | ×                 | ✓✓✓✓              |                   |                   |                 |       |
8.1 Security and functionality properties comparison

We evaluate the security and functionality features (P1: Mutual authentication, P2: Access Control, P3: Session key security, P4: Forward security, P5: Anonymity, P6: Against Insider Attack, P7: Against Multi-factor Security Attack, P8: Against User Impersonation Attack, P9: Against Server Impersonation Attack, P10: Against Patient Impersonation Attack, P11: Against Sensor Impersonation Attack, P12: Against Replay Attack, P13: Against Man-in-the-middle Attack, P14: Against Off-line Password Attack, P15: Against Smart card loss Attack.) of our SEMAS and compare it with six recently proposed typical multi-server authentication schemes in the literature. The details are shown in Table 3.

The results show that some schemes have weak forward security (Feng et al. 2018; Das et al. 2019) and anonymity (He and Wang 2015; Qi et al. 2018; Feng et al. 2018), and some schemes cannot resist the smart card loss attack, which in turn leads to other attacks (He and Wang 2015; Qi et al. 2018; Feng et al. 2018). Most schemes cannot support multiple business scenarios and do not have access control mechanisms.

8.2 Computation cost comparison

To evaluate the computational efficiency of SEMAS, we calculate and compare the computation overhead of authenticated key agreement phases of discussed protocols, including SEMAS, as shown in Table 5. We used gcc 5.4.0 to compile and run the SEMAS related operations based on the MIRACL library (MIRACL Crypto SDK, MIRACL 2021). The time-consuming overhead of the basic operations involved in these protocols is shown in Table 4 (Intel(R) Core(TM) i3-M380@2.53GHz processor, 8 GB RAM and with Ubuntu 16.04 operating system), and the notations $T_H$, $T_S$, and $T_E$ represent the computational cost of hash operation, symmetric encryption/decryption operation, and ECC scalar point multiplication operation, respectively. We assume that the computational complexity of the fuzzy extractor and ECC scalar point multiplication is close. Regardless of the overhead of XOR, hash, and point addition operation, the computation overhead of SEMAS is almost the same as that of other schemes, at about 0.01 ms (He and Wang 2015; Qi et al. 2018; Feng et al. 2018; Odelu et al. 2015; Das et al. 2019; Shuai et al. 2020; Luo et al. 2021) (Tables 5, 6).

8.3 Communication overhead comparison

To evaluate the communication efficiency of SEMAS, we calculate and compare the communication overhead of authenticated key agreement phases of discussed protocols, including SEMAS, as shown in Table 7. The byte length of the data structure transferred in these protocols is shown in Table 6. The notations $T_I$, $T_H$, $T_E$, $T_S$, and $T_N$ represent the byte length of identity, hash string, ECC block, symmetric ciphertext, random string, respectively. As in Odelu et al. (2015), we also assume that the length of the identity ($I_D_1$, $I_D_2$, $I_D_3$, $I_D_4$) (the time stamp is equal to the length), the hash value (e.g., SHA-1), and an elliptic curve point $P = (P_x, P_y)$ are 8 bytes, 20 bytes, and 40 bytes, respectively. In addition, we assume that the block size of symmetric ciphertext (e.g. AES) and a random number are 16 bytes, respectively (Table 6).

It can be seen from Table 7 that the user side overhead of SEMAS is almost the same as that of other online protocols, but the total communication overhead is higher than that of other protocols. The main reason is that in order to achieve
access control, anonymity and forward security, immune to offline password attack, and smart card loss attack, SEMAS introduces time stamp authentication mechanism and the user sends one more ECC block.

9 Conclusion

The secure communication and access control in the E-healthcare systems are very important, and the key means to achieve this goal is the authenticated key agreement and access authorization mechanism. This work first performs a cryptanalysis of existing schemes such as LACO and reveals the main reasons for the vulnerability of anonymity and forward security of these schemes, which can lead to impersonation attacks. Second, we proposed a multiple solution architecture for authentication and authorization in user-server, patient–server, user–patient, and other scenarios in E-healthcare. Third, based on the architecture, we design a secure and efficient multi-server authentication and access control scheme for E-healthcare. Security analysis shows that the proposed scheme can provide mutual authentication, access control, session key security, anonymity, and forward security and can resist known Internet attack such as insider attack, multi-factor security attacks, impersonation attacks, and intermediate data attacks. Efficiency analysis shows that under the premise of higher security, the proposed scheme has better computational efficiency than similar typical schemes. Due to high security, the communication efficiency is slightly lower than similar typical schemes. Nevertheless, the total communication overhead of the proposed scheme is only 520 bytes, while the user side communication overhead is almost the same as other schemes.

Acknowledgements This work was funded by the National Natural Science Foundation of China No. 61976142; the Zhejiang Province Natural Science Foundation of China under Grant No. LY19F020045; and the Smart City Construction Plan of Anning District, Lanzhou City (2021-2035).

9 Conclusion

The secure communication and access control in the E-healthcare systems are very important, and the key means to achieve this goal is the authenticated key agreement and access authorization mechanism. This work first performs a cryptanalysis of existing schemes such as LACO and reveals the main reasons for the vulnerability of anonymity and forward security of these schemes, which can lead to impersonation attacks. Second, we proposed a multiple solution architecture for authentication and authorization in user-server, patient–server, user–patient, and other scenarios in E-healthcare. Third, based on the architecture, we design a secure and efficient multi-server authentication and access control scheme for E-healthcare. Security analysis shows that the proposed scheme can provide mutual authentication, access control, session key security, anonymity, and forward security and can resist known Internet attack such as insider attack, multi-factor security attacks, impersonation attacks, and intermediate data attacks. Efficiency analysis shows that under the premise of higher security, the proposed scheme has better computational efficiency than similar typical schemes. Due to high security, the communication efficiency is slightly lower than similar typical schemes. Nevertheless, the total communication overhead of the proposed scheme is only 520 bytes, while the user side communication overhead is almost the same as other schemes.

