*electronics*

Article

# An Energy Efficient and Formally Secured Certificate-Based Signcryption for Wireless Body Area Networks with the Internet of Things

**Insaf Ullah [1], Abdullah Alomari [2,*], Noor Ul Amin [3], Muhammad Asghar Khan [1] and Hizbullah Khattak [3]**

[1] HIET, Hamdard University Karachi, Islamabad Campuse, Islamabad 44000, Pakistan; insafktk@gmail.com (I.U.); khayyam2302@gmail.com (M.A.K.)

[2] Department of Computer Science, Faculty of Computer Science and Information Technology, Albaha University, Albaha 65799, Saudi Arabia

[3] IT Departement, Hazara University, Mansehra 21120, KP, Pakistan; namin@hu.edu.pk (N.U.A.); hizbullahkhattakk@yahoo.com (H.K.)

* Correspondence: amalomari@bu.edu.sa; Tel.: +966-17-725-7700

Received: 12 September 2019; Accepted: 12 October 2019; Published: 15 October 2019

**Abstract:** Recently, the spectacular innovations in the fields of wireless body area networks (WBAN) and the Internet of Things (IoT) have made e-Care services rise as a promising application domain, which significantly advances the quality of the medical system, however, due to the openness of the wireless environment and privacy of people's physiological data, WBAN and IoT are prone to various cyber-attacks. There is a significant need for an efficient and highly secured cryptographic scheme that can meet the requirements of resource-constrained devices. Therefore, in this paper, we propose a certificate-based signcryption (CB-SN) scheme for the IoT-enabled WBAN. The proposed scheme is based on the concept of hyper-elliptic curve cryptography (HECC) that offers the same level of security as the elliptic curve and bilinear pairing with lower-key size. The formal security verification using the Automated Validation of the Internet Security Protocols and Applications (AVISPA) tool along with informal security analysis demonstrate that the proposed scheme is not just reducing the complexity of resource-constrained IoT devices, but proves to be secure against several well-known cryptographic attacks. Moreover, performance comparison with relevant existing schemes authenticates that the proposed scheme is far more secure and energy efficient.

**Keywords:** WBAN; IoT; certificate-based signcryption; AVISPA; hyper-elliptic curve

## 1. Introduction

In the current era, the Internet of things (IoT) is one of the most debatable topics among the research community of information technology. The IoT includes all those devices, which have the capacity of computing, communication, and connection with the Internet [1]. The IoT has so many applications in our daily lives, i.e., it is used in smart cities, smart homes, and e-health, etc. [2]. By providing faster access to the treatment of patients, the wireless body area networks (WBAN) integrate with the Internet of things (IoT) and play an important role in the patient health care system, because this ecosystem, enables all the users and devices to access the patient's psychological data from anywhere and anytime in the world by utilizing the Internet [3]. However, due to the open nature of the Internet, authenticity and data security are the two major concerns in the IoT based WBAN [4].

The authenticity of IoT is ensured through digital signature [5], and data security is met by using the encryption method [6], although the IoT has a resource hungry nature and cannot afford these two different algorithms separately, i.e., signature and then encryption at the same time. In 1997, Zheng was the pioneer to merge these two processes in one algorithm, called signcryption [7]. This scheme is based on the concept of old public key cryptography (PKC), which is suffering from certificate overheads, renewing, and revocation problems [8]. Shamir was the first to propose an alternate concept of PKC, called identity-based cryptography (IBC) [9]. This technique removed the limitations of PKC and used the identity in place of a certificate. Later, in 2002, Malone-Lee [10], for the first time merged the concept of IBC with the signcryption technique, namely, identity-based signcryption (IBS). The IBS includes three entities, for example, a sender (signcrypter), a receiver (unsigncrypter), and the private key generation center (PKGC), respectively. In this setup, the users (signcrypter and unsigncrypter) generate their identities and after that, send it to the PKGC. Then, the PKGC produces and delivers the private keys for all the participating users, by using the secured networks. Unfortunately, IBS suffers from the key escrow issue (KEI), because the private key is generated by the PKGC and one can easily use this key for forging the digital signature and decrypting the ciphertext [11].

To eliminate the above problem in IBS, in 2008, Barbosa and Farshim [12], put forward the concept of a certificateless signcryption (CL-SC) scheme. The CL-SC mechanism almost works the same as IBS, but the main difference is that the private key is generated by the users themselves. The central authority known as a key generation center (KGC) only provides the partial private key to the users by using an open link. Although it removes the issue of key escrow in IBS and certificate management in PKC-based signcryption, it still suffers from the needs of the partial private key distribution problem [13]. Another strategy, named heterogeneous signcryption was proposed by Sun [14]. This strategy contains two sub-methods, the first one works under the condition in which the sender belongs to the conventional PKC and the receiver belongs to the IBC, while in the second one, a sender uses the concept of IBC and receiver based on the PKC. Since, these two types (PKC and IBC) of the public key are suffering from some crucial problem, i.e., certificate overheads, renewing, certificate revocation, and KEI, respectively, these types of problems are not suitable for the IoT environment. To cater to this particular issue, heterogeneous signcryption was coined by Li et al. [14], in which the signcryption part belongs to the certificateless cryptosystem (CLC) and the unsigncryption side is based on the functionality of PKC, however, the scheme is affected by the secrete key distributions and certificate management issues. To remove the certificate management at the receiver side, Omala et al. [15], contributed a new heterogeneous signcryption scheme, in which the signcryption part belongs to the CLC and the unsigncryption part works under the notion of IBC. This method is also affected by the key escrow and the secret key distribution problem. A new type of cryptosystem was introduced by Gentry in 2003 [16], namely, certificate-based cryptography (CBC), in which one can use the functionality of old PKC in a better manner. The CBC enables each participant in the network to generate his public and private keys and give their public key to the certifier's authority (CsA). Later, by using the concept of IBC encryption, based on the participants' public key which serves its identity, CsA generates a certificate for each participant while using an open link. Noteably, this certificate acts as a partial private key and also uses a decryption key on the receiver side [17]. In 2008, Li et al. [18], provided a new scheme, which is used to merge the concept of CBC with signcryption, called certificate-based signcryption. In 2019, Braeken proposed pairing free certificate-based signcryption schemes using ECQV implicit certificates [19]. However, the proposed approach is based on a hyper-elliptic curve, i.e., it suffers from high computational cost. Moreover, the proposed scheme is not validated through any formal security tool.

Cagalaban and Kim [20], proposed an effective signcryption scheme for access control in the WBAN under the functionalities of IBC, which is suffering from KEI. Similarly, Hu et al. [21], proposed an access control for WBAN using the idea of fuzzy attribute-based signcryption, however, the proposed scheme suffered from high computational cost. In 2016, Li and Hong [22], proposed a signcryption scheme for access control in WBAN while utilizing both CLC and bilinear pairing (BP). In 2018, Li et al. [23], presented a CL-SC approach for an efficient access control in WBAN.

These approaches [22,23], faced the issues of secrete key exchange and extra energy consumption. In the same year, Omala et al. [15], by using signcryption, designed an access control scheme for WBAN, where they used CLC in the signcryption part and IBC in the unsigncryption part. Recently, in May 2019, Gao et al. [24], developed a CL-SC with an elliptic curve for secure and efficient access control in WBAN. Nevertheless, these two schemes [15,24] are commonly affected by the secret key distribution problems, more energy utilization, and extra bandwidth consumption.

### *1.1. Authors' Motivations and Contributions*

The authors, motivated by the aforementioned limitations regarding signcryption-based access control in WBAN, propose a new scheme, called an energy efficient and formally secured certificate-based signcryption (CB-SN), which does not suffer from the problems such as secret key distribution problems, more energy utilization, and extra bandwidth consumption. Some of the salient features signifying contributions of our research work, in this paper, are as follows:

1. We first provide the basic syntax for certificate based signcryption and then construct the scheme practically for WBAN with IoT;
2. The proposed scheme is shown to be resistant against various attacks through informal security analysis concerning integrity, confidentiality, replay, unforgeability, and forward secrecy, respectively;
3. We also generate the high level protocol specification language (HLPSL) code for our scheme in AVISPA Tool for the formal security checking, and the simulation results authenticates that the proposed scheme is SAFE, according to the checking structure of two well-known checker models, i.e., on-the-fly model checker (OFMC) and constraint logic-based attack searcher (ATSE);
4. We perform the computational cost and communication overhead comparison analysis with the relevant existing schemes, which demonstrates the presented scheme, in addition, is far more efficient.

### *1.2. Structure of The Paper*

The remainder of the paper is organized as follows: Section 2 gives the basic knowledge of preliminaries, Section 3 presents proposed architecture, Section 4 contains the construction of the proposed scheme, Section 5 presents the informal security analysis, Section 6 give the proposed scheme implementation detail in WBAN, Section 7 delivers implementation of the proposed scheme in AVISPA, and includes the discussion about performance with relevant existing schemes, and, finally, Section 8 culminates conclusions of the entire work.

## 2. Preliminaries

### *2.1. Hyper-Elliptic Curve*

This section briefly discusses the basic mathematics of a hyper-elliptic curve ($h\varepsilon c$). Suppose $\mathfrak{I}_t$ is a predetermined set and presume $\partial$ is the genus of $h\varepsilon c$ having order as $\partial \geq 2$. Let $(v)$, $\mathrm{f}(v) \varepsilon \mathfrak{I}_t[v]$, and deg $(h(v)) \preceq \partial$; and $\mathrm{f}(v)$ is a monic polynomial having deg $(\mathrm{f}(v)) = 2\partial + 1$ [25]. Therefore, $h\varepsilon c$ of genus $\partial \geq 2$ over $\mathfrak{I}_t$ is the set of points $(v,)$ $\mathfrak{I}_t * \mathfrak{I}_t$ as shown in the Equation (1).

$$h\varepsilon c\text{: } w^2 + (v)\, w = \mathrm{f}(v) \tag{1}$$

Note, $h\varepsilon c$ points are different from elliptic curve [26]. It forms the divisors which are the formal sum of finite integers like $d = \sum x_{\mathrm{i}} z_{\mathrm{i}}$, where $x_{\mathrm{i}} \varepsilon\ \mathfrak{I}_t$ and $z_{\mathrm{i}} \varepsilon\ h\varepsilon c$. Furthermore, it forms a Jacobian group $\mathscr{I}_{h\varepsilon c}\ (\mathfrak{I}_t)$ having the following order:

$$(\surd t - 1)^{2\partial} \leq \mathscr{I}_{h\varepsilon c}\ (\mathfrak{I}_t) \leq (\surd t + 1)^{2\partial} \tag{2}$$

### 2.2. Hyper-Elliptic Curve Discrete Logarithm Problem ($h\varepsilon c - dlP$)

Assume $d$ is the divisor which ispublicly available in the network and $\mathscr{L}$ is the randomly picked private number from $\mathfrak{I}_t$. Recovering $\mathscr{L}$ from $d_1 = d, \mathscr{L}$ is said to be $h\varepsilon c - dlP$ [27].

