symmetry

MDPI

Article

A Lightweight and Provable Secured Certificateless Signcryption Approach for Crowdsourced IIoT Applications

Insaf Ullah [1], Noor Ul Amin [2], Mahdi Zareei [3,*], Asim Zeb [4], Hizbullah Khattak [2], Ajab Khan [4] and Shidrokh Goudarzi [5]

[1] HIET, Hamdard University, Islamabad Campus, Islamabad 44000, Pakistan; insafktk@gmail.com
[2] IT Department, Hazara University, Mansehra, Mansehra 21120, KP, Pakistan; namin@hu.edu.pk (N.U.A.); hizbullahkhattak@yahoo.com (H.K.)
[3] Tecnologico de Monterrey,Escuela de Ingenieria y Ciencias, Monterrey 64849, Mexico
[4] Department of IT, Abbottabad University of Science and Technology, Havelian City 22500, Pakistan; asimzeb1@gmail.com (A.Z.); ajabk66@yahoo.com (A.K.)
[5] Faculty of Advanced Informatics School, Universiti Teknologi Malaysia, Kuala Lumpur 54100, Malaysia; gshidrokh2@live.utm.my
* Correspondence: m.zareei@tec.mx

Received: 18 August 2019; Accepted: 4 October 2019; Published: 8 November 2019

Abstract: Industrial Internet of Things (IIoT) is a new type of Internet of Things (IoT), which enables sensors to merge with several smart devices to monitor machine status, environment, and collect data from industrial devices. On the other hand, cloud computing provides a good platform for storing crowdsourced data of IIoT. Due to the semi-trusted nature of cloud computing and communication through open channels, the IIoT environment needs security services such as confidentiality and authenticity. One such solution is provided by the identity-based signcryption. Unfortunately, the identity-based signcryption approach suffers from the key escrow problem. Certificateless signcryption is the alternative of identity-based signcryption that can resolve the key escrow problem. Here, we propose a lightweight certificateless signcryption approach for crowdsourced IIoT applications with the intention of enhancing security and decreasing the computational cost and communication overhead. The security and efficiency of the proposed approach are based on the hyper elliptic curve cryptosystem. The hyper elliptic curve is the advance version of the elliptic curve having small parameters and key size of 80 bits as compared to the elliptic curve which has 160-bits key size. Further, we validate the security requirements of our approach through automated validation of Internet security protocols and applications (AVISPA) tool with the help of high level protocol specification language (HLPSL). Moreover, our lightweight and secured scheme will attract low resource devices and will become a perk in the environment of IIoT.

Keywords: signcryption; certificateless signcryption; crowdsourcing; IIoT; cloud computing; AVISPA; hyper elliptic curve

1. Introduction

Nowadays, the Internet of Things (IoT) is a growing technology, which not only enables conventional devices to communicate with each other, but also includes modern devices, e.g., sensors, mobile phones, smart camera, etc. Similarly, the merger of IoT with wireless sensor networks has increased the usage in our daily lives, for example, it is used in industrial automation, smart cities, smart transportation, smart homes, and health monitoring [1]. Among these, the Industrial Internet of Things (IIoT) is an emerging type of IoT, which provides smart factory systems for business and industries [2]. IIoT permits the sensors to merge with several smart devices to monitor machine status, environments, and collect data through industrial outturns. Further, IIoT uses the concept of crowdsourcing; it is the way toward the finishing tasks by requesting contributions from a vast gathering of individuals. Currently, crowdsourcing has turned out to be progressively predominant and is being utilized in a number of tasks such as open development and prediction, data collection, and labeling [3]. Furthermore, crowdsourcing empowers utilizing a substantial pool of workers, the nature of contributions that unequivocally relies on expert capacities and inspiration which varies from worker to worker. Specifically, since crowdsourcing experts are normally not specialists and since a few laborers probably won't apply exertion, answers acquired might be wrong [4,5]. Thus, extending the accuracy and effectiveness of crowdsourcing is the basic zone of IIoT research [6,7]. To store such type of huge data in crowdsourced environment cloud computing plays an important role.

Cloud computing provides a suitable platform for IIoT i.e., to store data from a crowdsourced environment with cost efficiency [8]. This system enables the semi trusted cloud server in storing the data of crowdsourced IIoT devices. Further, the sensors integrate with different smart devices to collect data and measure the status of these devices during the industrial process and send it back to the cloud server by utilizing open networks. Due to the semi trusted cloud server and communications through an open network, the security requirements, for example, authenticity, confidentiality, integrity, unforgeability, and non-repudiation, must be ensured for IIoT data. To ensure these security services, one such solution is signature-then-encryption mechanism. This mechanism is not suitable for small IoT devices, because it generates the signature and encryption on a message in two different steps. So to improve the cost efficiency Zheng [9] coined a new type of security technique called signcryption. This method permits the originator of the message by combining the concept of signature and encryption in a single step. The method is also based on public key infrastructure (PKI). In PKI, the participant's public key has some random number which belongs to some group, which does not provide participants with authenticity because group elements cannot offer the identity of the participants [10]. Therefore, to solve this problem the concept of a certificate authority (CA) is usedthat bounds the public key with certificates [11]. Public key infrastructure (PKI) has some deficiencies such as certificate distribution, storage, and manufacturing difficulties [12]. To improve these limitations, Shamir [13] was the first who introduced the concept of identity-based cryptography. The identity-based cryptography allows the participants to generate directly public keys from his/her identity like e-mail and a phone number without using a certificate authority (CA). The private keys for each participant are generated by the key generation center (KGC). The first identity-based signature was introduced by Shamir [13] while the first identity-based encryption scheme was contributed by Boneh and Fanklin [14]. Later on, to combine the functionality of identity-based signature and encryption into a single step, the concept of identity-based signcryption was introduced [15]. Identity-based signcryption schemes are suffering from the key escrow problem. To avoid the escrow problem of the key, Al-Riyami and Paterson [16] proposed a new type of cryptography called certificateless public key infrastructure. In certificateless public key infrastructure, the percipients' full private key contains two parts: the first one is a partial private key which is created by the key generation center and the second one is a secret value which is made by the participants. The first certificateless signcryption was tossed by Barbosa and Farshim [17], which simultaneously fulfills the property of certificateless encryption and signature in a single step.

