# An Efficient and Secure Authentication for Ambient Assisted Living System

Myung-Kyu Yi and Taeg-Keun Whangbo*

*College of IT Convergence, Gachon University, 13120, South Korea*
*E-mail: kainos@gachon.ac.kr; tkwhangbo@gachon.ac.kr*
**Corresponding Author*

Received 31 May 2021; Accepted 13 November 2021;
Publication 19 February 2022

## Abstract

Although the birthrate is declining, the average life expectancy continues to increase. Therefore, it is more important for elderly people to maintain their independence while staying at home. Ambient Assisted Living (AAL) includes the use of devices and methods of ensuring that elderly people can stay safe and age at home rather than at a facility. Assisted living services help people live as independently and safely as possible when they can no longer perform everyday activities on their own. Because the information transmitted in AAL systems is personal, the security and privacy of such data are becoming important issues that must be addressed. Herein, we propose an efficient and secure authentication scheme for an AAL system. Our proposed authentication scheme not only satisfies several important security requirements of such a system but also withstands various types of attacks. Moreover, the proposed authentication scheme achieves lightweight performance by manipulating basic cryptographic operations including bitwise-eXclusive-OR (XOR) and hash functions. We simulated our proposed authentication scheme using Automated Validation of Internet Security Protocols and Applications (AVISPA), which is a prominent security verification tool. Security and performance analysis show that our proposed

*Journal of Web Engineering, Vol. 21_3,* 693–712.
doi: 10.13052/jwe1540-9589.2136

scheme is not only robust against several attacks and has a lower computational cost in terms of execution time than those of existing authentication schemes.

**Keywords:** Web security and privacy, wearable computing, ambient assisted living, healthcare.

## 1 Introduction

Population aging is a global phenomenon, and populations are rapidly growing in many regions. The aging of populations globally is the result of continual decline in fertility rates and increase in life expectancy. As a result, the number of people aged 60 years and older may nearly triple to 2 billion by 2050, accounting for almost one quarter of the expected 9.2 billion population globally. As people age, many suffer from physical disabilities or memory loss and thus require more healthcare services. The main disadvantages of an aging population include increased healthcare costs. Elderly people are more prone to illnesses and ailments. The probability of occurrence of diabetes, hypertension, or cancer increases with age. An increase in the number of sick people will pressurize healthcare facilities, which may not be able to cope with the demand. Advancements in information and communication technology have allowed for remote delivery of health and care services, supporting vulnerable people in their own homes rather than in hospitals or residential facilities. One method to address this problem is to rely more on AAL technology, known as a welfare technology, in the homes of elderly and disabled people [1]. AAL provides a system comprising smart devices, wireless networks, software applications, computers, and medical sensors for healthcare monitoring. AAL aims to ensure safety and quality of health of elderly adults and extend the number of years during which senior citizens can live independently in an environment of their own preference. AAL can provide assistance to elderly and handicapped patients through continuous activity monitoring and access to medical support. AAL technologies can provide greater safety for the elderly, offering emergency-response mechanisms and fall-detection solutions. Because the information transmitted in AAL systems is highly sensitive, security and privacy of such data are becoming important issues that must be addressed. Security protocols designed for AAL systems should satisfy security requirements such as integrity, confidentiality, availability, and anonymity. Herein, we propose a novel authentication scheme for AAL systems, which can improve the efficiency and guarantee

security. The proposed authentication scheme for an AAL system only allows authorized users to access a secure data exchange with the help of a session key that is shared during authentication.

The rest of this paper is organized as follows. Section 2 reviews related studies. Section 3 presents the system architecture of the proposed scheme for an AAL system, along with the security requirements. The proposed authentication scheme is then described in Section 4. Section 5 presents the simulation results of our proposed authentication scheme using AVISPA. Finally, we provide some concluding remarks in Section 6.

## 2 Related Work

In this section, we briefly review previous works related to the wireless body area networks (WBANs) based AAL system [2–4, 6–8].

Liu et al. [6] proposed two remote certificateless authentication protocols to preserve the privacy of potential WBANs users when they access network medical service through WBANs terminals. With this scheme, the full private key of the user consists of not only the partial private key issued by a semi-trusted key generation center but also a user private key generated by the user. However, this scheme is insecure against stolen verifier table attacks. Moreover, the authors did not obtain a sufficiently lightweight protocol owing to the complex bilinear pairing operation and computations.