### 2.3. Automated Validation Tool for Security Validation and Application (AVISPA)

Currently, AVISPA is the utmost valuable tool among researchers of information security, in which they check the authenticity of their newly designed cryptographic protocol security properties. For the overall structure of AVISPA, it can be a better choice to see the article [24], in which it includes the most expensive formal language called high level protocol specification language (HLPSL), a compilation translator called HLPSL2IF, an intermediate format (IF), and the four backends models for checking the security properties, i.e., on-the-fly model checker (O-f-M-C), constraint logic-based attack searcher (CL-ATSe), SAT-based model checker (Sat-M-C), and tree automata based on automatic approximations for the analysis of security protocols (TA-4 SP), respectively. If the cryptographic user famines to trial their approach security, then he/she first compose the HLPSL code for their approach in SPAN, which is the graphical user interface (GUI) for AVISPA, furthermore, the HLPSL2IF is responsible for compiling this code into IF, and IF handover the code to the four model checkers, for example, (O-f-M-C), (CL-ATSe), (SAT-M-C), and (TA-4 SP) for the checking of man-in-the-middle attack and replay attack. Therefore, if these two attacks are possible in their respective protocol then these model checkers give UNSAFE simulation results and if it is not possible then it shows the SAFE results.

### 2.4. Syntax of Certificate-based Signcryption (CB-SN)

The proposed CB-SN scheme is the improved version of the Braeken et al. [17] scheme, and includes five algorithms such as, setup, public variable generation (PVG), certificate generation (CG), actors key generation (AKG), signcryption generation (SG), and unsigncryptions (US), respectively. We explain the syntax of CB-SN in the following steps:

1. Setup: Certifier's authority ($C_sA$) recognizes a security parameter $\mu$ as input data and runs the setup algorithm to make essential parameters set $\{h\varepsilon c$, δ, $\mathfrak{I}_t$, $d$, $\mathscr{I}_{h\varepsilon c}$, T, $\mathscr{H}_1$, $\mathscr{H}_2$, $\mathscr{H}_3\}$ and a master secret key Υ and public key T = Υ.$d$, respectively. The essential parameters set is directly accessible on a network, anyway Υ is kept by the $C_sA$ secret.
2. Public variable generation (PVG): Each actor with identity $ID_A$, runs the PVG algorithm to produce his public variable $PV_A$ by taking input, an essential parameter set $\{h\varepsilon c$, δ, $\mathfrak{I}_t$, $d$, $\mathscr{I}_{h\varepsilon c}$, T, $\mathscr{H}_1$, $\mathscr{H}_2$, $\mathscr{H}_3\}$. Then, the actor having identity $ID_A$, has sent the pair ($PV_A$, $ID_A$) to $C_sA$ via an insecure link.
3. Certificate generation (CG): By taking input ($PV_A$, $ID_A$), the essential parameter set $\{h\varepsilon c$, δ, $\mathfrak{I}_t$, $d$, $\mathscr{I}_{h\varepsilon c}$, T, $\mathscr{H}_1$, $\mathscr{H}_2$, $\mathscr{H}_3\}$ and Υ, $C_sA$ run the CG algorithm to produce a certificate $Cert_A$ for each actor with identity $ID_A$ and hand over a certificate with auxiliary variable ($\mathscr{C}ert_A$, $aux_A$) to the actors via unsecured link.
4. Actor's key generation (AKG): Inputting the pair of an auxiliary variable ($Cert_A$, $aux_A$), an essential parameter set $\{h\varepsilon c$, δ, $\mathfrak{I}_t$, $d$, $\mathscr{I}_{h\varepsilon c}$, T, $\mathscr{H}_1$, $\mathscr{H}_2$, $\mathscr{H}_3\}$ and the identity $ID_A$ of each actor, the participated actor's with identity $ID_A$ produces his private and public keys ($A_A$, $Q_A$).
5. Signcryption generation (SG): The SG algorithm is executed by the sender actor to produce signcryption text $\psi$ = {C,A,S}, of a message, m, and delivers $\psi$ to the receiver through an unsecured network. It takes as input the certificate of the sender and receiver ($Cert_s$, $Cert_u$), the identity of a sender and receivers ($ID_s$, $ID_u$), private and public key of the sender ($A_s$, $Q_s$), a message (m), an essential parameter set $\{h\varepsilon c$, δ, $\mathfrak{I}_t$, $d$, $\mathscr{I}_{h\varepsilon c}$, T, $\mathscr{H}_1$, $\mathscr{H}_2$, $\mathscr{H}_3\}$, and public key of receiver $Q_u$.
6. Unsigncryptions (US): The US algorithm is executed by the receiver actor to verify and decrypts the received signcryption text $\psi$ = {C,A,S}. It takes as input the certificate of the sender $Cert_s$,

the identity of a sender and receivers ($ID_s$, $ID_u$), private and public key of the receiver ($A_u$,$Q_u$), a signcryption text $\psi$ = {C,A,S}, and essential parameter set.

## 3. Proposed Architecture

Figure 1 indicates the overall working of a newly designed model of this paper, which includes three main actors, i.e., a certifiers authority (CsA), application providers (APs) and WBAN of a patient's body, respectively. Hence, in this model, it is the responsibility of CsA to create a certificate for APs and WBAN by using its own secret key and obtained identity with a public variable from actors (APs and WBAN). The APs are responsible for monitoring patient conditions and any time get access to the health related information HRI, by computing certificate-based signcryption of an access request query. The WBAN contains sensor nodes, which are already planted in the body of the patient and at least contain one controller, which receives PHRI. Upon the request from the access control query from APs, the controller checks the authorization of an actor and if the actor is legitimate, then, it sends the data regarding the query request, otherwise it rejects the query demand.

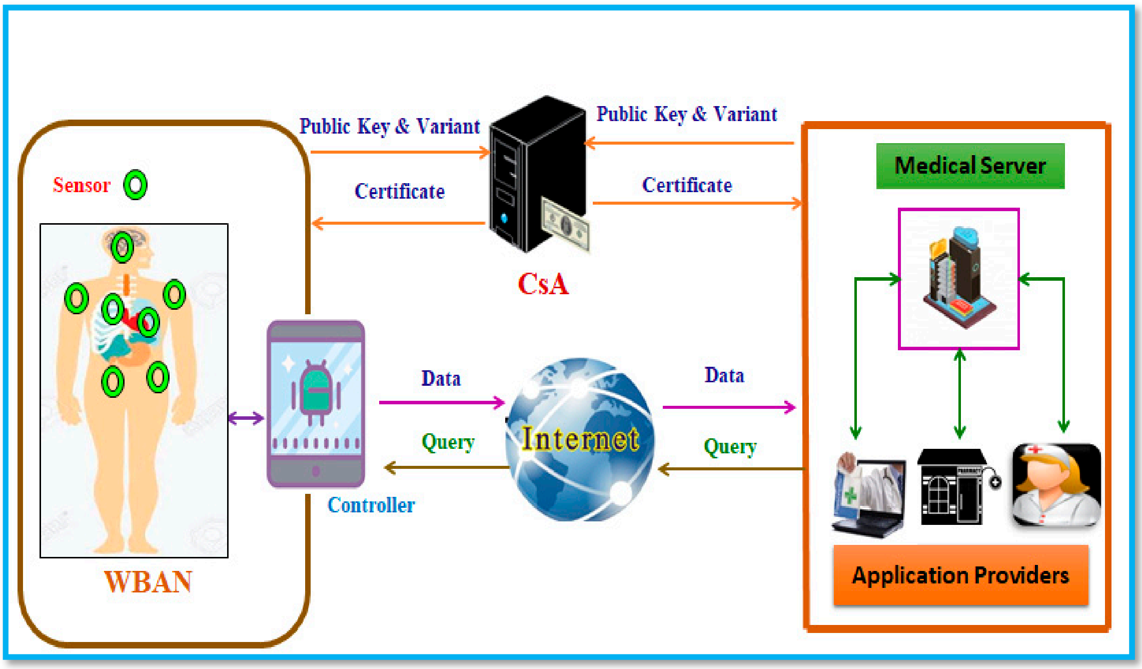


**Figure 1.** Proposed architecture.

Safe node and health node are the two wearable sensor nodes on each subject for environmental monitoring and for physiological parameters' measurements, respectively, in the proposed system. Furthermore, safe node is equipped with four environmental sensors to monitor the ambient temperature, relative humidity, CO2, and ultraviolet (UV) sensor. The health node is comprised of a Bluetooth 5 (802.15.1) module that is used to enable WBAN communication, a photoplethysmogram (PPG) sensor for heart rate monitoring, and a body temperature sensor. Bluetooth 5 (802.15.1) is considered the most favored option for wearable sensor nodes because of its low cost and low power consumption [28], however, to address the short communication range issue of Bluetooth 5 for medical records to a longer distance, a smart android-based mobile device, named "controller", is used inside the WBAN's communication range. At the end of this section, in Table 1, we provide an explanation about the symbols used in algorithm.