Normally, the security and efficiency of all the above discussed signcryption schemes are based on some computationally hard problems, for example, RSA, Bilinear pairing, and elliptic curve cryptosystems. The RSA [18,19] is based on a large factorization problem, which is huge and utilizes 1024 bits large key size, parameter, certificate size, and identity size [20]. This is not suitable for lower power IoT devices due to lack of processing resources. Further, bilinear pairing is 14.31 times worse than RSA [21], due to huge pairing and map-to-point function computation. To eliminate the deficiencies of RSA and bilinear pairing, a new type of cryptography called elliptic curve was introduced [22]. To build a new approach, elliptic curve cryptography utilizes fewer parameter sizes, public and private key size, identity, and certificate size, etc., further, the security hardiness and efficiency of the scheme based on 160-bit small keys, as compared to Bilinear pairing and RSA [23]. Still, the 160-bit key is not suitable and affordable for resource hungry IoT devices. A new type, the generalization of elliptic curve called hyper elliptic curve was proposed [24]. The hyper elliptic curve provides the same level security of elliptic curve, bilinear pairing, and RSA, using 80-bit key, identity and certificate size [25,26]. The hyper elliptic curve assumed to be a better choice for low power IoT devices.

1.1. Motivation and Contributions

Nowadays, IIoT is using the concept of crowdsourcing, which has become the hotest research topic in all over the world. On the other hand, cloud technology is most suitable for storing the huge data of the crowdsourced IIoT environment. Further, collecting data from IIoT devices through open channel needs secrecy and accessing data from cloud required authentication. Recently Karati et al. [8] proposed identity-based signcryption scheme for the crowdsourced IIoT applications. The proposed scheme efficiency and security is realized on a bilinear pairing. We investigate the following points in this paper.

- The scheme is using the concept of identity-based cryptosystem which suffers from the key escrow problem;
- Also, they used the mathematical concepts of bilinear pairing for the proposed scheme which needs huge computational power;
- Because of bilinear pairing, the scheme needs high bandwidth;
- The scheme does not fulfill the security property of resisting against a replay attack and forward secrecy;
- They did not validate the security requirement of their proposed scheme by utilizing the formal security validation tools like AVISPA, Scyther, etc.

Also, Karati et al. [27], proposed another scheme by using the concept of certificateless signature for the applications of crowdsourced IIoT. We found the following limitations in this scheme.

- The scheme has just provided the authentication of IIoT crowdsourced data;
- Also, they used the mathematical concepts of bilinear pairing for the proposed scheme which needs high computational power;
- Because of bilinear pairing, the scheme needs high bandwidth.
- The scheme does not fulfill the security property of resisting against replay attack, confidentiality, and forward secrecy;
- The authentication of the scheme is not validated through the formal security validation tools like AVISPA, Scyther, etc.

Moreover, we studied all the existing certificateless signcryption schemes. We found that these schemes are based on hard problems like elliptic curve, RSA, and bilinear pairing. Because of the heavy parameters and key sizes of these hard problems (elliptic curve, RSA, and bilinear pairing), the existing schemes are suffering from high computational cost and communication overhead. Also, all the existing certificatelesssigncryption schemes are not resisting against the replay attack and nor validated through the formal security validation tools like AVISPA, Scyther, etc. As we already discussed in the introduction section, the hyper elliptic curve is the new version of the elliptic curve which provides the same security level of elliptic curves, RSA, and bilinear pairing with low key size. So keeping in view the above discussion, to remove all the above limitations, we present a new scheme, called a lightweight and provable secured certificateless signcryption scheme for crowdsourced IIoT applications. We explain our contributions in the following steps.

- We design certificatelesssigncryption for crowdsourced IIoT based on the hyper elliptic curve;

- We remove the key escrow problem;
- We will showthe results for proving the efficiency in term of computational cost and communication overhead;
- Our scheme will provide security services such as confidentiality, anti-replay attack, integrity, authentication, sender authentication, message authentication, and unforgeability, respectively;
- We will validate our scheme security services through a well-known simulation tool AVISPA with the help of HLPSL language.

1.2. Outlines of Paper

The paper is organized such that Section 1 provides the introduction; Section 2 presents a comprehensive literature review; Section 3 provides the basic preliminaries of the hyper elliptic curve, and the basic notations used in the scheme; Section 4 shows the detail about the proposed model; Section 5 elaborates the generic model for certificatelesssigncryption; Section 6 provides the practical construction of the proposed scheme; Section 7 exhibits the correctness of the new scheme; Section 8 includes the detail discussion about the security properties of the new scheme; Section 9 discusses the computational cost; Section 10 illustrates the communication overhead details; Section 11 provides the final conclusion; and at the end, AVISPA code and simulation results are provided in the Appendix A.

2. Related Work

In modern communication systems, information security plays a vital role to secure critical data/information because people communicate through the public network. To secure critical data/information required to be hidden from illegal access (confidentiality), knowing about the message originator (authentication), protection from modification (integrity), and availability for a legal user when he/she requires [28]. Therefore, confidentiality can be ensured through encryption procedures, while the integrity and authenticity can be assured by using digital signature methods. In the old days before sending documents, the sender had to first sign the document and then encrypt it which is called the signature then encryption mechanism. But this method has some limitations such as it requires more machine cycles and more energy which affect the efficiency of the system.

To improve such deficiencies, Zheng [9] tossed a new scheme called signcryption that fulfills the task of signature and encryption in one step. The scheme is based on the public key infrastructure (PKI) and suffered from the deficiencies, for example, certificate distribution, storage, and manufacturing difficulties. Hence, to remove such deficiencies, Malone Lee [15] coined a new topic, by combining the capabilities of the identity-based signature and identity-based encryption into a single step, called identity-based signcryption. After the first identity-based signcryption scheme, a number of identity-based signcryption schemes were introduced in [29–35]. Hence, identity-based signcryption schemes are suffering from the key escrow problem.