To avoid a bilinear pairing operation, Zhao et al. [7] proposed an efficient authentication protocol without using a bilinear pairing operation for a WBANs. They proposed an efficient and anonymous identity-based authentication scheme for WBANs using Elliptic-Curve Cryptography(ECC). Owing to the use of an ID-based concept, no certificate is required during communication. However, Zhao's scheme is insecure because an adversary can trace the user based on the constant value of the pseudo identity.

He et al. [8] presented a new authentication model suited for an AAL system based on elliptic-curve cryptography. This scheme not only supports several important security requirements of the AAL system but also withstands various types of attacks. However, this scheme is prone to tracking attacks and fails to achieve untraceability.

Moreover, owing to the high resource constraints of such sensors, the previous works are not lightweight enough to be suitable for AAL sensor nodes. The proposed authentication scheme delivers lightweight performance by manipulating basic cryptographic operations including XOR and hash functions.

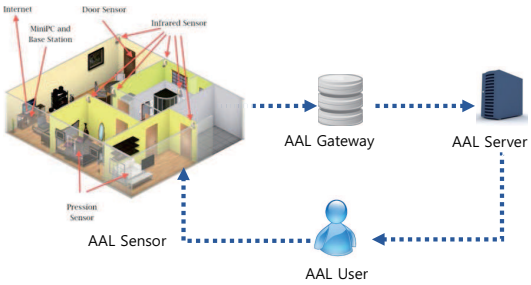


**Figure 1** The proposed system architecture for AAL.

## 3 System Architecture and Security Requirements

In this section, we present the architecture of a WBANs based AAL system and security requirements that should be satisfied by the proposed scheme.

### 3.1 System Architecture of WBAN-based AAL System

We first describe the architecture of a WBANs based AAL system as shown in Figure 1 [9]. The architecture of an AAL system typically consists of three main components. At one end of the architecture, the network consists of WBANs based AAL sensor nodes with low power consumption. The WBANs based AAL sensor is responsible for gathering the required user data. The data generated from the AAL sensors are forwarded to the AAL gateway. The AAL gateway is responsible for the acquisition, management, and forwarding of data on vital signs captured by the AAL sensors. The AAL gateway accordingly sends data to the AAL server through its Internet connection. The AAL server processes and stores data and performs specific situational detection for the safety of elderly people. Moreover, the AAL server analyzes the accumulated data and provides the results as an AAL service. Therefore, the AAL server can send notifications to the registered caregivers of the users.

### 3.2 Security Requirements for AAL System

The authentication scheme is susceptible to attacks if used on insecure communication channels such as the Internet. In this section, some security requirements that should be satisfied by the proposed scheme are stated as follows:

- Mutual authentication: To allow only authenticated users to use an AAL service, a mutual authentication is required. Both the client and server

must authenticate the identity of the other before actual communication occurs.

- Session-key agreement: After performing mutual authentication between entities, sensitive data transferred through the Internet need to be encrypted by a session key shared among them. Therefore, a robust authentication protocol for the AAL system should provide a session-key agreement.
- No verification table: In several previously proposed authentication schemes, the access point must maintain a verification table for mutual authentication. However, an adversary can impersonate a user by modifying the values in the verification table to obtain AAL services from the access point. Thus, the proposed authentication scheme for the AAL system should avoid keeping the verification table for authentication purposes.
- Perfect forward secrecy: It is a method of ensuring that all transactions sent over the Internet are secure. The authentication scheme must prevent an adversary from being able to access data from a group of transactions even if they can hack the encryption for a single communication sent over the Internet.
  Many of the available sensors used for monitoring blood sugar, blood pressure, and pulse-rate are capable of sending vital signs to the WBAN-based AAL system, it is importat to use group authentication to improve the efficiency of one-by-one authentication. But, this is outside the scope of this paper to discuss how to apply a AAL system.

## 4 Proposed Authentication Scheme

In this section, we present a new authentication scheme for an AAL system. First, in Table 1, we define some of the notations used in our proposed scheme. Our proposed scheme is divided into two phases: registration and authentication. We assume that the public parameters of ECC have previously been generated by the AAL server.