Acknowledgements This work was funded by the National Natural Science Foundation of China No. 61976142; the Zhejiang Province Natural Science Foundation of China under Grant No. LY19F020045; and the Smart City Construction Plan of Anning District, Lanzhou City (2021-2035).

Author Contributions Qiao Yan contributed to supervision; Hailong Yao was involved in conceptualization, methodology, and writing—original draft preparation; Xingbing Fu contributed to writing—reviewing and editing; and Zhibin Zhang and Caihui Lan provided software.

Funding This work was funded by the National Natural Science Foundation of China No. 61976142; the Zhejiang Province Natural Science Foundation of China under Grant No. LY19F020045; and the Smart City Construction Plan of Anning District, Lanzhou City (2021-2035).

Availability of data and material Not applicable.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

Code availability The code that supports the findings of this study is available from the corresponding author upon reasonable request.

References

Abdalla M, Fouque PA, Pointcheval D (2005) Password-based authenticated key exchange in the three-party setting. In: International Workshop on Public Key Cryptography. Springer, pp 65-84. https://doi.org/10.1007/978-3-540-30580-4_6
Aghili SF, Mala H, Shojafar M, Peris-Lopez P (2019) LACO: lightweight three-factor authentication, access control and ownership transfer scheme for E-health systems in IoT. Futur Gener Comput Syst 96:410–424. https://doi.org/10.1007/s11277-015-2616-7
Alsahlani AYF, Popa A (2020) Analyzing of LAM-CIoT: lightweight authentication mechanism in cloud-based IoT environment. In: Proceedings of the 2020 IEEE Symposium Series on Computational Intelligence (SSCI), IEEE, pp 1837-1844. https://doi.org/10.1109/SSCI47803.2020.9308139
Amin R, Biswas GP (2015) Design and analysis of bilinear pairing based mutual authentication and key agreement protocol usable in multi-server environment. Wireless Pers Commun 84:439–462. https://doi.org/10.1007/s11277-015-2616-7
Amin R, Hafizul Islam S, Biswas GP, Khurram Khan M, Kamar N (2018) A robust and anonymous patient monitoring system using wireless medical sensor networks. Futur Gener Comput Syst 80:483–495. https://doi.org/10.1016/j.future.2016.05.032
Amin R, Hafizul Islam S, Gope P, Raymond Choo KK, Tapas N (2019) Anonymity preserving and lightweight multimedia server authentication protocol for telecare medical information system.
sor networks. IEEE Trans Depend Sec Comput. https://doi.org/10.1109/TDSC.2020.2974220

Wang D, Zhang X, Zhang Z, Wang P (2020) Understanding security failures of multi-factor authentication schemes for multi-server environments. Comput Sec 88:101619. https://doi.org/10.1016/j.cose.2019.101619

Wang D, Zhang Z, Wang P, Yan J, Huang X (2016) Targeted online password guessing: an underestimated threat. In: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security (CCS’16), ACM, 1242–1254. https://doi.org/10.1145/2976749.2978339

Wazid M, Das Ashok K, Vasilakos Athanasios V (2018) Authenticated key management protocol for cloud-assisted body area sensor networks. J Netw Comput Appl 123:112–126. https://doi.org/10.1016/j.jnca.2018.09.008

Wazid M, Das AK, Bhat V, Vasilakos AV (2020) LAM-CIoT: lightweight authentication mechanism in cloud-based IoT environment. J Netw Comput Appl 150:102496. https://doi.org/10.1016/j.jnca.2019.102496

Yang D, Yang B (2010) A biometric password-based multi-server authentication scheme with smart card. In: Proceedings of the 2010 international conference on computer design and applications, IEEE, NY, pp 554–559. https://doi.org/10.1109/ICDAA.2010.5541128

Yao H, Fu X, Wang C, Meng C, Hai B, Zhu S (2019a) Cryptanalysis and improvement of a remote anonymous authentication protocol for mobile multi-server environments. In: Proceedings of the 2019 IEEE fourth international conference on data science in cyberspace, IEEE, NY, pp 19222220. https://doi.org/10.1109/DSC.2019.00015

Yao H, Wang C, Fu X, Liu C, Wu B, Li F (2019) A privacy-preserving RLWE-based remote biometric authentication scheme for single and multi-server environments. IEEE Access 7:109597–109611. https://doi.org/10.1109/ACCESS.2019.2933576

Yao H, Wang C, Fu X, Liu C, Wu B, Li F (2020) Impersonation attacks on lightweight anonymous authenticated key exchange scheme for IoT. Cryptology ePrint Archive, Report 2020/143, https://eprint.iacr.org/2020/143.pdf. Accessed 26 Apr. 2020

Yoon EJ, Yoo KY (2013) Robust biometrics-based multi-server authentication with key agreement scheme for smart cards on elliptic curve cryptosystem. J Supercomput 63:235–255. https://doi.org/10.1007/s11227-010-0512-1

Zhang L, Zhang Y, Tang S, Luo H (2018) Privacy protection for E-health systems by means of dynamic authentication and three-factor key agreement. IEEE Trans Industr Electron 65(3):2795–2805. https://doi.org/10.1109/TIE.2017.2739683

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.