**Table 1.** Notations used in proposed algorithm.

| S.NO | Symbol | Explanation |
|---|---|---|
| 1 | $h\varepsilon c$ | Hyper-elliptic curve |
| 2 | $\partial$ | Genus of hyper-elliptic curve |
| 3 | $d$ | Divisor in hyper-elliptic curve |
| 4 | $\mathscr{I}_{h\varepsilon c}$ | Jacobian of hyper-elliptic curve |
| 5 | $\Upsilon$ | Master secret key |
| 6 | T | Master public key |
| 7 | $\mathscr{H}_1, \mathscr{H}_2, \mathscr{H}_3$ | Hash functions |
| 8 | $\mathscr{Q}_{\mathrm{s},}$ $\mathscr{Q}_\mathrm{u}$ | Public keys of sender and receiver |
| 9 | $\mathscr{A}_{\mathrm{s},}$ $\mathscr{A}_\mathrm{u}$ | Private keys of sender and receiver |
| 10 | $\mathscr{C}ert_{\mathrm{s},}$ $\mathscr{C}ert_\mathrm{u}$ | Certificates for sender and receiver |
| 11 | $\mathrm{ID}_\mathrm{s}$, $\mathrm{ID}_\mathrm{u}$ | Identities for sender and receiver |
| 12 | $\mathscr{SK}$ | Session secret key |
| 13 | N | A fresh nonce |
| 14 | $m/\mathscr{C}$ | Message/encrypted message |
| 15 | $\mathcal{E}_{\mathscr{SK}}/\mathscr{D}_{\mathscr{SK}}$ | Encryption/decryption |
| 16 | $\psi$ | Signcryption text |
| 17 | $\|$ | Concatenation |

## 4. Constructions of CB-SN

The proposed scheme is an extension of the scheme presented by Braeken et al. [17], and the working steps of the newly designed CB-SN scheme are as follows:

1. Setup: The $\mathrm{C_sA}$ produce an essential parameter set $\{h\varepsilon c, \delta, \mathfrak{I}_t, d, \mathscr{I}_{h\varepsilon c}, \mathrm{T}, \mathscr{H}_1, \mathscr{H}_2, \mathscr{H}_3\}$ and after that the $\mathrm{C_sA}$ select a master secret key Y{1,2, … , t – 1} and calculate the master public key T = Y.d, respectively. The essential parameters set is directly accessible on a network and the master secret key Y is kept by the $\mathrm{C_sA}$ secret.
2. PVG: Given an essential parameter set $\{h\varepsilon c, \delta, \mathfrak{I}_t, d, \mathscr{I}_{h\varepsilon c}, \mathrm{T}, \mathscr{H}_1, \mathscr{H}_2, \mathscr{H}_3\}$, each actor with identity $\mathrm{ID_A}$, choose a random number $\omega_\mathrm{A}$ and computes his public variable $\mathscr{PV}_\mathrm{A} = \omega_\mathrm{A}.d$. Then the actor with identity $\mathrm{ID_A}$ delivers ($\mathrm{PV_A}$, $\mathrm{ID_A}$) to the $\mathrm{C_sA}$ by using the open channel.
3. CG: Given an essential parameter set $\{h\varepsilon c, \delta, \mathfrak{I}_t, d, \mathscr{I}_{h\varepsilon c}, \mathrm{T}, \mathscr{H}_1, \mathscr{H}_2, \mathscr{H}_3\}$, public variable and identity of each actor ($\mathrm{PV_A}$, $\mathrm{ID_A}$) and master secret key Y, $\mathrm{C_sA}$ select a random number $\chi_\mathrm{A}\varepsilon$ {1, 2 … ., $t$ – 1} and calculate $\phi_\mathrm{A} = \chi_\mathrm{A}.d$. After that, $\mathrm{C_sA}$ computes the certificate $\mathrm{Cert_A} = \phi_\mathrm{A} + \mathrm{PV_A}$ and auxiliary variable $\mathrm{aux_A} = \mathscr{H}_1$ ($\mathrm{Cert_A}$, $\mathrm{ID_A}$). $\chi_\mathrm{A} + \Upsilon$, then, hands over a certificate $\mathrm{Cert_A}$ with auxiliary variable ($\mathrm{Cert_A}$, $\mathrm{aux_A}$) to the actors via insecure link.
4. AKG: Given the tuple ($\mathrm{Cert_A}$, $\mathrm{aux_A}$), identity $\mathrm{ID_A}$ of each actor, and essential parameter set $\{h\varepsilon c, \delta, \mathfrak{I}_t, d, \mathscr{I}_{h\varepsilon c}, \mathrm{T}, \mathscr{H}_1, \mathscr{H}_2, \mathscr{H}_3\}$, each actor makes their private key $\mathrm{A_A} = \mathrm{H_1}$ ($\mathrm{Cert_A}$, $\mathrm{ID_A}$). $\mathrm{w_A} + \mathrm{aux_A}$ make their public key as $\mathrm{Q_A} = \mathrm{A_A}$. d.
5. SG: Given an essential parameter set $\{h\varepsilon c, \delta, \mathfrak{I}_t, d, \mathscr{I}_{h\varepsilon c}, \mathrm{T}, \mathscr{H}_1, \mathscr{H}_2, \mathscr{H}_3\}$, the sender and receiver's certificates ($\mathrm{Cert_s}$, $\mathrm{Cert_u}$), the identity of the sender and the receiver ($\mathrm{ID_s}$, $\mathrm{ID_u}$), the private and public key of the sender ($\mathrm{A_s}$, $\mathrm{Q_s}$), a massage (m), and public key of receiver $\mathrm{Q_u}$. Then the sender produces a signcryption tuple $\psi$ = {C, A} by utilizing the following steps:

    - The sender first computes the public key of receiver $\mathrm{Q_u} \overset{?}{=} \mathrm{H_2}$ ($\mathrm{Cert_u}$, $\mathrm{ID_u}$). $\mathrm{Cert_u} + \mathrm{T}$;
    - Next, choose a random number $\Omega\varepsilon$ {1, 2, … , t – 1} and compute $\beta = \Omega\ .\ d$;
    - Select a fresh nonce N;
    - Compute the session key $\mathscr{SK} = \Omega\ .\ \mathscr{Q}_\mathrm{u}$ and produced the cipher text $\mathscr{C} = \mathcal{E}_{\mathscr{SK}}\ (m\|\ \mathrm{N})$;
    - Compute the hash value $\Lambda = \mathscr{H}_2(m\|\ \mathrm{N})$ and signature $\mathscr{S} = \Omega - \Lambda\ ,\ \mathrm{A}_{\mathrm{s};}$
    - Then, hand over $\psi = \{\mathscr{C}, \Lambda, \mathscr{S}\}$ to the receiver using insecure channel.

6. US: Given the sender's certificate $\mathrm{Cert_s}$, identity of a sender and receivers ($\mathrm{ID_s}$, $\mathrm{ID_u}$), private and public key of the receiver ($\mathrm{A_u}$,$\mathrm{Q_u}$), a signcryption text $\psi$ = {C, A, S}, essential parameter set

$\{h\varepsilon c$, δ, $\mathfrak{I}_t$, $d$, $\mathcal{I}_{h\varepsilon c}$, T, $\mathcal{H}_1$, $\mathcal{H}_2$, $\mathcal{H}_3\}$, and public key of sender $Q_s$. The receiver performs the following steps to verify and decrypts the received signcryption text $\psi$= {C. A, S}.

- The receiver first computes the public key of sender $Q_s \overset{?}{=} H_2$ ($\text{Cert}_s$, $\text{ID}_s$). $\text{Cert}_s$ + T;
- Compute $\beta$ = S . d + Λ . $Q_{s;}$
- Recover the secret key SK = $A_u$ . $\beta$ and produced the plaintext (m|| N) = $D_{\mathcal{SK}}$ (C);
- Compute the hash value $\Lambda^{/} = H_2(\text{m}\|\ \text{N})$;
- Accept the signcryption text $\psi = \{\mathcal{C},\Lambda,\mathcal{S}\}$ if $\Lambda^{/} = H_2(\text{m}\|\ \text{N}) \overset{?}{=} \Lambda = H_2(\text{m}\|\ \text{N})$.

### 4.1. Correctness

The sender can compute the public key of receiver from the following computations:

$\mathcal{Q}_u \overset{?}{=} \mathcal{H}_2\ (\mathcal{C}ert_u, \text{ID}_u).\ \mathcal{C}ert_u + \text{T}$

$= \mathcal{H}_2\ (\mathcal{C}ert_u, \text{ID}_u)\ .\ (\phi_u + \mathcal{PV}_u\ ) + \Upsilon$, where $\mathcal{C}ert_u = \phi_u + \mathcal{PV}_u$ and $\text{T} = \Upsilon.d$

$= \mathcal{H}_2\ (\mathcal{C}ert_u, \text{ID}_u)\ .\ (\phi_u + \omega_u.d) + \Upsilon.\ d$, where $\mathcal{PV}_u = \omega_u.d$

$= \mathcal{H}_2\ (\mathcal{C}ert_u, \text{ID}_u)\ .\ (\chi_u.d + \omega_u.d) + \Upsilon.\ d$, where $\phi_u = {}_u.d$

$= \chi_u\ .d.\ \mathcal{H}_2\ (\mathcal{C}ert_u, \text{ID}_u) + \omega_u.d.\ \mathcal{H}_2\ (\mathcal{C}ert_u, \text{ID}_u) + \Upsilon.d$

$= (\chi_u\ .\ \mathcal{H}_2\ (\mathcal{C}ert_u, \text{ID}_u) + \omega_u.\ \mathcal{H}_2\ (\mathcal{C}ert_u, \text{ID}_u) + \Upsilon).\ d$

$= (aux_u + \omega_u.\ \mathcal{H}_2\ (\mathcal{C}ert_u, \text{ID}_u))$, where $aux_u = \mathcal{H}_1\ (\mathcal{C}ert_u, \text{ID}_u)\ .\ \chi_u + \Upsilon$

$= \mathcal{A}_u$, where $\mathcal{A}_u = \mathcal{H}_1\ (\mathcal{C}ert_u, \text{ID}_u).\omega_u + aux_u$

$= \mathcal{A}_u\ .d = \mathcal{Q}_u$ , while it same process at the receiver side for making the public key of the sender, by using the following computations.