### 4.1 Registration

We assume that the AAL sensor and AAL gateways share a session key, $K_s$. Similarly, we assume that the AAL gateway and AAL servers share a session, $K_g$, and the AAL sensor and AAL server share a session key, $K_p$. During this phase, the AAL sensor, AAL gateway, and AAL server authenticate each

**Table 1** Major notations for the proposed scheme

| Parameter | Description |
|---|---|
| $p, q, r$ | Randomly chosen prime number |
| $\oplus$ | XOR logical operation |
| $H(A)$ | Hash function of A |
| $G$ | Generator of the elliptic curve |
| $q \times G$ | Elliptic-Curve Scalar Multiplication |
| $T_\alpha$ | Timestamp of $\alpha$ |
| $N_\beta$ | Nonce value of $\beta$ |
| $K$ | Common session key |
| $E_m$ | Encryption operation with $m$ key |
| $D_m$ | Decryption operation with $m$ key |
| $\|\|$ | Concatenation operation |

other and negotiate a common session key, $K$, for the later authentication phase. As shown in Figure 2, the registration phase is divided into four rounds, which are as follows:

**Round 1:**

- Step 1. An AAL sensor chooses a random number $p$.
- Step 2. Simultaneously, the AAL sensor computes $U_{s1}$ = p$G$, where $G$ is the elliptic-curve generator.
- Step 3. The AAL sensor sends $U_{s1} \oplus K_s$ to the AAL gateway.

**Round 2:**

- Step 1. When receiving the message, the AAL gateway computes $U_{s1} \oplus K_s \oplus K_s$ to obtain the value of $U_{s1}$.
- Step 2. The AAL gateway chooses a random number $q$. The AAL gateway then computes $U_{g1} = q \times U_{s1}$.
- Step 3. The AAL gateway computes $U_{g2} = q \times G$.
- Step 4. The AAL gateway computes $U_{g1} \oplus K_g$ and sends $U_{g1} \oplus K_g$ to the AAL server.
- Step 5. The AAL gateway computes $U_{g2} \oplus K_g$ and sends $U_{g1} \oplus K_g$ to the AAL server.

**Round 3:**

- Step 1. When receiving the messages, the AAL server computes $U_{g1} \oplus K_g \oplus K_g$ to obtain the value of $U_{g1}$.

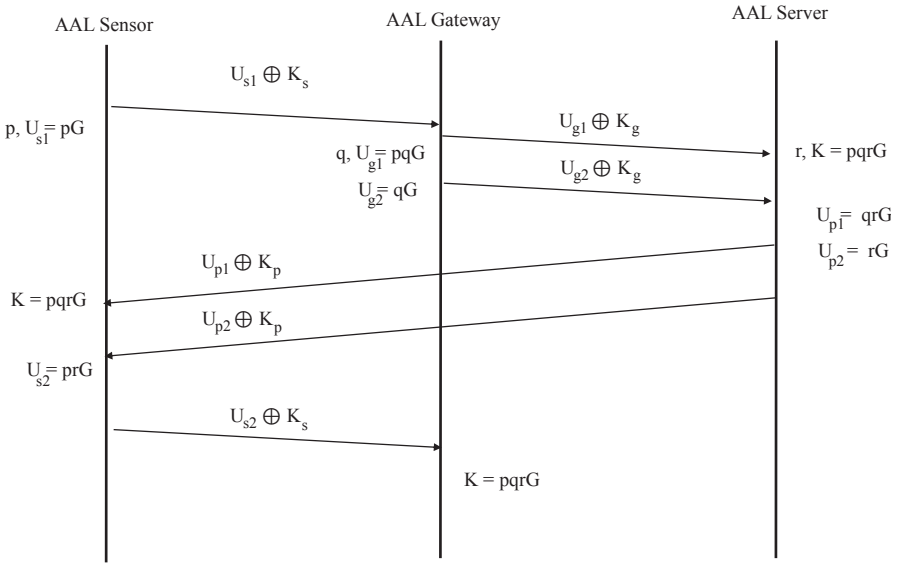


**Figure 2** Registration phase.

- Step 2. The AAL server chooses a random number $r$. The AAL server then computes the session key $K = r \times U_{g1}$( $i.e.,\ K = p \cdot q \cdot r \times G$).
- Step 3. The AAL server computes $U_{p1} = q \times r \times G$ and $U_{p2} = r \times G$.
- Step 4. The AAL server computes $U_{p1} \oplus K_g$ and sends $U_{p1} \oplus K_p$ to the AAL sensor.
- Step 5. The AAL server computes $U_{p2} \oplus K_g$ and sends $U_{p2} \oplus K_p$ to the AAL sensor.

**Round 4:**

- Step 1. When receiving the messages, the AAL sensor computes $U_{p1} \oplus K_p \oplus K_p$ to obtain the value of $U_{p1}$.
- Step 2. The AAL sensor simultaneously computes $U_{p2} \oplus K_p \oplus K_p$ to obtain the value of $U_{p2}$.
- Step 3. The AAL sensor computes session key $K = p \cdot q \cdot r \times G$.
- Step 4. The AAL sensor computes $U_{s2} = p \times r \times G$ and $U_{s2} \oplus K_s$.
- Step 5. The AAL sends $U_{s2} \oplus K_g$ to the AAL gateway.