Then, to avoid the escrow problem of the key, a certificateless signcryption was tossed by Barbosa and Farshim [17], which simultaneously fulfills the property of certificateless encryption and signature in a single step. After this scheme, another certificateless signcryption scheme [36] was contributed, but this scheme was based on the random oracle model. Hence, the random oracle model was not supposed to be used in a practical manner, therefore, to remove this shortcoming, the pioneer standard model-based certificateless signcryption was proposed by Liu et al. [37] in 2010. The limitation of this scheme is that Selvi etal. [38] show in their paper, this scheme cannot resist against a public key replacement attack. In the same year, by using the one-time Schnorr-based signature to the user's public key, Jin et al. [39] give the improved version of the scheme proposed in [38]. After a year, in the Liu et al. scheme [40], Weng et al. [41] point out that the two malicious-but-passive KGCs attack. These attacks were managed in the improved scheme [40]. Also, in 2013, Miao et al. [42] pointed out the scheme [39] suffered from two types of public key replacement attacks. The attack one is same as [38] attack, the other one is a new kind of attack. After some time, there are two standard model-based certificateless signcryption schemes proposed by Xiong [43] and Cheng et al. [44], and up till now there are no flaws reported in these two schemes. In 2016, Abdul Wahida and Masahiro Mamb [45] contributed an elliptic curve-based certificateless signcryption and gave the implementations in Javascript. They gave comparisons with respect to computational and communication costs of the proposed and some existing schemes. They claimed from their comparisons that the newly presented scheme is better than existing schemes. A new standard model-based certificateless signcryption scheme was projected by Caixue et al [46]. They proved their scheme security requirements using the hard problem; a modified decisional bilinear Diffie–Hellman problem and unforgeability security requirement by assuming through a square computational Diffie–Hellman problem. Parvin Rastegari and Mehdi Berenjkoub [47] proposed another scheme called efficient certificatelesssigncryption based on the standard model. Their analysis shows that the presented scheme is more secure and efficient as compared to all the random oracle model-based certificatelesssigncryption schemes existing in the literature. In 2017, Yu And Yang [48] designed and analyzed a new certificateless signcryption scheme without bilinear pairing. The scheme security requirements are proven through the random oracle model. In addition, they claimed, the newly presented algorithm is more attractive for applications like an email system, online auction, and private contractual signature. Xi-Jun et al. [49] present the cryptanalysis of a pairing-free certificateless signcryption scheme [48]. They pointed out that their scheme could be totally broken since confidentiality and unforgeability are not captured. Zhou [50] proposed a new certificateless signcryption approach without using the concept of the random oracle model. The scheme security hardiness is based on bilinear pairing and security proofs realized on the standard model. Liling and Wancheng [51] proposed a new certificateless signcryption based on the efficiency and the hardiness of the elliptic curve cryptosystem. They claimed that the newly designed scheme is secured and needs a low computational cost due to elliptic curve low parameter and key size. Luo and Ma [52], proposed a certificatelesshybrid signcryption for to provide the best solutions forcloud storage.

3. Preliminary

The concept of the hyper elliptic curve was first introduced by N. Koblitz [53], which is the generalized form of elliptic curve [54]. Unlike the elliptic curve point, the points of the hyper elliptic curvecannot be derived from a group, it computes the additive Abelian group which is derived from the divisor. In contrast with elliptic curves, the hyper elliptic curve has acceptable constancy with a small base field size. According to the lower size of parameters, with the same security level of elliptic curves and RSA, the hyper elliptic curve is attractive in fewer hardware resource devices [55]. Let f be the ultimate field and suppose f^{*} is to be the algebraic closure of ultimate field f. Then the hyper elliptic curve $H_{\mathcal{E}}$ of genus $\mathcal{G}$ where $\mathcal{G} > 1$ over the ultimate field f which can be defined as $H_{\mathcal{E}}$: $\beta^2 + \hbar(\alpha)\beta = \mathrm{F}(\alpha)$, where $(\alpha, \beta) \in f^* \times f$. Further, $\hbar(\alpha) \in f(\alpha)$ is a polynomial and the degree of this is at most $\mathcal{G}$ and $\mathrm{F}(\alpha) \in f(\alpha)$is the monic polynomial and have degree is $2\mathcal{G} + 1$. The divisor of the hyper elliptic curve is the pair of polynomials and can be represented by using the Mumford [56]. The most important factor of every cryptographic system is the discrete logarithm problem in some Abelian group. Suppose there is a randomly selected number x from the Abelian group and computing $x.\mathcal{D} = \mathcal{D} + \mathcal{D} + \mathcal{D} + \ldots \ldots \ldots + \mathcal{D}$ is scalar multiplication of divisors. And it is said to be a hyper elliptic curve discrete logarithm problem because finding the random number x from $x.\mathcal{D} = \mathcal{D} + \mathcal{D} + \mathcal{D} + \ldots \ldots \ldots + \mathcal{D}$ is infeasible. Also, the Table 1 shows the symbols/notations used in the algorithm.

Table 1. Notations used in the scheme.

No	Notation	Description
1	$H_{\mathcal{E}}$	Hyper elliptic curve
2	$\mathcal{D}$	Divisor of hyper elliptic curve
3	$\mathcal{G}$	Means genus on a hyper elliptic curve
4	$\hbar$	Irreversible hash function
5	ID_s	Identity of the IIoT data owner/ CLSR
6	ID_r	Identity of the data consumer/ CLUR
7	$\mathcal{Y}_s$,$\mathcal{Y}_r$	The public keys of data owner/ CLSR and data consumer/ CLUR
8	$\mathcal{P}_s = (\mathcal{X}_s,\delta_s)$	The private key pair of IIoT data owner/ CLSR
9	$\mathcal{P}_r = (\mathcal{X}_r,\delta_r)$	The private key pair of data consumer/ CLUR
10	n	It is the largest prime number of $H_{\mathcal{E}}$ and $n = 2^{80}$
11	N_c	It is the nonce
12	m,$\mathcal{C}$	Represents the plaintext and cipher text
13	K	Shared secret key
14	$\mathrm{E}_{K,}$ D_K	Means encryption and decryption
15	S	Means digital signature
16	Ω	Means signcryption tuple

4. Proposed Model

Here, in Figure 1, we present our proposed model consisting of four entities calledthe network manager (NMR), crowdsourced IIoT, cloud server, and receiver, respectively. Note that, the NMR acts like a key generation center in the certificateless public key cryptosystem. When the process began, the crowdsourced IIoT and receiver gave their identities to the NMR and the NMR generates the partial private keys for both on the behalf of their identities. Further, the NMR is responsible for generating all the public parameters, the master secret, and public key and publishing the public parameters and master public key publicly. The crowdsourced IIoT is responsible for generating the certificateless signcryptionin IIoT plaintext, by using his private key and shared secret key and then sending this signcrypted text to the cloud. A cloud server is responsible for storing and processing of IIoT data for the users if required. If the receiver required the crowedsourced IIoT data, then it is obligatory for him to request first to the cloud for the signcrypted text. Then, the cloud gives this signcrypted text to the receiver, and the receiver performs the unsigncryption process by using his private key and the shared secret key.