$\mathcal{Q}_s \overset{?}{=} \mathcal{H}_2\ (\mathcal{C}ert_s, \text{ID}_s)\ .\ \mathcal{C}ert_s + \text{T}$

$= \mathcal{H}_2\ (\mathcal{C}ert_s, \text{ID}_s)\ .\ (\phi_s + \mathcal{PV}_s\ ) + \Upsilon.\ d$, where $\mathcal{C}ert_s = \phi_s + \mathcal{PV}_s$ and $\text{T} = \Upsilon.d$

$= \mathcal{H}_2\ (\mathcal{C}ert_s, \text{ID}_s)\ .\ (\phi_s + \omega_s\ .\ d) + \Upsilon$, where $\mathcal{PV}_s = \omega_s\ .\ d$

$= \mathcal{H}_2\ (\mathcal{C}ert_s, \text{ID}_s)\ .\ (\chi_s\ .\ d + \omega_s.d) + \Upsilon.\ d$, where $\phi_s = \chi_s\ .\ d$

$= \chi_s\ .\ d\ .\ \mathcal{H}_2\ (\mathcal{C}ert_s, \text{ID}_s) + \omega_s\ .\ d\ .\ \mathcal{H}_2\ (\mathcal{C}ert_s, \text{ID}_s) + \Upsilon.\ d$

$= (\chi_s\ .\ \mathcal{H}_2\ (\mathcal{C}ert_s, \text{ID}_s) + \omega_s\ .\ \mathcal{H}_2\ (\mathcal{C}ert_s, \text{ID}_s) + \Upsilon)\ .\ d$

$= (aux_s + \omega_s\ .\ \mathcal{H}_2\ (\mathcal{C}ert_s, \text{ID}_s))\ .\ d$ where $aux_s = \mathcal{H}_1\ (\mathcal{C}ert_s, \text{ID}_s)\ .\ \chi_s + \Upsilon$

$= \mathcal{A}_s\ .\ d$, where $\mathcal{A}_s = \mathcal{H}_1\ (\mathcal{C}ert_s, \text{ID}_s)\ .\ \omega_s + aux_s$

$= \mathcal{A}_s\ .\ d = \mathcal{Q}_s$

And the receiver also recovers the secret key by using the following steps:

$\mathcal{SK} = \mathcal{A}_u\ .\ \beta$

$= \mathcal{A}_u\ .\ (\mathcal{S}.d + \Lambda\ .\ \mathcal{Q}_s)$, where $\beta = \mathcal{S}\ .\ d + \Lambda\ .\ \mathcal{Q}_s$

$= \mathcal{A}_u\ .\ (\mathcal{S}\ .\ d + \Lambda\ .\ \mathcal{A}_s\ .\ d)$, where $\mathcal{Q}_s = \mathcal{A}_s\ .\ d$

$= \mathcal{A}_u\ .\ (\Omega - \Lambda\ .\ \mathcal{A}_s\ .\ d + \Lambda\ .\ (\mathcal{A}_s\ .\ d))$, where $\mathcal{S} = \Omega - \Lambda\ .\ \mathcal{A}_s$

$= \mathcal{A}_u\ .\ d\ (\Omega - \Lambda\ .\ \mathcal{A}_s + \Lambda\ .\ \mathcal{A}_s) = \mathcal{A}_u\ .\ d = \mathcal{Q}_{u\ .}\ (\Omega) = \mathcal{SK}$, where $\mathcal{A}_u\ .\ d = \mathcal{Q}_u$

## 5. Informal Security Analysis

The proposed CB-SN scheme ensures the following informal security requirements.

### 5.1. Confidentiality

Confidentiality means that the contents of a plain text (m) should hide from intruders and the intruders cannot get any meaning from the signcrypted text without knowing the shared secret key. In the proposed CB-SN scheme, if the intruders desire to scramble the contents of a plain text from a signcryption text $\psi$ = {C,A, S}, then it is mandatory for them to reveal the shared secret key SK by computing Equation (2). To compute this equation, it is important for the intruders to extract Ω from Equation (3), which is hard for them because this leads to computing $h\varepsilon c - dl$. Thus, our CB-SN scheme ensures cipher text confidentiality.

$$\mathcal{SK} = \Omega\ .\ \mathcal{Q}_{\mathrm{u}} \tag{3}$$

$$\beta = \Omega\ .\ d \tag{4}$$

### 5.2. Integrity

Integrity means that the contents of a plain text (m) can only be modified by the intended participant or user. In our CB-SN scheme, the sender computes a hash value of a message (m) like $\Lambda = \mathrm{H}_2(\mathrm{m}\|\ \mathrm{N})$ and delivers the value $\Lambda$ with a cipher text to the receiver. Therefore, if an event occurred, i.e., an intruder tries to modify in cipher text C like $\mathrm{C}^*$, then, the intruder must modify m into $\mathrm{m}^*$ and $\Lambda = \mathrm{H}_2(\mathrm{m}\|\ \mathrm{N})$ into $\Lambda^* = \mathrm{H}_2(\mathrm{m}\|\ \mathrm{N})^*$, which is infeasible because of the collision resistance property of a hash function.

### 5.3. Unforgeability

Without the private key of a sender, the illegal user cannot produce the original signature which is called unforgeability. In our proposed CB-SN scheme, the sender computes a digital signature, i.e., $\mathcal{S} = \Omega - \Lambda\ .\ \mathcal{A}_{\mathrm{s}}$. This includes the sender private number $\Omega$ and private key $\mathcal{A}_{\mathrm{s}}$, which is only known to the sender. If the intruder tries to create the same signature, he cannot do it, because discovering two unknown variables from the same equation is infeasible. Therefore, our CB-SN scheme meets the unforgeability security service.

### 5.4. Public Verifiability

Public verifaibility is the property of signcryption , in which the third party removes the clash among the sender and receiver, which is already causing. In our case, the sender computes the digital signature for a plain text by using his private key and it is common practice in asymmetric cryptosystem that the public key of a user is related to his private key. Therefore, for eliminating a conflict, the third party/judge can use either $\mathcal{Q}_{\mathrm{u}} \overset{?}{=} \mathcal{H}_2\ (\mathcal{C}ert_{\mathrm{u}},\ \mathrm{ID}_{\mathrm{u}})\ .\ \mathcal{C}ert_{\mathrm{u}} + \mathrm{T}$ or $\mathcal{Q}_{\mathrm{u}} \overset{?}{=} \mathcal{H}_2\ (\mathcal{C}ert_{\mathrm{u}},\ \mathrm{ID}_{\mathrm{u}})\ .\ \mathcal{C}ert_{\mathrm{u}} + \mathrm{T}$, in which the equality of the previous two equations is available in the Section 4.1 of this paper.

### 5.5. Forward Secrecy

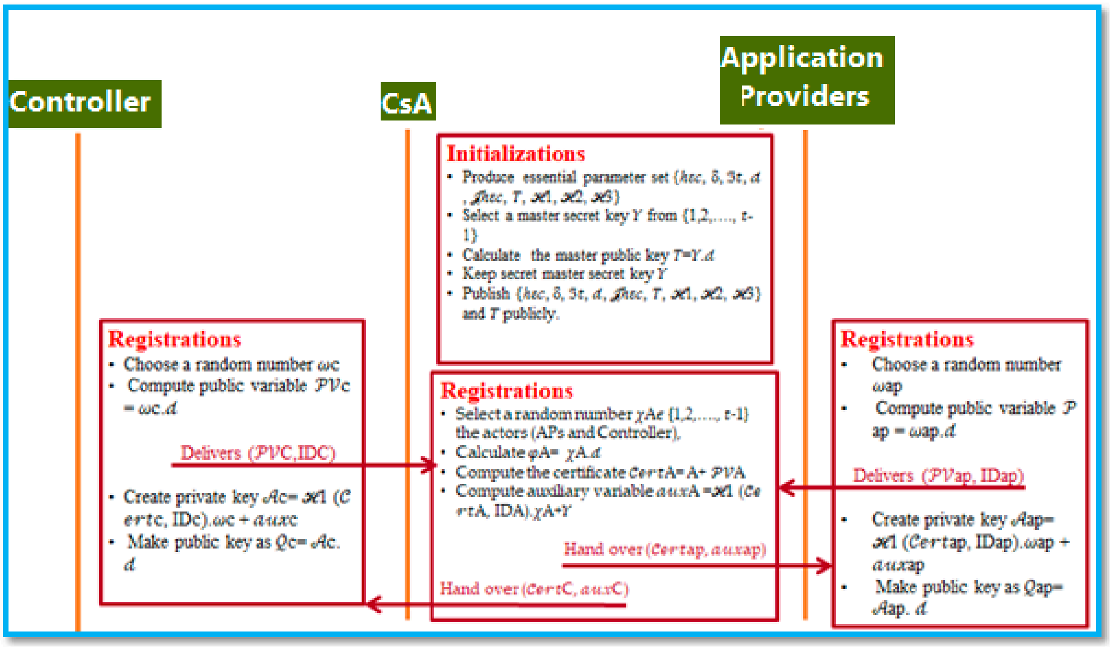
When the private key of a legitimate sender is compromised by an intruder, then, the existing communicated messages that is still safe is called forward secrecy. In our case, the sender encrypts the message ($m$) like $\mathrm{C} = \mathcal{E}_{\mathcal{SK}}\ (m\|\mathrm{N})$, by utilizing the secret shared key SK. If the intruder compromised the private key of the sender, furthermore, he needs the secret shared SK from Equation (2), for the decryption of cipher text. Hence, computing the secret key SK from Equation (2) is infeasible for the intruder, which is already discussed in the confidentiality section.

### 5.6. Anti-Replay Attack

When an intruder has captured the already communicated signcrypted text and continuously transmits this text to the receiver it is called a replay attack. Therefore, the replay attack is not possible in our case, because the sender sends a fresh nonce (N) in the encrypted text (C), and furthermore, the encryption process ($\mathrm{C} = \mathcal{E}_{\mathcal{SK}}\ (m\|\mathrm{N})$) is processed through a shared secret key SK, in which the nonce (N) and secret key SK are renewed for every session of communication. Therefore, the anti-replay attack is provided in this paper.

## 6. CB-SN Access Control for WBAN

Figures 2 and 3 illustrate the overall process of implementation, which contains the following four steps:



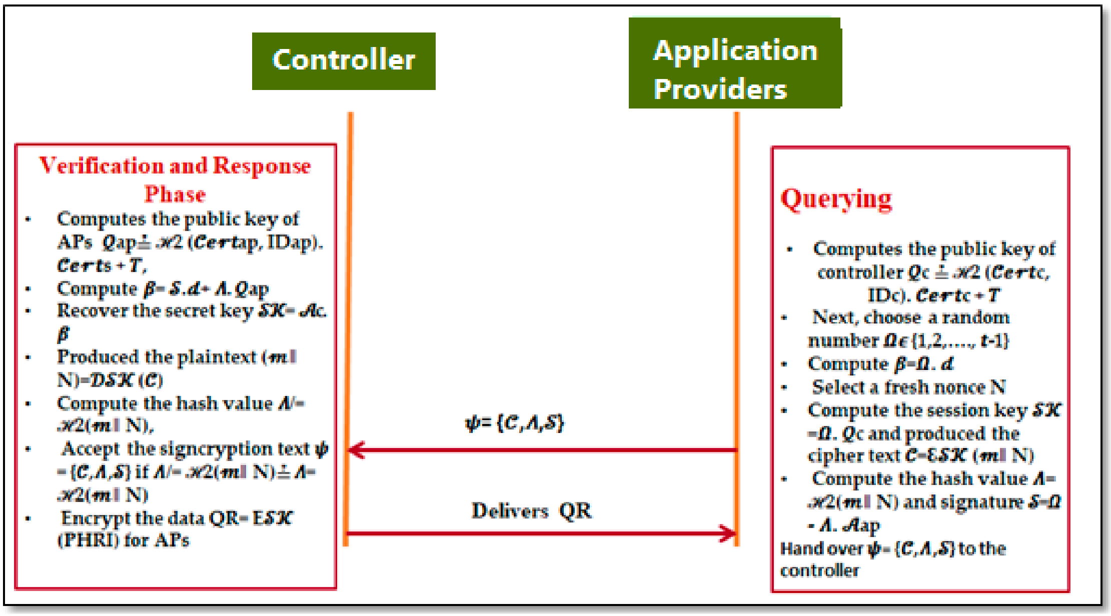
Initializations
- Produce essential parameter set {hεc, δ, 𝔍t, 𝒹, 𝒥hεc, T, ℋ1, ℋ2, ℋ3}
- Select a master secret key Υ from {1,2,...., t-1}
- Calculate the master public key T=Υ.𝒹
- Keep secret master secret key Υ
- Publish {hεc, δ, 𝔍t, 𝒹, 𝒥hεc, T, ℋ1, ℋ2, ℋ3} and T publicly.