Finally, when receiving the messages, the AAL gateway can compute the session key $K = p \times q \times r \times G$. As a result, the AAL sensor, AAL gateway, and AAL servers share the common session key $K = p \times q \times r \times G$.

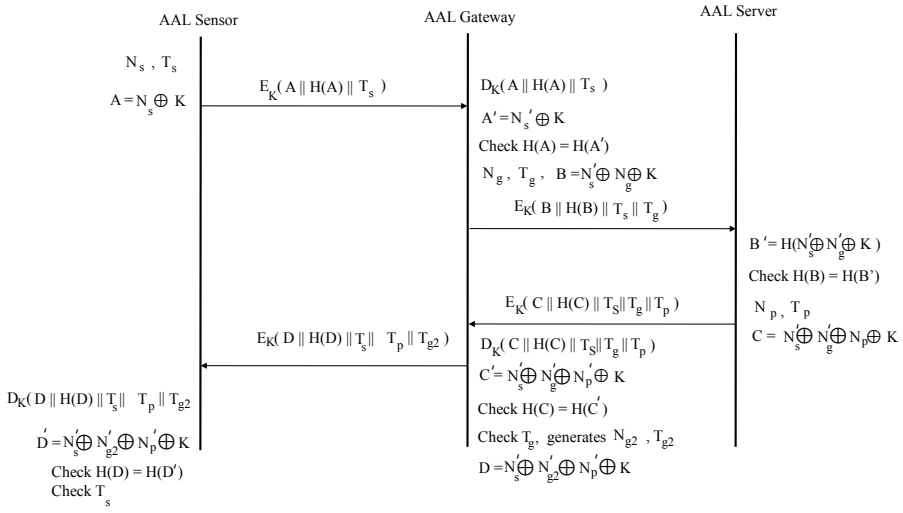


**Figure 3** Authentication phase.

## 4.2 Authentication

The AAL sensor, AAL gateway, and AAL server authenticate each other based on the common session key, $K$, from the registration phase. As shown in Figure 3, the authentication phase is divided into the following three rounds:

**Round 1:**

- Step 1. An AAL sensor randomly selects a nonce value, $N_s$, and generates a timestamp, $T_s$.
- Step 2. The AAL sensor computes $A = N_s \oplus K$ and $H(A) = h(N_s \oplus K)$.
- Step 3. The AAL sensor uses a symmetric key algorithm to encrypt $\{A \parallel H(A) \parallel T_s\}$ with $K$ and sends it to the AAL gateway.

**Round 2:**

- Step 1. When receiving the message, the AAL gateway extracts $\{A \parallel H(A) \parallel T_s\}$ using its $K$ to decrypt the received ciphertext and computes $N_s{}' = A \oplus K$ and $H(A^{'}) = h(N_s^{'} \oplus K)$.
- Step 2. The AAL gateway checks the value of $H(A) = H(A^{'})$. If the two values are equal, the AAL gateway confirms that the AAL sensor is a valid user. Otherwise, the AAL gateway stops the protocol and sends an authentication-failed message to the AAL sensor.

- Step 3. If the AAL sensor is a valid user, the AAL gateway selects the nonce value, $N_g$, and generates the timestamp, $T_g$.
- Step 4. The AAL gateway computes $B = N_s^{'} \oplus N_g \oplus K$ and $H(B) = h(N_s^{'} \oplus N_g \oplus K)$.
- Step 5. Finally, the AAL gateway uses a symmetric key algorithm to encrypt $\{B \parallel H(B) \parallel T_s \parallel T_g\}$ with $K$ and sends it to the AAL gateway.

**Round 3:**

- Step 1. Upon receiving the messages, the AAL server extracts $\{B \parallel H(B) \parallel T_s \parallel T_g\}$ using its $K$ to decrypt the received ciphertext and computes $N_s^{'} \oplus N_g^{'} = B \oplus K$ and $H(B^{'}) = h(N_s^{'} \oplus N_g^{'} \oplus K)$.
- Step 2. The AAL server checks the value of $H(B) = H(B^{'})$. If the two values are equal, the AAL server confirms that the AAL gateway is a valid user. Otherwise, the AAL server stops the protocol and sends an authentication-failed message to the AAL gateway.
- Step 3. If the AAL gateway is a valid user, the AAL server generates a timestamp, $T_p$, and selects the nonce value $N_p$.
- Step 4. The AAL server then computes $C = N_s^{'} \oplus N_g^{'} \oplus N_p \oplus K$ and $H(C) = h(N_s^{'} \oplus N_g^{'} \oplus N_p \oplus K)$.
- Step 5. Finally, the AAL server uses a symmetric key algorithm to encrypt $\{C \parallel H(C) \parallel T_s \parallel T_g \parallel T_p\}$ with $K$ and sends it to the AAL gateway.