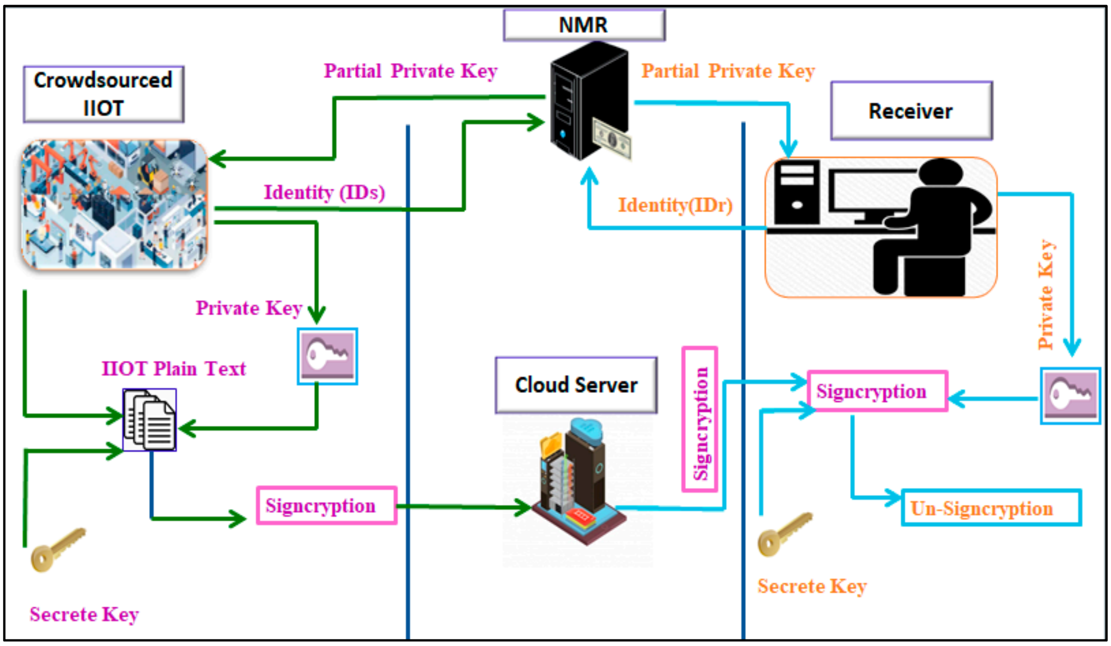


Figure 1. Of the proposed scheme.

5. Generic Model for Certificateless Signcryption

Our proposed scheme consists of six sub steps, namely, setup (STP), user key generation (UKG), set partial private key (SPPK), set private key (SPK), set private key (SPK), certificateless signcryption (CLSN), and CL-unsigncryption (CLUS), respectively.

5.1. Setup (STP)

This algorithm is executed by the network manager (NMR) which plays the role of key generation center, further, the NMR selects all the public parameters param ($f^*, f, H\mathcal{E}, \hbar, \alpha, \beta$), master secret key $m\beta$ and master public key $m\sigma$. After selecting and generating the public parameters and keys, NMR will publish the master public key $m\sigma$ and public parameters param ($f^*, f, H\mathcal{E}, \hbar, \alpha, \beta$) publicly and keep secret the master secret key $m\beta$.

5.2. User Key Generation (UKG)

This algorithm is run by each user with identity ID$\mathcal{u}$ for selecting the secret value $\mathcal{X}\mathcal{u}$ and computing a public key $\mathcal{Y}\mathcal{u}$.

5.3. Set Partial Private Key (SPPK)

The NMR executed this algorithm, by taking each user's identity ID$\mathcal{u}$ to produce a partial private key for every participant ($\mathcal{L}u$, δu). Later on, by using the safe channel NMR sends the partial private key = ($\mathcal{L}u$, δu) to each user.

5.4. Set Private Key (SPK)

This algorithm is executed by each user by taking input the received partial private key ($\mathcal{L}u$, δu) and identity ID$\mathcal{u}$ of each user to produce the private key ($\mathcal{X}\mathcal{u}$, δu) for each participant.

5.5. CertificatelessSigncryption (CLSN)

This algorithm is usually run by the IIoT data owner. It takes the private key pair $\mathcal{P}$s = ($\mathcal{X}$s, δs) of Cl-Signrypter (CLSR)/ data owner, the identity IDs of CLSR and identity IDr of Cl-Un-Signcrypter (CLUR)/ data consumer, a plain text m, and the public key of CLUR $\mathcal{Y}$r to produce the signcryption tuple Ω = (C, S, H, $\mathcal{Z}$).

5.6. CL-Unsigncryption (CLUS)

This algorithm is executed by the data consumer after getting the signcryption tuple Ω = (C, S, H) from the data owner side. It takes input the signcryption tuple Ω = (C, S, H, $\mathcal{Z}$), the private key pair $\mathcal{P}$r = ($\mathcal{X}$r, δr) and public key $\mathcal{Y}$r of CLUR, the identity of CLUR IDr and IDs CLSR, and the public key $\mathcal{Y}$s of CLSR for verification of signature and decryption of encrypted text

6. Proposed CertificatelessSigncryption

In this phase, we practically construct the certificateless signcryption for crowdsourced IIoT. The following steps explain the construction of our new scheme.

6.1. STP

The NMR, first of all, randomly picks their master secret key $m\beta$ from {1,2,3, ...,n − 1} and produces the master public key as: $m\sigma = m\beta.\ \mathcal{D}$. Also, selects the public parameters param ($f^*, f, \mathcal{E}, \hbar$, α, β) and published a param ($f^*, f, H\mathcal{E}, \hbar, \alpha, \beta$) and $m\sigma$ publicly.

6.2. SPPK

The NMR first selects a random ru from {1,2,3, ..., n − 1}, calculates $\mathcal{L}u = ru.\ \mathcal{D}$, then compute $\delta u = ru + m\beta$ (ID, $\mathcal{L}u$, $\mathcal{Y}u$), finally sends $\psi = (\mathcal{L}u, \delta u)$ to each user with identity IDu by using the secured channel.

6.3. UKG

Each user with identity IDu select randomly secret value $\mathcal{X}u$ from {1,2,3, ..., n−1}, further, compute the public key as: $\mathcal{Y}u = \mathcal{X}u.\mathcal{D}$.

6.4. SPK

In this phase, each user with identity IDu, after getting the partial private key from the NMR produces the private key $\mathcal{P}u = (\mathcal{X}u, \delta u)$.

6.5. CLSN

In this phase, the CLSR takes his own private key pair $\mathcal{P}$s = ($\mathcal{X}$s, δs), its own identity IDs, massage m, and the public key $\mathcal{Y}$r and identity IDr of CLUR is an input and computes the signcryption tuple $\Omega = (C, S, H, \mathcal{Z})$, after then, transmits it to the CL-Un-Signcrypter by using the insecure channel. For this purpose the CLSR first randomly picks a number Y from {1,2,3, ..., n−1}, picks a fresh nonce Nc from {1,2,3, ..., n−1}, calculates $\gamma = \mathcal{L}$r + $m\sigma$.(IDr,$\mathcal{L}r$,$\mathcal{Y}r$), computes the hash value $H = \hbar(m$, IDs, Nc), computes $\mathcal{Z} = Y.$, generates a secret key for the encryption K = (Y. (γ +), $\mathcal{Z}$, IDr, $\mathcal{L}r$, $\mathcal{Y}r$), produces the ciphertext C = EK(m, IDs, Nc), computes the digital signature $S = \mathcal{X}$s + H. (Y + δs) mod n, and at the end sends the tuple $\Omega = (C, S, H, \mathcal{Z})$ to the CL-Un-Signcrypter.