Registrations
- Choose a random number ωc
- Compute public variable 𝒫𝒱c = ωc.𝒹
- Create private key 𝒜c= ℋ1 (𝒞ℯ𝓇tc, IDc).ωc + 𝒶𝓊𝓍c
- Make public key as 𝒬c= 𝒜c. 𝒹

Registrations
- Select a random number χAϵ {1,2,...., t-1} the actors (APs and Controller).
- Calculate φA= χA.𝒹
- Compute the certificate 𝒞ℯ𝓇tA= A+ 𝒫𝒱A
- Compute auxiliary variable 𝒶𝓊𝓍A =ℋ1 (𝒞ℯ𝓇tA, IDA).χA+Υ

Registrations
- Choose a random number ωap
- Compute public variable 𝒫ap = ωap.𝒹
- Create private key 𝒜ap= ℋ1 (𝒞ℯ𝓇tap, IDap).ωap + 𝒶𝓊𝓍ap
- Make public key as 𝒬ap= 𝒜ap. 𝒹

**Figure 2.** Initialization and registration phase of certificate-based signcryption (CB-SN) access control for wireless body area networks (WBAN).



Verification and Response Phase
- Computes the public key of APs 𝒬ap≟ ℋ2 (𝒞ℯ𝓇tap, IDap). 𝒞ℯ𝓇ts + T,
- Compute β= S.𝒹+ Λ.𝒬ap
- Recover the secret key 𝒮𝒦= 𝒜c. β
- Produced the plaintext (𝓂‖ N)=D𝒮𝒦 (C)
- Compute the hash value Λ/= ℋ2(𝓂‖ N),
- Accept the signcryption text ψ = {C,Λ,S} if Λ/= ℋ2(𝓂‖ N)≟ Λ= ℋ2(𝓂‖ N)
- Encrypt the data QR= E𝒮𝒦 (PHRI) for APs

Querying
- Computes the public key of controller 𝒬c ≟ ℋ2 (𝒞ℯ𝓇tc, IDc). 𝒞ℯ𝓇tc + T
- Next, choose a random number Ωϵ {1,2,...., t-1}
- Compute β=Ω. 𝒹
- Select a fresh nonce N
- Compute the session key 𝒮𝒦 =Ω. 𝒬c and produced the cipher text C=E𝒮𝒦 (𝓂‖ N)
- Compute the hash value Λ= ℋ2(𝓂‖ N) and signature S=Ω - Λ. 𝒜ap

Hand over ψ= {C,Λ,S} to the controller

**Figure 3.** Querying and response phase of CB-SN access control for WBAN.

### 6.1. Initialization

The CsA calls the setup process where CAs produce an essential parameter set $\{h\varepsilon c, \delta, \mathfrak{J}_t, d, \mathcal{J}_{h\varepsilon c}$, T, $\mathcal{H}_1, \mathcal{H}_2, \mathcal{H}_3\}$ and after that select a master secret key Υ from {1,2, … , $t - 1$} and calculate the master public key T = Υ . $d$, respectively. The essential parameters set is directly accessible on a network and the master secret key Y is kept by the $C_s$A secret.

### 6.2. Registration

In addition, each actor (APs and controller) with identity $ID_A$, chooses a random number $\omega_A$ and computes their public variable $\mathcal{PV}_A = \omega_A.d$. Then, the actor (APs and controller) with identity $ID_A$ delivers ($PV_A$, $ID_A$) to the $C_s$A by using open channel. Furthermore, the $C_s$A selects a random number $\chi_A \varepsilon$ {1,2, … , $t-1$}, calculates $\phi_A = \chi_A \cdot d$, computes the certificate $\mathrm{Cert}_A = \phi_A + \mathrm{PV}_A$, computes auxiliary variable $aux_A = \mathscr{H}_1 (\mathscr{C}ert_A, \mathrm{ID}_A) \cdot \chi_A + \Upsilon$ and, then, hand over a certificate $\mathrm{Cert}_A$ with auxiliary variable ($\mathrm{Cert}_A$, $\mathrm{aux}_A$) to the actors (APs and controller) via insecure link. Moreover, each actor (APs and controller) creates his private key $\mathrm{A}_A = \mathrm{H}_1 (\mathrm{Cert}_A, \mathrm{ID}_A) \cdot \mathrm{w}_A + \mathrm{aux}_A$ and make his public key as $\mathrm{Q}_A = \mathrm{A}_A \cdot \mathrm{d}$.

### 6.3. Querying Phase

The APs call the SG algorithm, that is, the APs compute the public key of controller $\mathrm{Q}_c \overset{?}{=} \mathrm{H}_2 (\mathrm{Cert}_c, \mathrm{ID}_c) \cdot \mathrm{Cert}_c + \mathrm{T}$. Next, they choose a random number $\Omega \varepsilon$ {1, 2, … , $t-1$} and compute $\beta = \Omega \cdot d$, select a fresh nonce N, compute the session key $\mathrm{SK} = \Omega \cdot \mathrm{Q}_c$ and produced the cipher text $\mathrm{C} = \mathcal{E}_{\mathscr{S}\mathscr{K}} (m \parallel \mathrm{N})$, compute the hash value $\Lambda = \mathrm{H}_2(m \parallel \mathrm{N})$ and signature $\mathscr{S} = \Omega - \Lambda \cdot \mathrm{A}_{ap}$, then, hand over $\psi$ = {C,S} to the controller using insecure channel.

### 6.4. Verification and Response

For this purpose, the controller first calls a US algorithm, that is, the controller, first, computes the public key of APs $\mathscr{Q}_{ap} \overset{?}{=} \mathscr{H}_2 (\mathscr{C}ert_{ap}, \mathrm{ID}_{ap}) \cdot \mathscr{C}ert_s + \mathrm{T}$, computes $\beta = \mathscr{S} \cdot d + \Lambda \cdot \mathscr{Q}_{ap}$, recovers the secret key $\mathscr{S}\mathscr{K} = \mathscr{A}_c \cdot \beta$ and produces the plaintext $(m \parallel \mathrm{N}) = \mathscr{D}_{\mathscr{S}\mathscr{K}} (\mathscr{C})$, computes the hash value $\Lambda^{/} = \mathscr{H}_2(m \parallel \mathrm{N})$, accepts the signcryption text $\psi = \{\mathscr{C}, \Lambda, \mathscr{S}\}$, if $\Lambda^{/} = \mathscr{H}_2(m \parallel \mathrm{N}) \overset{?}{=} \Lambda = \mathscr{H}_2(m \parallel \mathrm{N})$ and encrypts the data $\mathrm{QR} = \mathrm{E}_{\mathscr{S}\mathscr{K}}$ (PHRI) for Aps, and delivers it using the open networks.

## 7. Performance

We choose three main parameters that are security services, energy (computational cost), and bandwidth (communication cost), in the proposed CB-SN access control scheme and existing ones, i.e., Li et al. [22], Omala et al. [15], Gao et al. [24], Braeken et al. [17], Braeken et al. [19] Schemes 1, 2, and 3, for measuring the performance. Appendix A shows the mplementations of CB-SN Access Control Scheme in AVISPA.

### 7.1. Security Performance

The security performance of a designed and existing access control scheme as shown in Table 2, in which we pick the security services and verification tool, i.e., confidentiality, unforgeability, authentication, integrity, anti-replay attack, forward secrecy, public verifiability, random oracle model, and formal verification through AVISPA, respectively. The symbols CFY, UFY, ATN, ITY, ARA, FSY, PVY, ROM, FVTA, √, and ⊗ indicate confidentiality, unforgeability, authentication, integrity, anti-replay attack, forward secrecy, public verifiability, random oracle model, formal verification through AVISPA, satisfying the service, and does not satisfy, respectively. Therefore, it is clearly shown that, our proposed CB-SN meet all the claimed security services and the schemes, i.e., Li et al. [22], Omala et al. [15], Gao et al. [24], Braeken et al. [17], Braeken et al. [19] Schemes 1, 2, and 3 do not meet the services such as FSY and PVY, as well as FVTA or some other verification tool.