**Round 4:**

- Step 1. When receiving the messages, the AAL gateway extracts $\{C \parallel H(C) \parallel T_s \parallel T_g \parallel T_p\}$ using its $K$ to decrypt the received ciphertext and computes $C^{'} = C \oplus K$ and $H(C)^{'} = h(N_s^{'} \oplus N_g^{'} \oplus N_p^{'} \oplus K)$.
- Step 2. The AAL gateway then checks $H(C) = H(C^{'})$ and aborts if two values are not equal.
- Step 3. The AAL gateway checks the freshness of $T_g$ and aborts if the check fails.
- Step 4. Otherwise, the AAL gateway generates a new nonce value, $N_{g2}$, and a new timestamp, $T_{g2}$. The AAL gateway then computes $H(D) = h(N_s^{'} \oplus N_{g2}^{'} \oplus N_p^{'} \oplus K)$.
- Step 5. Finally, the AAL gateway uses a symmetric key algorithm to encrypt $\{D \parallel H(D) \parallel T_s \parallel T_{g2} \parallel T_p\}$ with $K$ and sends it to the AAL sensor.

**Round 5:**

- Step 1. When receiving the messages, the AAL sensor extracts $\{D \parallel H(D) \parallel T_s \parallel T_g2 \parallel T_p\}$ with $K$ to decrypt the received ciphertext and computes $D^{'} = D \oplus K$ and $H(D^{'}) = h({N_s}^{'} \oplus {N_{g2}}^{'} \oplus N_p^{'} \oplus K)$.
- Step 2. The AAL sensor checks $H(D) = H(D)^{'}$ and aborts if the two values are not equal.
- Step 3. The AAL sensor checks the freshness of $T_s$ and aborts if the check fails.

After all these steps in the authentication phase are successfully completed, reauthentication can be performed directly from Round 1 with new values of $T_{s2}$ and $N_{s2}$.

## 5 Security Analysis and Performance Evaluation

In this section, we present how our proposed authentication scheme satisfies the previously mentioned security requirements. We then evaluate its the performance using based on the experimental results.

### 5.1 Security Analysis

In this section, we show that our proposed scheme can provide mutual authentication, session-key agreement, and perfect forward secrecy. We also show that the proposed scheme can withstand man-in-the-middle attacks, replay attacks, and mutual authentication.

- Mutual authentication: Our proposed scheme achieves mutual authentication among the AAL sensor, AAL gateway, and AAL server. After registration, the AAL sensor, AAL gateway, and AAL servers share the same temporary session key that is known only by them. With the assistance of a temporary key, the AAL servers authenticate each other. At the end of the authentication, they create a new temporary session key, and the previous one is removed.
- Session-key agreement: In our proposed scheme, the AAL sensor data transmitted to the AAL sensor, AAL gateway, and AAL server should be encrypted using the session key shared between them. Therefore, a shared session key must be generated during the mutual authentication process. Thus, the proposed scheme provides session-key agreement.
- No verification table: The proposed scheme does not require a verification table for authentication purposes. Thus, the proposed authentication

scheme has no problems with the drawbacks of maintaining a verification table.

- Perfect forward secrecy: Our proposed scheme provides strong forward secrecy. In our protocol, the established session key is $K$, where $p$, $q$, and $r$ are random numbers selected by the AAL sensor, AAL gateway, and AAL servers, respectively. Previously established session keys remain secure even when the long-term keys of the server and the user are disclosed because it is computationally infeasible for an adversary to calculate the session key without an elliptic-curve point.
- Attack resistance: Protection against replay attacks of the AAL system with our proposed scheme is achieved using timestamps. In our proposed scheme, the timestamp mechanism is included in each message. The AAL sensor, AAL gateway, and AAL server can detect the replay of a message by checking the freshness of the timestamp. Therefore, the proposed authentication protocol can withstand a replay attack. If the adversary carries out a man-in-the-middle attack, the adversary needs to choose a nonce value and compute a hashed value. However, the adversary cannot obtain the value of the common session key. Moreover, our proposed scheme provides mutual authentication among the AAL sensor, AAL gateway, and AAL servers. Therefore, our proposed scheme can withstand a man-in-the-middle attack.