6.6. CLUS

This algorithm is executed by the receiver after getting the signcryption tuple $\Omega = (C, S, H)$ from the CLSN side. It takes input the signcryption tuple $\Omega = (C, S, H, \mathcal{Z})$, the private key $\mathcal{P}$r =< xr, δr>, and public key $\mathcal{Y}$r of CLUR, identity of CLUR IDr and CLSR IDs, and the public key $\mathcal{Y}$s of CLSR for the verification of digital signature and decryption of the signcrypted text. Thus, the CLUR first computes the secret key K = ($\mathcal{Z}$. (xr + δr), $\mathcal{Z}$, IDr, $\mathcal{L}r$, $\mathcal{Y}r$), recovers the plaintext from ciphertext (m, IDs, Nc) = DK(C), compute β= $\mathcal{L}$s + $m\sigma$.(IDs, $\mathcal{L}$s, $\mathcal{Y}$s), and finally accepts the signcrypted text if $S.\ \mathcal{D} = \beta + \mathcal{Z}.(m$, IDs, Nc) + $\mathcal{Y}$s. (m, IDs, Nc) is hold.

7. Correctness

The CLUR can recover the secret key easily if the following computations are successfully holding.
K = ($\mathcal{Z}$. ($\mathcal{X}$r + δr), $\mathcal{Z}$, IDr, $\mathcal{L}r$, $\mathcal{Y}r$)
K = ($\mathcal{Z}$. ($\mathcal{X}$r + δr)) = ($\mathcal{Z}$. ($\mathcal{X}$r + δr))
= ($\mathcal{Z}$. ($\mathcal{X}$r + (rr + $m\beta$ (IDr,$\mathcal{L}$r, $\mathcal{Y}$r)))) where δr = rr + $m\beta$ (IDr, $\mathcal{L}$r, $\mathcal{Y}$r)
= (Y. $\mathcal{D}$. ($\mathcal{X}$r + (rr + $m\beta$ (IDr,$\mathcal{L}$r, $\mathcal{Y}$r)))) where $\mathcal{Z} = Y.\ \mathcal{D}$
= (Y. ($\mathcal{X}$r.$\mathcal{D}$ + (rr.$\mathcal{D}$ + $m\beta.\mathcal{D}$ (IDr,$\mathcal{L}$r, $\mathcal{Y}$r))))
= (Y. ($\mathcal{Y}$r + ($\mathcal{L}$r+.(IDr, $\mathcal{L}$r, $\mathcal{Y}$r)))) where $\mathcal{Y}$r = $\mathcal{X}$r.$\mathcal{D}\mathcal{L}$r = rr. $\mathcal{D}$ and $m\sigma = m\beta.\mathcal{D}$
= (Y. ($\mathcal{Y}$r + γ)) where $\gamma = \mathcal{L}$r + $m\sigma$.(IDr,$\mathcal{L}r$,$\mathcal{Y}r$)
= K, hence proved.

Moreover, if the conflict occurs between CLSR and CLUR, then the key generation center resolves it by using the following computations.
$S.\ \mathcal{D} = \beta + \mathcal{Z}.(m$, IDs, Nc) + $\mathcal{Y}$s. (m, IDs,Nc)
While δs.$\mathcal{D}$ = (rs + $m\beta$ (IDs,$\mathcal{L}$s, $\mathcal{Y}$s)). $\mathcal{D}$
= (rs.$\mathcal{D}$ + $m\beta$. $\mathcal{D}$ (IDs,$\mathcal{L}$s, $\mathcal{Y}$s))
= ($\mathcal{L}$s +.(IDs, $\mathcal{L}$s, $\mathcal{Y}$s)) where $\mathcal{L}$s = rs.$\mathcal{D}$ and $m\sigma = m\beta.\mathcal{D}$
($\mathcal{L}$s +.(IDs, $\mathcal{L}$s, $\mathcal{Y}$s))= β

Then,
$S. \mathcal{D} = (\delta \mathrm{s} + H.(Y + \mathcal{X}\mathrm{s})). \mathcal{D}$
$= (\delta \mathrm{s} + (m, \mathrm{IDs,Nc}).(Y + \mathcal{X}\mathrm{s})). \mathcal{D}$ where $H = (m, \mathrm{IDs, Nc})$
$= (\delta \mathrm{s} +.(\hbar(m, \mathrm{IDs, Nc}) + \mathcal{X}\mathrm{s}.(\hbar(m, \mathrm{IDs, Nc}))). \mathcal{D}$
$= (\delta \mathrm{s}.\mathcal{D}+ Y. \mathcal{D} ((m, \mathrm{IDs,Nc}) + \mathcal{X}\mathrm{s}. \mathcal{D} ((m, \mathrm{IDs,Nc}))$
$= (\beta + ((m, \mathrm{IDs,Nc}) + \mathcal{Y}\mathrm{s} (\hbar(m, \mathrm{IDs, Nc})))$ where $\delta \mathrm{s}. \mathcal{D} = \beta$, $Y. \mathcal{D} = \mathcal{Z}$, and $\mathcal{X}\mathrm{s}.\mathcal{D} = \mathcal{Y}\mathrm{s}$
$= (\beta + \mathcal{Z} ((m, \mathrm{IDs,Nc}) + \mathcal{Y}\mathrm{s} (\hbar(m, \mathrm{IDs, Nc})))$ hence hold.

8. Security Analysis

This phase presents the security analysis of our designed approach. Our design scheme ensures the security requirements such as confidentiality, the resistance against replay attack, integrity, authenticity, and unforgeability.