**Table 2.** Comparison with respect to security properties.

| Security Services | [22] | [15] | [24] | [17] | [19] S1 | [19] S2 | [19] S3 | Proposed |
|---|---|---|---|---|---|---|---|---|
| CFY | √ | √ | √ | √ | √ | √ | √ | √ |
| UFY | √ | √ | √ | √ | √ | √ | √ | √ |
| ATN | √ | √ | √ | √ | √ | √ | √ | √ |
| ITY | √ | √ | √ | √ | √ | √ | √ | √ |
| ARA | √ | √ | √ | ⊗ | ⊗ | ⊗ | ⊗ | √ |
| FSY | ⊗ | ⊗ | ⊗ | √ | √ | √ | √ | √ |
| PVY | ⊗ | ⊗ | ⊗ | ⊗ | ⊗ | ⊗ | ⊗ | √ |
| ROM | √ | √ | √ | ⊗ | ⊗ | ⊗ | ⊗ | ⊗ |
| FVTA | ⊗ | ⊗ | ⊗ | ⊗ | ⊗ | ⊗ | ⊗ | √ |

### 7.2. Computational Cost

The comparison among the proposed CB-SN access control scheme and existing ones, i.e., Li et al. [22], Omala et al. [15], Gao et al. [24], Braeken et al. [17], Braeken et al. [19] Schemes 1, 2, and 3, on the basis of major operations is provided in this section. Normally, the computational cost includes an expensive mathematical operation, for example, bilinear pairing ($bp$), modular exponential ($mxp$), elliptic curve scalar multiplication ($esm$), and hyper-elliptic curve divisor scalar multiplication ($hsm$), while designing a cryptographic algorithm. Next, in Table 3, we provide a required major operation of the proposed CB-SN access control scheme and existing ones, i.e., Li et al. [22], Omala et al. [15], Gao et al. [24], Braeken et al. [17], Braeken et al. [19] Schemes 1, 2, and 3. The calculated values of Table 3, regarding $bp$, $mxp$, and $esm$ are based on [27], and $sm$ is based on the assumption of [29]. Therefore, according to [29], the single $bp$ required 14.90 ms, single $mxp$ consumes 1.25 ms, and single $esm$ needs 0.97 ms, while the assumption of [29], about single $hsm$ is that, it required 0.48 ms. The experiment was performed by using the hardware resources, i.e., Intel Core i74510UCPU, 8 GB RAM, and 2.0GHz processor and software resources such as C++ with Multi-precision Integer and Rational Arithmetic C Library (MIRACL) and window 7. By using the data of Table 3, our scheme reduced the computational cost on the basis of ms from Li et al. [22] is ($2bp$ + $6esm$ + $2mxp$) - ($7hsm$)/ ($2bp$ + $6esm$ + $2mxp$) = 36.94 – 3.36/36.94*100 = 90.90%, Omala et al. [15] is ($6esm$) - ($7\ hsm$)/($6\ esm$) = 5.82 – 3.36/5.82*100 = 42.26%, Gao et al. [23] is ($7esm$) - ($7hsm$)/($7esm$) = 6.77 – 3.36/6.77*100 = 50.36%, Braeken et al. [17] is ($9esm$) - ($7hsm$)/($9esm$) = 8.73 – 3.36/8.73*100 = 61.51%, Braeken et al. [19] scheme 1(S1) is ($7esm$) - ($7hsm$)/($7esm$) = 6.77 – 3.36/6.77*100 = 50.36%, Braeken et al. [19] scheme 2(S2) is ($7esm$) - ($7hsm$)/($7esm$) = 6.77 – 3.36/6.77*100 =50.36%, Braeken et al. [19] scheme 3(S3) is ($8esm$) - ($7hsm$)/($8esm$) = 7.76 – 3.36/7.76*100 = 56.70%, respectively. Moreover, in Table 4, we deliver the computational cost on the basis of milliseconds (ms) among the proposed CB-SN access control and those of Li et al. [23], Omala et al. [15], and Gao et al. [24]. Furthermore, in Figure 4, the clear computational cost reduction is shown.

**Table 3.** Comparison with respect to major operations.

| Schemes | Signcryption Generation (SG) (APs) | Unsigncryption (US) (Controller) | Total |
|---|---|---|---|
| Li et al. [22] | 1 $mxp$ + 4 $esm$ | 2 $bp$ + 2 $esm$ +1 $mxp$ | 2 $bp$ + 6 $esm$ +2 $mxp$ |
| Omala et al. [15] | 3 $esm$ | 3 $esm$ | 6 $esm$ |
| Gao et al. [23] | 3 $esm$ | 4 $esm$ | 7 $esm$ |
| Braeken et al. [17] | 4 $esm$ | 5 $esm$ | 9 $esm$ |
| Braeken et al. [19] Scheme 1 | 5 $esm$ | 2 $esm$ | 7 $esm$ |
| Braeken et al. [19] Scheme 2 | 3 $esm$ | 4 $esm$ | 7 $esm$ |
| Braeken et al. [19] Scheme 3 | 4 $esm$ | 4 $esm$ | 8 $esm$ |
| Proposed CB-SN | 3 $hsm$ | 4 $hsm$ | 7 $hsm$ |

**Table 4.** Computational cost comparison in milliseconds.

| Schemes | Signcryption Generation (SG)(APs) | Unsigncryption (US) (Controller) | Total |
|---|---|---|---|
| Li et al. [22] | 5.13 ms | 31.81 ms | 36.94 ms |
| Omala et al. [15] | 2.91 ms | 2.91 ms | 5.82 ms |
| Gao et al. [23] | 2.91 ms | 3.88 ms | 6.77 ms |
| Braeken et al. [17] | 3.88 ms | 4.85 ms | 8.73 ms |
| Braeken et al. [19] Scheme 1 | 4.85 ms | 1.92 ms | 6.77 ms |
| Braeken et al. [19] Scheme 2 | 2.91 ms | 3.88 ms | 6.77 ms |
| Braeken et al. [19] Scheme 3 | 3.88 ms | 3.88 ms | 7.76 ms |
| Proposed CB-SN | 1.44 ms | 1.92 ms | 3.36 ms |

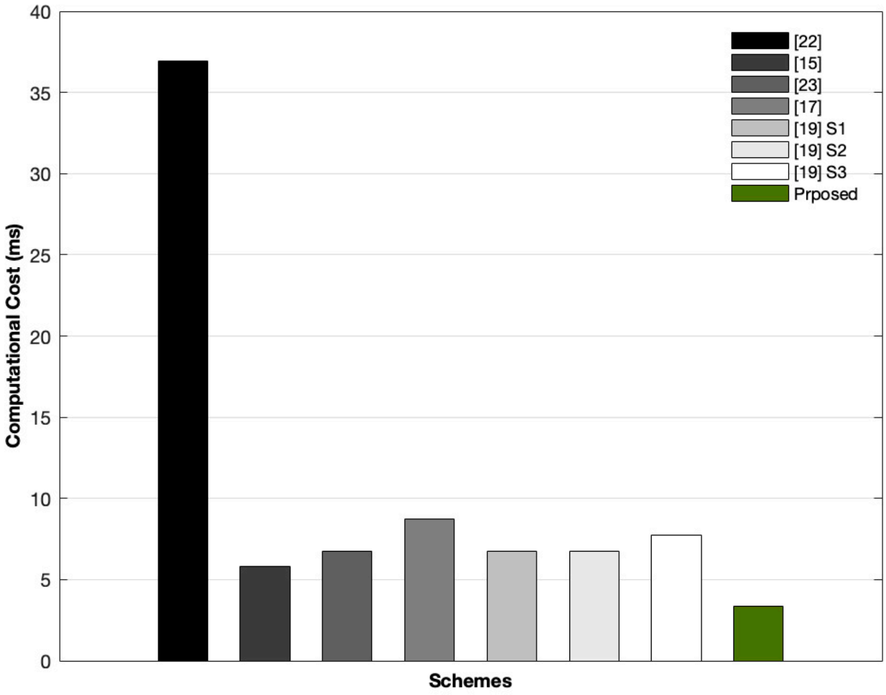


**Figure 4.** Total computational cost reduction.

### *7.3. Communication Cost*

We compare our newly proposed CB-SN access control scheme with the existing related access control schemes, i.e., Li et al. [22], Omala et al. [15], Gao et al. [24], Braeken et al. [17], Braeken et al. [19] Schemes 1, 2, and 3 on the basis of communication cost. Usually, the communication cost of the signcryption schemes is calculated by using the cipher text and the extra parameters such as signature, hash value, and identity, etc., during the communication process. For the comparison, we suppose that, $|\mathfrak{G}_1| = |\mathfrak{G}| = |\mathfrak{G}_2| = 1024$ bits, $|\mathcal{Z}_q| = 160$ bits, $|\mathcal{Z}_\mathrm{n}| = 80$ bits, and $|\mathrm{H}| = 512$ bits, $|m| = 1024$ bits, and $|\mathrm{ID}| = 80$ bits. According to our suppositions, the communication cost for Li et al. [22] is $3|\mathfrak{G}_1| + |\mathrm{ID}| + |m|$, for Omala et al. [15] is $|\mathfrak{G}_1| + |\mathrm{ID}| + |m| + |\mathcal{Z}_q|$, for Gao et al. [2] is $|\mathrm{ID}| + |m| + 5|\mathcal{Z}_q|$, for Braeken et al. [17] $|\mathrm{ID}| + |m| + 5|\mathcal{Z}_q|$, for Braeken et al. [19] Scheme 1 is $|\mathrm{ID}| + |m| + 5|\mathcal{Z}_q|$, for Braeken et al. [19] Scheme 2 is $|\mathrm{ID}| + |m| + 5|\mathcal{Z}_q|$, Braeken et al. [19] Scheme 3 is $|\mathrm{ID}| + |m| + 5|\mathcal{Z}_q|$, and for our designed CB-SN access control scheme is $|\mathcal{Z}_\mathrm{n}| + |\mathrm{H}| + |m|$, respectively.

1. The reduction in communication cost of the proposed CB-SN access control scheme from Li et al. [22] is $(3|\mathfrak{G}_1| + |\mathrm{ID}| + |m|)\ (|\mathcal{Z}_\mathrm{n}| + |\mathrm{H}| + |m|)/(3|\mathfrak{G}_1| + |\mathrm{ID}| + |m|) = (4176 - 1616)/(4176)*100 = 61.30\%$,
2. The reduction in communication cost of our designed CB-SN access control scheme from Omala et al. [15] is $(|\mathfrak{G}_1| + |\mathrm{ID}| + |m| + |\mathcal{Z}_q|)(|\mathcal{Z}_\mathrm{n}| + |\mathrm{H}|+|m|)/(|\mathfrak{G}_1|+ |\mathrm{ID}| + |m| + |\mathcal{Z}_q|) = (2288 - 1616)/(2288)*100 = 29.37\%$.
3. The reduction in communication cost of of the proposed CB-SN access control scheme from Gao et al. [23] is $(|\mathrm{ID}| + |m| + 5|\mathcal{Z}_q|) - (|\mathcal{Z}_\mathrm{n}| + |\mathrm{H}| + |m|)/(|\mathrm{ID}| + |m| + 5|\mathcal{Z}_q|) = (1904 - 1616)/(1904)*100 = 15.12\%$.
4. The reduction in communication cost of the proposed CB-SN access control scheme from Braeken et al. [17] and Braeken et al. [19] Scheme 1, 2, and 3 is $(|m| + |\mathrm{H}| + |\mathcal{Z}_q|) - (|\mathcal{Z}_\mathrm{n}| + |\mathrm{H}| + |m|)/|m| + 2|\mathcal{Z}_q|) = (1696 - 1616)/(1696)*100 = 4.71\%$.