Our proposed scheme thus provides a highly effective and perfectly robust forward secrecy property for mutual authentication. Results of comparison between proposed protocols and related works in terms of security requirements are shown in Table 2.

**Table 2** Comparison between our protocol and other authentication protocol in terms of security requirements

| | Yeah et al. | Lui et al. | Zao's et al. | He et al. | Our Scheme |
|---|---|---|---|---|---|
| $MutualAuthentication$ | ◯ | ◯ | ◯ | ◯ | ◯ |
| $Anonymity$ | ◯ | ◯ | ◯ | ◯ | ◯ |
| $Nontraceability$ | ◯ | ◯ | × | × | ◯ |
| $NoVerificationtable$ | ◯ | ◯ | ◯ | ◯ | ◯ |
| $SessionKeyAgreement$ | ◯ | ◯ | ◯ | ◯ | ◯ |
| $PerfectForwardSecrecy$ | ◯ | ◯ | ◯ | ◯ | ◯ |
| $Computationallyfeasibility$ | × | × | ◯ | ◯ | ◯ |

### 5.2 Performance Evaluation

In this section we analyze the performance of the proposed scheme for the AAL system. Similar to [8], we found that several authentication schemes for the WBAN environment could be applied to the AAL system after some modifications. Therefore, we compare the proposed scheme's computational cost in terms of execution time to execute various operations with those of existing schemes. In Table 3, we define some notations used in the performance evaluation.

Using the experimental results obtained in [6–8], we can calculate the following:

$$T_h \cong 0.4T_{mm} \tag{1}$$

$$T_{sym} \cong 0.4T_{mm} \tag{2}$$

$$T_{exp} \cong 240T_{mm} \tag{3}$$

$$T_{ecsm} \cong 29T_{mm} \tag{4}$$

$$T_{pair} \cong 620T_{mm} \tag{5}$$

$$T_{xor} \cong 12T_{mm} \tag{6}$$

From Equations (1), we can evaluate the computation cost in terms of execution time and compare our proposed scheme with the existing schemes. The computational costs of the proposed authentication scheme and the existing authentication schemes are listed in Table 3. In the proposed authentication scheme, the AAL system executes four hash function operations, eight symmetric encryption or decryption operations, and eight bitwise operations. Thus, the execution time of the AAL system is $4T_h + 8T_{sym} + 8T_{xor} \approx 100.8T_{mm}$. We can see that the proposed authentication scheme has

**Table 3** Major notations for performance evaluation

| Denotation | Description |
|---|---|
| $T_h$ | Execution time of one hash function operation |
| $T_{sym}$ | Execution time of one symmetric encryption or decryption operation |
| $T_{mm}$ | Execution time of one modular multiplication |
| $T_{exp}$ | Execution time of one modular exponentiation operation |
| $T_{ecc}$ | Execution time of one elliptic-curve scale multiplication |
| $T_{pair}$ | Execution time of one bilinear pairing operation |
| $T_{xor}$ | Execution time of one bitwise XOR operation |

**Table 4** Computational cost comparisons

| | Total Computational Cost | $T_{mm}$ |
|---|---|---|
| Liu et al. | $6T_h + 2T_{sym} + 4T_{ecsm} + 1T_{exp} + 1T_{pair}$ | $1013.6T_{mm}$ |
| Zao et al. | $9T_h + 2T_{sym} + 9T_{ecsm}$ | $265.4T_{mm}$ |
| He et al. | $4T_h + 8T_{sym} + 6T_{ecsm}$ | $178.8T_{mm}$ |
| Our scheme | $4T_h + 8T_{sym} + 8T_{xor}$ | $100.8T_{mm}$ |

a lower computational cost in terms of execution time than those of existing authentication schemes.

### 5.3 Simulation for Formal Security Verification

In this section, we describe the simulation of our scheme for formal security verification using the widely-accepted AVISPA tool [10, 11]. AVISPA is used for the automated validation of Internet security-sensitive protocols and applications. The tool measures whether the security protocol is safe or unsafe according to specified goals and is supported by a High Level Protocol Specification Language (HLPSL). We declared the required security properties such as secrecy and authentication in HLPSL as shown in Figures 4–6. The results indicate that our scheme is secure against passive and active attacks.