8.1. Confidentiality

Our method ensures the requirements of confidentiality. In our method, if the intruder wants to steal the original contents of a message, then he must know about the secret key K. Thus, for knowing the secret key intruder must perform the following steps:

Step 1: Intruder easily gets the secret key K, if an intruder computes the Equation (1). For this, Intruder must need the secret random number Y from the Equation (2). Hence, getting $\mathcal{V}$ from the Equation (2), it is infeasible for the intruder and an equivalent is like solving the hyper elliptic curve discrete logarithm problem.

$$\mathrm{K} = (Y\,(\gamma + \mathcal{Y}),\mathcal{Z}, \mathrm{ID}r, \mathcal{L}r, \mathcal{Y}r) \tag{1}$$

$$\mathcal{Z} = Y. \mathcal{D} \tag{2}$$

Step 2: Intruder also gets easily the secret key K by computing Equation (3). Hence, computing Equation (3) intruder requires the private keys $< x\mathrm{r}, \delta \mathrm{r} >$ of CLSR from the Equations (4), (5) and (6). For this intruder must solve two hyper elliptic curve discrete logarithm problems which are infeasible.

$$\mathrm{K} = (\mathcal{Z}.\ (\mathcal{X}\mathrm{r} + \delta \mathrm{r}\ ), \mathcal{Z}, \mathrm{ID}r, \mathcal{L}r, \mathcal{Y}r) \tag{3}$$

$$\mathcal{Y}\mathrm{r} = \mathcal{X}\mathrm{r}.\mathcal{D} \tag{4}$$

$$\delta \mathrm{r} = r\mathrm{r} + m\beta\ (\mathrm{ID}r, \mathcal{L}\mathrm{r}, \mathcal{Y}\mathrm{r}) \tag{5}$$

$$\mathcal{L}\mathrm{r} = r\mathrm{r}.\mathcal{D} \tag{6}$$

8.2. Replay Attack

Our method resists against the security service of reply attack. In our scheme, if intruder wants to send the past messages to the CLUR, intruder generates and sends a tuple like $\Omega = (C, S, H,)$ with fresh Nc to CLUR. Therefore, later the CLUR checks the validity of nonce Nc, if the nonce is fresh then the message is from the original sender otherwise not.

8.3. Integrity

Our designed mechanism enables the CLUR to verify that the ciphertext was modified or not at the time communication by using the Equation (7). If the intruder changes the ciphertext C to $C/$, then the received plain text must be changed from m to $m/$. Thus, our designed method meets the security service of integrity because the generated digital signature of m is not the same as the digital signature of $m/$.

8.4. Authentication

In this section, we discuss two types of authentication, the first one is sender authentication and the second one is the message authentication.

(1) Sender Authentication

Our designed method meets the sender authentication security requirement. In our mechanism, the CLUR used the public key pair ($\mathcal{L}$s, $\mathcal{Y}$s) and identity IDs of CLSR for verifying the originator of the message. The CLSR generates the digital signature by using their private key pair ($\mathcal{X}$s, δs). Thus, our designed method granted the security requirement of sender authentication.

(2) Message Authentication

In our scheme, before the delivery of the message to the CLUR, the CLSR generates a digital signature on it like $S = \mathcal{X}\text{s} + H.(Y + \delta\text{s})$; by using the private key pair ($\mathcal{X}$s, δs) of CLSR. The CLUR can verify the message by using the received signature S, if the Equation (7) is held; it means that the message is coming from authentic CLSR.

$$S.\,\mathcal{D} = \beta + \mathcal{Z}.(m,\,\text{IDs},\,\text{Nc}) + \mathcal{Y}\text{s}.\;(m,\,\text{IDs},\,\text{Nc}) \tag{7}$$

8.5. Unforgeability

In our proposed mechanism, if the intruder tries to generate a valid signature, then he/she should first compute the Equation (8). For this, the intruder needs the private key pair ($\mathcal{X}$s, δs) of the CLSR. Therefore, to know about the private keys; an intruder should first compute the Equation (9) and (10). For Equation (9) Intruder has to solve the hyper elliptic curve discrete problem which is infeasible, while for (10) intruder needs the random *rs* from the Equation (11) which also leads to infeasibility. Hence our designed technique ensures the security service of unforgeability.

$$S = \mathcal{X}\text{s} + H.(Y + \delta\text{s}) \tag{8}$$

$$\mathcal{Y}\text{s} = \mathcal{X}\text{s}.\mathcal{D} \tag{9}$$

$$\delta\text{s} = r\text{s} + m\beta\;(\text{IDs},\,\mathcal{L}\text{s},\,\mathcal{Y}\text{s}) \tag{10}$$

$$\mathcal{L}\text{s} = r\text{s}.\mathcal{D} \tag{11}$$

9. Computational Cost

In this section we compare our approach with recently contributed certificateless signature for IIoT AIK [27], an identity-based signcryption scheme for IIoT ASGMPM [8], and the certificatelesssigncryption schemes such as PM [47], HB [48], ZC [50], LW [51], and LM [52] with respect to computational cost. We consider the major operations, for example, bilinear pairing operation (BP), exponential (EX), elliptic curve point multiplication (EM), and hyper elliptic curve divisor multiplication (HDM) in proposed scheme and those of ASGMPM [8], AIK [27], PM [47], HB [48], ZC [50], LW [51], and LM [52] for computational cost comparisons. Table 2 shows the consuming major operations of the proposed scheme and the existing schemes. We also provide comparisons of computational cost with respect to milliseconds (ms). For this purpose, we use the results of schemes [21,26] which shows the computational time in milliseconds of different cryptographic operations such as single BP consumes 14.31 ms, EX consumes 1.25 ms, EM consumes 0.97 ms, and HDM consumes 0.48 ms, respectively. For this experiment, the authors of the schemes [21,26], used the hardware resources, for example, 8 GB RAM, Intel Core i74510UCPU, and 2.0GHz processor. The software resources like window 7 and C++ with Multi-Precision Integer and Rational Arithmetic C Library (MIRACL). Further, Table 3 shows more clear comparisons shown in milliseconds (ms). Additionally, we use the reduction formula [57], for the reduction of the computational cost of the proposed and those of ASGMPM [8], AIK [27], PM [47], HB [48], ZC [50], LW [51], and LM [52], respectively. So for the reduction, we use the values of Table 3 and the following steps represent the clear reduction:

- The proposed scheme reduced at the computational time from AIK [27] is 4EX + 1BP − 7HDM/ 4EX + 1BP = 22.19 − 3.36/22.19 = 0.84*100 = 84.85%.
- The proposed scheme computational cost reduction From ASGMPM [8] is 6EX + 2BP − 7HDM/6EX + 2BP= 40.44 − 3.36/40.44 = 0.91*100 = 91.69%.
- The proposed scheme computational cost reduction from PM [47] is 8EX + 9BP − 7HDM/8EX + 9BP = 144.73 − 3.36/144.73 = 0.97*100 = 97%.
- The proposed scheme computational cost reduction from HB [48] is 10EX − 7HDM/10EX = 19.7 − 3.36/19.7 = 0.82*100 = 82.94%.
- The proposed scheme computational cost reduction from ZC [50] is 12EX + 5BP − 7HDM /12EX + 5BP = 95.19 − 3.36/95.19 = 0.96*100 = 96.47%.
- The proposed scheme computational cost reduction from LW [51] is 12EM − 7HDM /12EM = 11.64 − 3.36/11.64 = 0.71*100 = 71.13%.
- The proposed scheme computational cost reduction from LM [52] is 7EM − 7HDM /7EM = 6.79 − 3.36/6.79 = 0.50*100 = 50.51%.