Comparative analysis of communication cost of the proposed CB-SN scheme with relevant existing schemes are provided in Figure 5, which shows a clear savings of communication cost.

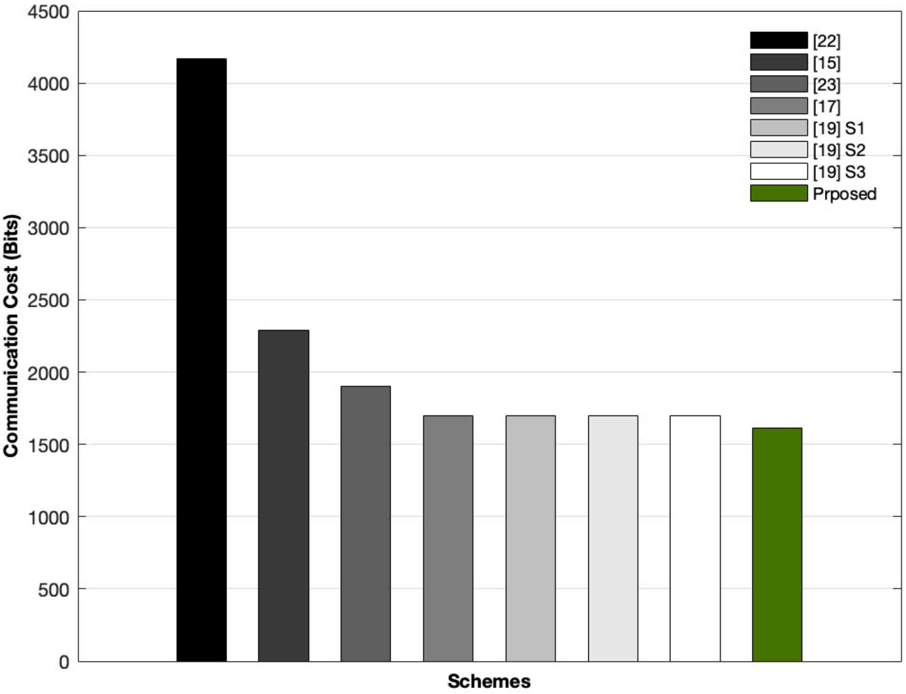


**Figure 5.** Total communication cost.

## 8. Conclusions

Connected health brings together multidisciplinary technologies, such as the Internet of things (IoT) and wireless body area networks (WBAN), to provide preventive or proactive healthcare services by connecting devices and persons to build up the modern healthcare system. However, due to the openness of the wireless environment, the privacy of people's physiological data, and resource-constrained nature of IoT devices, especially in terms of energy supply, often body sensors are vulnerable to different kinds of known and unknown cryptographic attacks. To tackle these issues, a comprehensive review of the existing certificate-based signcryption schemes was carried out in the literature. We found that these schemes are based on hard problems, i.e., elliptic curve and bilinear pairing, suffering from high computational cost and communication overhead. Therefore, to solve this problem, a new design scheme, called certificate-based signcryption access control scheme, is introduced, because the new scheme involves a hyper-elliptic curve, which offers the same level of security as the elliptic curve and bilinear pairing with lower key size. Furthermore, in the proposed scheme addresses and resolves intelligently the issues of increased computational cost and undesired communication overhead.

**Author Contributions:** Conceptualization, I.U. and A.A.; Methodology and Implementation, I.U., A.A. and M.A.K.; Simulation, I.U., A.A., and M.A.K.; Validation, I.U., A.A., H.K. and M.A.K.; Data Curation, A.A., and H.K.; Writing-Original Draft Preparation, A.A., N.U.A. and I.U.; Writing-Review & Editing, A. A. and M.A.K.; Supervision, N.U.A.

**Funding:** This research received no external funding.

**Conflicts of Interest:** The authors declare no conflict of interest.

## Appendix A. Implementations of CB-SN Access Control Scheme in AVISPA

In this subphase, we implement our proposed CB-SN access control for WBAN in AVISPA tool. For this experiment, we used hardware resources, for example, Haier Win8.1 PC, Intel (R) Core (TM) i3-4010U CPU @ 1.70 GHz, supporting 64-bit operating system, and x64-based processor. In addition, the software resources such as Oracle VM virtual Box (version: 5.2.0.118431) and SPAN (version: SPAN-Ubuntu-10.10-light_1). Our implementation contains four roles that are APs which are provided in Algorithm A1, a controller which is shown in Algorithm A2, and environment and session roles which are provided in Algorithm A3, respectively. Here, we provide clarification of some symbols, which are used in HLPSL language in these four roles, as well as in the CB-SN access control for WBAN. We use the arrow sign ↔ to represent the similarity, whereas the symbol that occurs before this sign represents HLPSL, after using for the algorithm. Therefore, Qap ↔ $\mathcal{Q}_{ap}$, Qc ↔ $\mathcal{Q}_{c}$, N ↔ N, Omega ↔ $\Omega$, A ↔ $\Lambda$, {E(M')}_Sk' ↔ $\mathcal{C} = \mathcal{E}_{\mathcal{SK}}(m \parallel N)$, Sk' ↔ $\mathcal{SK}$, M' ↔ $m$, {Minuss(Omega'.A')}_inv(Qap) ↔ $\mathcal{S} = \Omega - \Lambda \,.\, \mathcal{A}_{ap}$, and inv (Qap) ↔ $\mathcal{A}_{ap}$, respectively. Therefore, we test (1000 times), and our scheme gives SAFE results under the on-the-fly model checker (O-f-M-C) and constraint logic-based attack searcher (CL-ATSe), which are provided in Figures A1 and A2.

**Algorithm A1** High level protocol specification language (HLPSL) code for application providers (Aps) role

```
  role
  role_Aps(Aps:agent,Controller:agent,Qap:public_key,Qc:public_key,SND,RCV:channel(dy))
  played_by Aps
  def =
    local
        State:nat, N:text, Minuss:hash_func, Omega:text, A:text, M:text, E:hash_func,
    Sk:symmetric_key
    init
        State := 0
    transition
        1. State=0 /\ RCV(start) =|> State':=1 /\ SND(Aps.Controller)
        2. State=1 /\ RCV(Controller.{N'}_Qc) =|> State':=2 /\ A':=new() /\ Omega':=new() /\ Sk':=new() /\
M':=new() /\ secret(M',sec_2,{Aps}) /\ witness(Aps,Controller,auth_1,M') /\
SND(Aps.{E(M')}_Sk'.{Minuss(Omega'.A')}_inv(Qap))
end role
```

```
  role
  role_Aps(Aps:agent,Controller:agent,Qap:public_key,Qc:public_key,SND,RCV:channel(dy))
  played_by Aps
  def =
    local
        State:nat, N:text, Minuss:hash_func, Omega:text, A:text, M:text, E:
    hash_func, Sk:symmetric_key
    init
        State:= 0
    transition
        1. State = 0 /\ RCV(start) =|> State':=1 /\ SND(Aps.Controller)
        2. State = 1 /\ RCV(Controller.{N'}_Qc) =|> State':=2 /\ A':=new() /\ Omega':=new() /\ Sk':=new() /\
M':=new() /\ secret(M',sec_2,{Aps}) /\ witness(Aps,Controller,auth_1,M') /\
SND(Aps.{E(M')}_Sk'.{Minuss(Omega'.A')}_inv(Qap))
end role
```

**Algorithm A2** HLPSL code for controller role.

```
  role
  role_Controller(Aps:agent,Controller:agent,Qap:public_key,Qc:public_key,SND,RCV:channel(dy))
  played_by Controller
  def =
    local
        State: nat, N:text, Minuss:hash_func, Omega:text, A:text, M:text, E:
    hash_func, Sk:symmetric_key
    init
        State := 0
    transition
        1. State=0 /\ RCV(Aps.Controller) =|> State':=1 /\ N':=new() /\ SND(Controller.{N'}_Qc)
        6. State=1 /\ RCV(Aps.{E(M')}_Sk'.{Minuss(Omega'.A')}_inv(Qap)) =|> State':=2 /\
request(Controller,Aps,auth_1,M') /\ secret(M',sec_2,{Aps})
end role
```

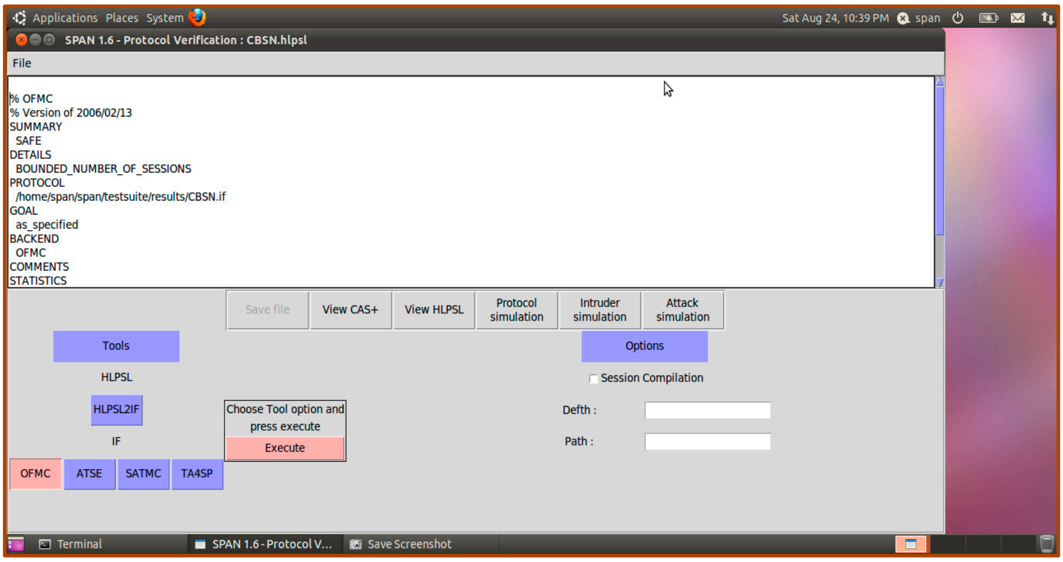
**Algorithm A3** HLPSL code for session and environmental role.

```
  role session1 (Aps:agent,Controller:agent, Qap:public_key, Qc:public_key)
  def =
    local
        SND2,RCV2,SND1,RCV1:channel(dy)
    composition
        role_Controller(Aps,Controller,Qap,Qc,SND2,RCV2)/\
    role_Aps(Aps,Controller,Qap,Qc,SND1,RCV1)
  end role

  role session2(Aps:agent,Controller:agent,Qap:public_key,Qc:public_key)
  def =
    local
        SND1,RCV1:channel(dy)
    composition
        role_Aps(Aps,Controller,Qap,Qc,SND1,RCV1)
  end role

  role environment()
  def =
    const
        hash_0:hash_func,qap:public_key,alice:agent,bob:agent,
        qc:public_key,const_1:agent,const_2:public_key,
        const_3:public_key,auth_1:protocol_id,sec_2:protocol_id
        intruder_knowledge = {alice,bob}
    composition
        session2(i,const_1,const_2,const_3) /\ session1(alice,bob,qap,qc)
  end role
  goal
    authentication_on auth_1
    secrecy_of sec_2
  end goal

environment ()
```