We declared the required security properties such as secrecy and authentication in HLPSL Firstly, the roles of the AAL sensor are defined in HLPSL language as shown in Figure 4. The AAL sensor has public parameters with the created channels, i.e., Sdn and Rcv, for sending and receiving, respectively. It also has local variables. It has some constant variables to declare the goals of the protocol. The process is initiated by the AAL sensor. At state $= 0$, the AAL sensor receives a start command from the AVISPA. At the end of the process, the requested and secret keywords can be seen. These keywords enable us to check the authentication of the AAL sensor to the AAL gateway and the confidentiality of the shared session key, respectively. As shown in Figure 5, the role of the AAL gateway are defined in HLPSL language. Their parameters are almost similar to Figure 4. The AAL gateway starts the transition by receiving a message from the AAL sensor. Finally, it also checks for strong authentication and the confidentiality of the shared key. Similar to Figures 7 and 8, we set the roles between the AAL gateway and the AAL server. As shown in Figure 6, the roles of session and environment are defined in HLPSL language. The environment role involves general composition and

```
role AALsensor (
        C, S     : agent,     % C client, S server
        H        : hash_func, % HMAC hash func.
        Ks       : text,      % Kg is the pre-existing shared secret
        G   : text,   % Generator of ECC
        Snd, Rcv : channel(dy))
played_by C def=

local State      : nat,
     Time1       : text,   % Time stamp
      Rp     : text,   % Random number for ECC
      EpC    : message,      % Elliptic point exp(text,text)
      K      : text,   % Shared session key
      p      : text,   % Public key of AALgateway
      Sig    : hash(agent.message.text.text)

  const
        succ                 : hash_func,   % Successor function
        sec_k                : protocol_id
        AALsensor_AALgateway: protocol_id
  init  State := 0
  transition
  State = 0
      /\ Rcv(start)
      =|>
      State' := 1
      /\ Time1' := new()
      /\ Rp     := new()
      /\ EpC'   := exp(Rp',G)
      /\ Sig'   := H(C.EpC'.Time1'.K)
      /\ Snd(C.S.Time1'.Sig'.EpC')
      /\ witness(C,S, AALsensor_AALgateway_,Sig')
  State = 1
      /\Rcv(C.S.succ(Time1').Sig'.Ps')
      =|>
      State' := 2
      /\ SK'  := exp(q',G)
      /\ Sig' := H(C.K'.succ(Time1).Ks)
      /\ request(C,S, AALsensor_AALgateway_sig,Sig')
      /\ secret(K,sec_sk,{C,S})

end role
```

**Figure 4** Role of AAL sensor in AVISPA.

```
role AALgateway (
        S,C      : % C client, S server
        H        : hash_func,    % HMAC hash func.
        Ks       : text, % Kg is the pre-existing shared secret
        G   : text,   % Generator of ECC
        Snd, Rcv : channel (dy))
played_by S def=

local State : nat,
     Time1 : text,
     Rq    : text,         % Random number for ECC
     K     : text,         % Shared session key
     q    : text           % Private key of AALserver
     Sig   : hash(agent.protocol_id.hash(text).text)

  const
       sec_k                 : protocol_id,
       AALgateway_AALsensor_sig: protocol_id,
       succ                  : hash_func,     % Successor function

  init State := 0

  transition
     State = 0
          /\Rcv(C.S.Time1'.Sig'.EpC') =|>
     State'= 1
          /\K'  := exp(q',EpC)
          /\Ps' := exp(q',G)
          /\Sig'     := H(S.Rq'.succ(Time1').K')
          /\Snd(C.S.succ(Time1').Sig'.Rq')
          /\witness(S,C, AALgateway_AALsensor_sig)
               /\ request(S,C, AALgateway_AALsensor_sig,Sig')
               /\ secret(K,sec_sk,{C,S})

end role
```

**Figure 5** Role of AAL gateway in AVISPA.

the initial knowledge of the intruder. From the initial knowledge, the intruder will attempt to attack. The environment role also includes the goal of the proposed scheme. The Constraint-Logic-based Attack Searcher (CL-AtSe) is built in a modular way in the AVISPA. It supports type-flaw detection

```
role session(C, S  : agent,
          H : hash_func,
          G : text )
def=
local CS,CR,SS,SR:channel(dy)
  composition
     AALLgateway(S,C,H,G,SS,SR) /\
     AALsensor(C,S,H,G,CS,CR)

end role
```

```
role environment()
local Snd, Rcv: chanell(dy)
def=

 const a, b, i       : agent,
       k1, k2        : text,
       h             : hash_func,
 AALsensor_AALgateway_sig, sec_sk  : protocol_id

 intruder_knowledge = {a,b,i,
                       h,succ,g
                       }
 composition
       session(a,b,h,k1)
    /\  session(a,i,h,k2)
    /\  session(i,b,h,k3)

end role
```