Table 2. Comparisons with respect to major operations.

Schemes	Signature/Signcryption	Verification/Un-Signcryption	Total
AIK [27]	2EX	2EPX + 1BP	4EX + 1BP
ASGMPM [8]	4EX	2EX + 2BP	6EX + 2BP
PM [47]	7EX + 2BP	1EX + 7BP	8EX + 9BP
HB [48]	5EX	5EX	10EX
ZC [50]	7EX + 1BP	5EX + 4BP	12EX + 5BP
LW [51]	7EM	5EM	12EM
LM [52]	4 EM	3 EM	7 EM
Proposed Scheme	4HDM	3HDM	7HDM

Table 3. Comparisons with respect to milliseconds (ms).

Schemes	Signature/Signcryption (ms)	Verification/ Un-Signcryption (ms)	Total (ms)
AIK [27]	3.94	18.25	22.19
ASGMPM [8]	7.88	32.56	40.44
PM [47]	42.59	102.14	144.73
HB [48]	9.85)	9.85	19.7
ZC [50]	28.1	67.09	95.19
LW [51]	6.79	4.85	11.64
LM [52]	3.88	2.91	6.79
Proposed Scheme	1.92	1.44	3.36

10. Communication Overhead

Communication overhead means that how many extra bits will be sent along with the actual message. Bandwidth is an important part of the data communication channels. If the scheme needs fewer bits along with the message at the time of sending, it means the channel required low bandwidth. Here, we compare our proposed scheme with those ASGMPM [8], AIK [27], PM [47], HB [48], ZC [50], LW [51], and LM [52], with respect to communication overheads and shows that our scheme needs very little communication overhead. Our results based on [57,58], in which |G1| ≌ |G2| ≌ |G| ≌ 1024 bits for bilinear pairing, |P| ≌ 1024 bits for the discrete logarithm problem, |q| ≌ 160 bits for elliptic curve, |n| ≌ 80 bits for a hyper elliptic curve, and $|m|$ = 1024 bits, respectively. The following are the required communication overhead for ASGMPM [8], AIK [27], PM [47], HB [48], ZC [50], LW [51], and LM [52]:

- The communication overhead for AIK [27] is $|m|$ + 2|G| = 3072.
- The communication overhead for ASGMPM [8] is $|m|$ + 2|G1| + 2|G2| + |G| = 6144.

- The communication overhead for PM [47] is $|m|$ + 4|G1| + 1|G2| = 6144, for HB [48] is $|m|$ + 4|P| = 5120.
- The communication overhead for ZC [50] is $|m|$ + 3|G1| + 3|G2| = 7166.
- The communication overhead for LW [51] is $|m|$ + 2|q| = 1344, for LM [52] is $|m|$ + 2|q| = 1344.
- The communication overhead for the proposed scheme is $|m|$ + 3|n| = 1264.

Moreover, we conclude that the proposed scheme is better in the following steps:

- Our scheme is3072 − 1264/3072 = 58.85% efficient than AIK [27] in term of communication overhead.
- The proposed scheme is 6144 − 1264/6144 = 79.42% efficient than ASGMPM [8] and PM [47] in terms of communication overhead.
- Our scheme is also 5120 − 1264/5120 = 75.31% efficient than HB [48] concerning communication overhead.
- Our scheme is 7166 − 1264/7166 = 82.36% proficient than ZC [50] concerning communication overhead.
- Our scheme is 1344 − 1264/1264 = 5.95% efficient than LW [51] and LM [52] with respect to communication overhead.

11. Conclusions

This paper contributes, a lightweight and provable secured certificatelesssigncryption approach for the crowdsourced IIoT applications. The efficiency and security hardiness of the proposed approach is based on the hyper elliptic curve cryptography. The security requirements of the proposed approach are validated through the most famous tool, automated validation of Internet security protocols and applications (AVISPA). The proposed approach ensures the security services of resistance against replay attack, confidentiality, integrity, authentication (of sender and message), non-repudiation and unforgeability. Furthermore, our designed approach has reduced in computational cost from 71.13% to 97% and in communication overhead from 5.95% to 82.36%, compared to the existing approaches. Due to the cost efficiency and enhanced security, our approach is more attractive for low power devices of IIoT.

12. Future Work

In the future, we are planning to conduct a practical test to measure performance.

Author Contributions: Conceptualization, I.U. and N.U.A.; methodology, H.K.; validation, I.U., N.U.A. and H.K.; formal analysis, S.G.; investigation, I.U.; data curation, I.U.; writing—original draft preparation, I.U.; writing—review and editing, N.U.A., M.Z., A.Z., H.K., A.K. and S.G.; visualization, I.U.; supervision, A.K.; project administration, N.U.A.; funding acquisition, M.Z.

Funding: This research was funded by SEP-CONACyT Research Project, grant number 255387. The APC was funded by School of Engineering and Sciences and the Telecommunications Research Group at Tecnologico de Monterrey.

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

Appendix A Implementation and Validation

Here, in this phase, we are going to implement and validate the security requirement of our approach practically by using the most popular simulation tool AVISPA [59]. So, the AVISPA is a well-known automatic tool, which works under two validation state, namely, SAFE if the cryptographic scheme resists against man-in-the-middle attack and UNSAFE if the scheme cannot resist against man-in-the-middle attack. Here, in Figure A1, we show the basic structure of AVISPA tool. For specifying the cryptographic scheme, AVISPA used a well-known role oriented language, named HLPSL (high-level protocol specification language). To provide an interface to the user, AVISPA merged with SPAN. Therefore, to check whether the cryptographic scheme is either SAFE or UNSAFE, the user first converts the pseudo-code of the proposed algorithm to the HLPSL source code. After, that HLPSL2IF translator, translate the HLPSL code into the intermediate format (IF). The HLPSL2IF translator checks the scheme security under the four backend checkers named, On-the-fly-Model-Checker (OF-MC), CL-based Attack Searcher (CL-AtSe), SAT-based Model-Checker (SATMC), and Tree-Automata-based Protocol Analyzer (TA4SP), respectively. According to the user requirements, each backend has its own functionality as discussed in [60,61]. According to [57], HLPSL2IF check the protocol security keeping in view the initial knowledge under these backends.