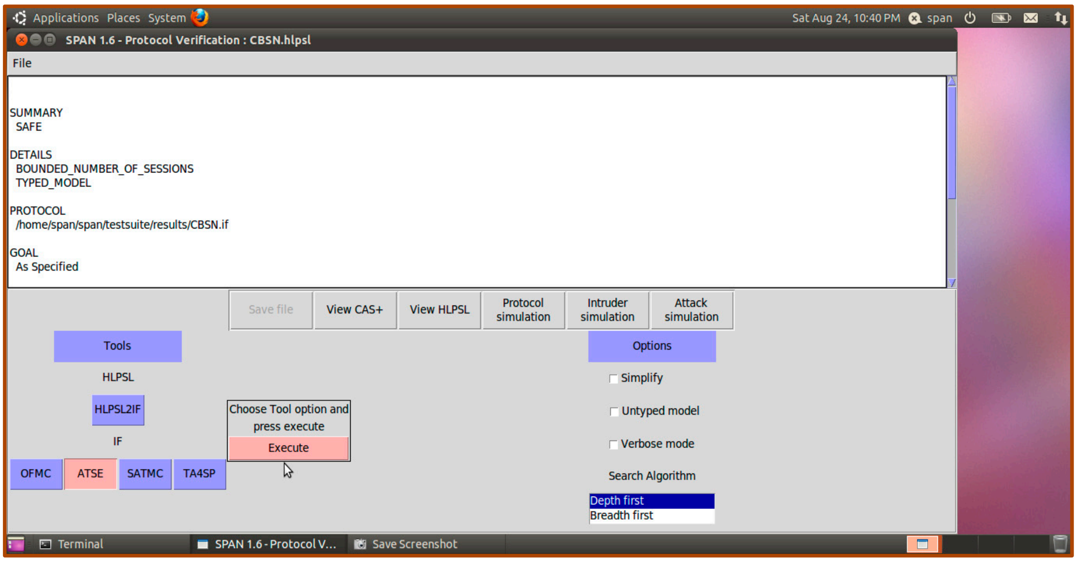
```
% OFMC
% Version of 2006/02/13
SUMMARY
 SAFE
DETAILS
 BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
 /home/span/span/testsuite/results/CBSN.if
GOAL
 as_specified
BACKEND
 OFMC
COMMENTS
STATISTICS
```

□ Session Compilation

Defth :

Path :

**Figure A1.** Simulation result of OFMC.



```
SUMMARY
 SAFE

DETAILS
 BOUNDED_NUMBER_OF_SESSIONS
 TYPED_MODEL

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

GOAL
 As Specified
```

□ Simplify

□ Untyped model

□ Verbose mode

Search Algorithm

Depth first (selected)

Breadth first

**Figure A2.** Simulation result of ATSE.

## References

1. Alkhayyat, A.; Thabit, A.A.; Al-Mayali, F.A.; Abbasi, Q.H. WBSN in IoT Health-Based Application: Toward Delay and Energy Consumption Minimization. *J. Sens.* **2019**, *2019*, 2508452. [CrossRef]
2. El-hajj, M.; Fadlallah, A.; Chamoun, M.; Serhrouchni, A. A Survey of the Internet of Things (IoT) Authentication Schemes. *Sensors* **2019**, *19*, 1141. [CrossRef] [PubMed]
3. Chaudhary, S.; Singh, A.; Kakali, C. Wireless Body Sensor Network (WBSN) Security and Privacy Issues: A Survey. *Int. J. Comp. Int. IoT* **2019**, *2*, 515–521.
4. Zhou, C. An improved lightweight certificateless generalized signcryption scheme for mobile-health system. *Int. J. Dist. Sen. Netw.* **2019**. [CrossRef]
5. Kumar, M.; Verma, H.K.; Sikka, G. A secure lightweight signature-based authentication for Cloud-IoT crowdsensing environments. *Trans. Emerg. Telecommun. Technol.* **2018**. [CrossRef]
6. Rajesh, S.; Paul, V.; Menon, V.G.; Khosravi, M.R. A Secure and Efficient Lightweight Symmetric Encryption Scheme for Transfer of Text Files between Embedded IoT Devices. *Symmetry* **2019**, *11*, 293. [CrossRef]
7. Zheng, Y. Digital signcryption or how to achieve cost (signature & encryption) cost (signature)+ cost (encryption). In *Advances in Cryptology, CRYPTO'97*; Springer: Cham, Switzerland, 1997; pp. 165–179.

8. Waheed, A.; Iqbal, J.; Din, N.; Islam, S.U.; Umar, A.I.; Amin, N.U. Improved Cryptanalysis of Provable Certificateless Generalized Signcryption. *Int. J. Adv. Comput. Sci. Appl.* **2019**, *10*. [CrossRef]
9. Shamir, A. Identity-based cryptosystems and signature schemes. In *Workshop on the Theory and Application of Cryptographic Techniques*; Springer: Berlin, Germany, 1985; pp. 47–53.
10. Malone-Lee, J. Identity Based Signcryption. Cryptology ePrint Archive; Report 2002/098. 2002. Available online: http://eprint.iacr.org/2002/098 (accessed on 30 July 2019).
11. Luo, W.; Ma, W. Secure and Efficient Data Sharing Scheme Based on Certificateless Hybrid Signcryption for Cloud Storage. *Electronics* **2019**, *8*, 590. [CrossRef]
12. Barbosa, M.; Farshim, P. Certificateless signcryption. In Proceedings of the ACM Symposium on Information, Computer and Communications Security, Tokyo, Japan, 18–20 March 2008; pp. 369–372.
13. Lu, Y.; Li, J. Provably Secure Certificate Based Signcryption Scheme without Pairings. *KSII Trans. Internet Inf. Syst.* **2014**, *8*, 2554–2571.
14. Li, F.; Han, Y.; Jin, C. Practical signcryption for secure communication of wireless sensor networks. *Wirel. Pers. Commun.* **2016**, *89*, 1391–1412. [CrossRef]
15. Omala, A.A.; Mbandu, A.S.; Mutiria, K.D.; Jin, C.; Li, F. Provably Secure Heterogeneous Access Control Scheme for Wireless body area networks. *JMS* **2018**, *42*, 108. [CrossRef] [PubMed]
16. Gentry, C. Certificate-Based Encryption and the Certificate Revocation Problem. In Proceedings of the International Conference on Theory Application of Cryptographic Techniques, Warsaw, Poland, 4–8 May 2003; pp. 272–293.
17. Braeken, A.; Shabisha, P.; Touhafi, A.; Steenhaut, K. Pairing free and implicit certificate based signcryption scheme with proxy re-encryption for secure cloud data storage. In Proceedings of the 2017 3rd International Conference of Cloud Computing Technologies and Applications, Rabat, Morocco, 24–26 Octcber 2017.
18. Le, M.-H.; Hwang, S.O. Certificate-Based Signcryption Scheme without Pairing: Directly Verifying Signcrypted Messages Using a Public Key. *ETRI J.* **2016**, *38*, 724–734. [CrossRef]
19. Braeken, A. Pairing Free Certificate Based Signcryption Schemes Using ECQV Implicit Certificates. *KSII Trans. Internet Inf. Syst.* **2019**, *13*, 1546–1565.
20. Cagalaban, G.; Kim, S. Towards a secure patient information access control in ubiquitous healthcare systems using identity-based signcryption. In Proceedings of the 13th International Conference on Advanced Communication Technology (ICACT), Seoul, Korea, 13–16 February 2011; pp. 863–867.
21. Hu, C.; Zhang, N.; Li, H.; Cheng, X.; Liao, X. Body area network security: A fuzzy attribute-based signcryption scheme. *IEEE J. Sel. Areas Commun.* **2013**, *31*, 37–46. [CrossRef]
22. Li, F.; Hong, J. Efficient Certificateless Access Control for Wireless body area networkss. *IEEE Sens. J.* **2016**, *16*, 5389–5396. [CrossRef]
23. Li, F.; Han, Y.; Jin, C. Cost-effective and anonymous access control for wireless body area networkss. *IEEE Syst. J.* **2018**, *12*, 747–758. [CrossRef]
24. Gao, G.M.; Peng, X.G.; Jin, L.Z. Efficient Access Control Scheme with Certificateless Signcryption for Wireless body area networkss. *Int. J. Netw. Secur.* **2019**, *21*, 428–437.
25. Ullah, I.; Amin, N.U.; Naeem, M.; Khattak, S.J.; Ali, H. A Novel Provable Secured Signcryption Scheme PSSS: A Hyper-Elliptic Curve-Based Approach. *Mathematics* **2019**, *7*, 686. [CrossRef]
26. Ullah, S.; Li, X.-Y.; Zhang, L.A. Review of Signcryption Schemes Based on Hyper Elliptic Curve. In Proceedings of the 3rd International Conference on Big Data Computing and Communications (BIGCOM), Chengdu, China, 10–11 August 2017.
27. Ullah, I.; Haqb, U.I.; Amin, N.U.; Umar, I.A.; Khattak, H. Proxy Signcrypion Scheme Based on Hyper Elliptic Curves. *IJC* **2016**, *20*, 157–166.
28. Khan, M.A.; Qureshi, I.M.; Khanzada, F. A Hybrid Communication Scheme for Efficient and Low-Cost Deployment of Future Flying Ad-Hoc Network (FANET). *Drones* **2019**, *3*, 16. [CrossRef]
29. Rahman, A.U.; Ullah, I.; Naeem, M.; Anwar, R.; Amin, N.U.; Khattak, H.; Ullah, S. A Lightweight Multi-Message and Multi-Receiver Heterogeneous Hybrid Signcryption Scheme based on Hyper Elliptic Curve. *Int. J. Adv. Comput. Sci. Appl. (IJACSA)* **2018**, *9*, 160–167. [CrossRef]