**Figure 6** Session and environment roles in AVISPA.

and handles associativity of message concatenation. We have used the CL-AtSe backends of the AVISPA framework. As shown in Figure 7, the results show that the protocol is safe using CL-AtSe backends, which means that the protocol meets the specified goal successfully. From the formal analysis aspect also, it can be seen that the proposed scheme satisfies the necessary security properties of the AAL system.

```
SUMMARY
  SAFE

DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
  TYPED_MODEL

PROTOCOL
  /home/span/span/testsuite/results/aal_smart_authen.if

GOAL
  As Specified

BACKEND
  CL-AtSe

STATISTICS

  Analysed   : 85 states
  Reachable  : 45 states
  Translation: 0.00 seconds
  Computation: 0.00 seconds
```

**Figure 7** Representative results in AVISPA.

## 6 Conclusion

Owing to rapidly decreasing birthrates, most countries are facing the problem of an aging population. As a result, research on aging and the means to support an aging population has thus become a priority for many governments around the world. AAL technology is of considerable interest for supporting the independence and quality of life of elderly people. Because the information transmitted in AAL systems is highly personal, security and privacy of such data are becoming important issues that must be addressed. Herein, we propose an efficient and secure authentication scheme for WBANs based AAL system. Our proposed authentication scheme not only supports several important security requirements of the AAL system but also withstands various types of attacks. Moreover, the proposed authentication scheme achieves lightweight performance by manipulating basic cryptographic operations including bitwise-exclusive-OR (XOR) and hash functions. The security analysis and simulation results obtained using the AVISPA tool show that the proposed scheme is secure and efficient compared with the state-of-the-art authentication schemes for WBANs based AAL system.

## Acknowledgement

This work was supported by the GRRC program of Gyeonggi province. [GRRC-Gachon2021(B04), Development of AI-based Healthcare Devices]

## References

[1] Parisa Rashidi and Alex Mihailidis. A Survey on Ambient-Assisted Living Tools for Older Adults. *IEEE Journal of Biomedical and Health Informatics*. 2013, 17(3) 579–590.

[2] Bhawna Narwal Amar and Kumar Mohapatra. A survey on security and authentication in wireless body area networks. *Journal of Systems Architecture*. 2021, 113.

[3] Bacem Mbarek et al. An Efficient Mutual Authentication Scheme for Internet of Things. *Internet of Things*. 2020, 9.

[4] Yanrong Lu et al. An Enhanced Biometric-Based Authentication Scheme for Telecare Medicine Information Systems Using Elliptic Curve Cryptosystem. *Journal of Medical Systems*. 2015, 39(32).

[5] C. Yeh et al. An Authentication Protocol for Ubiquitous Health Monitoring Systems *J. Medical and Biological Engineering*. 2013, 33(4).

[6] J. Liu et al. Certificateless Remote Anonymous Authentication Schemes for Wireless Body Area Networks *IEEE Trans. Parallel Distrib. Syst.*. 2014, 25(2):332–342.

[7] Z. Zhao. An Efficient Anonymous Authentication Scheme for Wireless Body Area Networks Using Elliptic Curve Cryptosystem *J. Medical Systems*. 2014, 38(2):1–7.

[8] D. He and S. Zeadally. Authentication protocol for an ambient assisted living system *IEEE Commun. Mag.*. 2015, 53(1):71–77.

[9] Paolo Bellagente et al. Framework-Oriented Approach to Ease the Development of Ambient Assisted-Living Systems *IEEE Systems Journal*. 2019, 13(4):4421–4432.

[10] D von Oheimb D. The high-level protocol specification language HLPSL developed in the EU project AVISPA. Proceedings of APPSEM 2005 Workshop. 2005.

[11] AVISPA: Automated Validation of Internet Security Protocols and Applications. http://www.avispa-project.org/

## Biographies

**Myung-Kyu Yi**. He received the Ph.D. degree in Computer Science and Engineering from Korea University in 2005. He is currently an Research Professor with Gachon University. His research interests include heathcare, security, machine learning and deep learning, human activity recognition.

**Taeg-Keun Whangbo** received the M.S. degree from City University of New York in 1988 and the Ph.D. degree both in Computer Science from Stevens Institute of Technology in 1995. Currently, he is a professor in the Department of Computer Science, Gachon University, Korea. He is also the Vice President in Gachon University. Before he joined the Gachon University, he was the software developer in Q-Systems which is located in New Jersey from 1988 to 1993. He was also the researcher in Samsung Electronics from 2005 to 2007. His research areas include Computer Vision, Artificial Intelligence, Healthcare, HCI and AR/VR.