So, we provide the HLPSL code for our proposed scheme, including three main roles named, CLSN, CLUS, the session, the environment, and goals, respectively. We did this experiment by using the hardware resources, for example, Haier Win8.1 PC, Inter (R) Core (TM) i3-4010U CPU @ 1.70 GHz, supporting 64-bit operating system, x64-based processor. Also, the software resources such as Oracle VM Virtual Box (version: 5.2.0.118431) and SPAN (version: SPAN-Ubuntu-10.10-light_1). In Table A1, we provide the HLPSL code for CLSN, in which the role_Clsn represents CLSR, Ys:public_key means the public key $\mathcal{Y}$s of CLSR, Yr:public_key represents the public key $\mathcal{Y}$r of CLUR, SND,RCV:channel(dy)means sending and receiving the messages throughDolev-Yao model channel, H represents $H = \hbar(m, \mathrm{IDs}, \mathrm{Nc})$, {En(M′)}_K′ means the encryption $C = \mathrm{EK}(m, \mathrm{IDs}, \mathrm{Nc})$ through secret key, M′e message m, K′ represent the secret key $\mathrm{K} = (Y.\,(\gamma + \mathcal{Y}r\,), \mathcal{Z}, \mathrm{ID}r, \mathcal{L}r, \mathcal{Y}r\,)$,{H′.Y′}_inv(Ys) represents the digital signature $S = \mathcal{X}\mathrm{s} + H.(\,Y + \delta\mathrm{s})$,and inv(Ys) means the private key pair ($\mathcal{X}$s,δs) of CLSR. In Table A2, we illustrate the HLPSL code for CLUS. In this code, {Nc′}_Yr means encryption of nonce through the public key $\mathcal{Y}$r of CLUR and RCV(Clsn.{En(M′)}_K′.{H′.Y′}_inv(Ys)) means the received signcrypted tuple $\Omega = (C, S, H, \mathcal{Z})$. In Tables A3 and A4, we show the code for a session and environment role. For the intruder, the AVISPA used a special identifier (i).We test the Code of our approach under the functionality of two backends of AVISPA tool, called ASTE and OFMC.

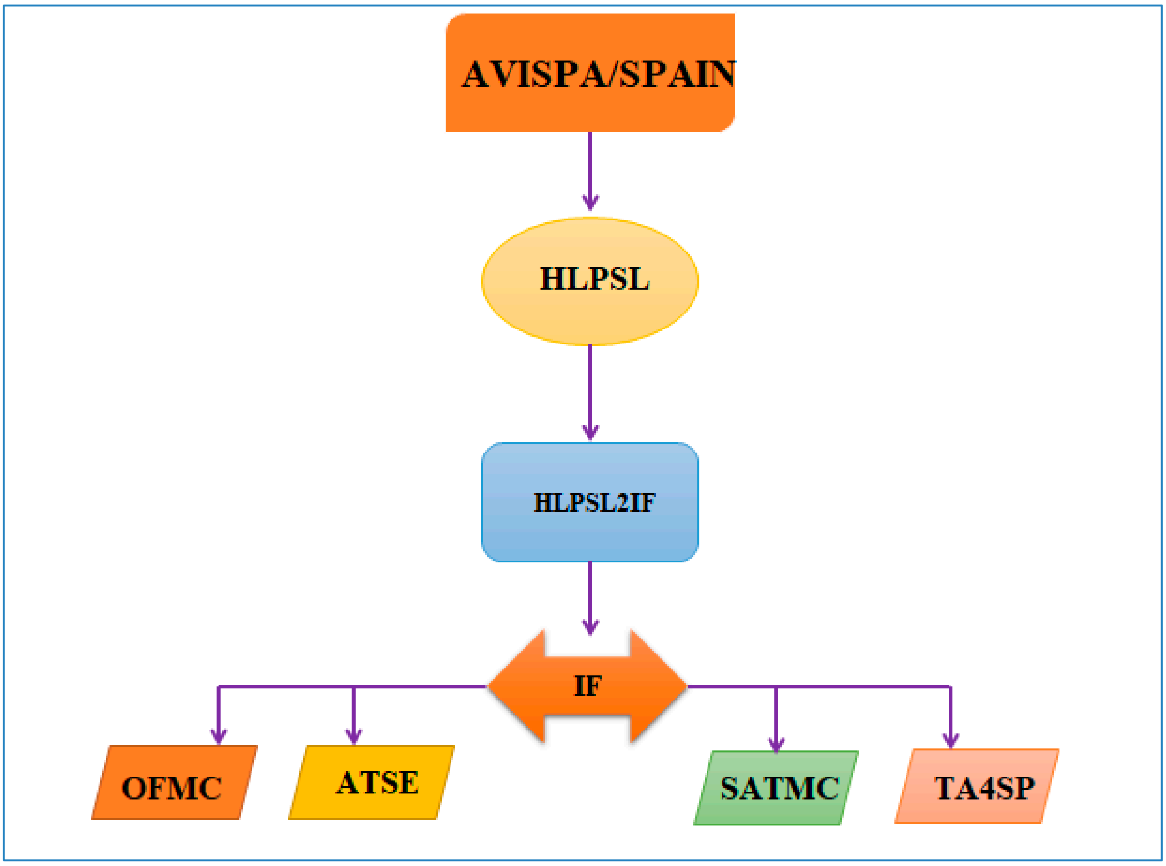


Figure A1. Structure of AVISPA.

Table A1. HLPSL code for CLSN.

```
role role_Clsn(Clsn:agent,
Clus:agent,Ys:public_key,
Yr:public_key,
SND,RCV:channel(dy))
played_byClsn
def=
local
State:nat,Nc:text,H:text,Y:text,
M:text,En:hash_func,K:symmetric_key
init
State := 0
transition
1. State=0 /\ RCV(start) =|> State’:=1 /\ SND(Clsn.Clus)
2. State=1 /\ RCV(Clus.{Nc’}_Yr) =|> State’:=2 /\ Y’:=new()
/\ H’:=new() /\ K’:=new() /\ M’:=new()
/\ secret(M’,sec_2,{Clsn})
/\ witness(Clsn,Clus,auth_1,M’)
/\ SND(Clsn.{En(M’)}_K’.{H’.Y’}_inv(Ys))
end role
```

Table A2. HLPSL code for CLUS.

```
role role_Clus(Clsn:agent,Clus:agent,
Ys:public_key,Yr:public_key,
SND,RCV:channel(dy))
played_byClus
def=
local
State:nat,Nc:text,H:text,Y:text,M:text,
En:hash_func,K:symmetric_key
init
State := 0
transition
1. State=0 /\ RCV(Clsn.Clus) =|> State’:=1
/\ Nc’:=new() /\ SND(Clus.{Nc’}_Yr)
6. State=1 /\ RCV(Clsn.{En(M’)}_K’.{H’.Y’}_inv(Ys)) =|> State’:=2
/\ request(Clus,Clsn,auth_1,M’)
/\ secret(M’,sec_2,{Clsn})
end role
```

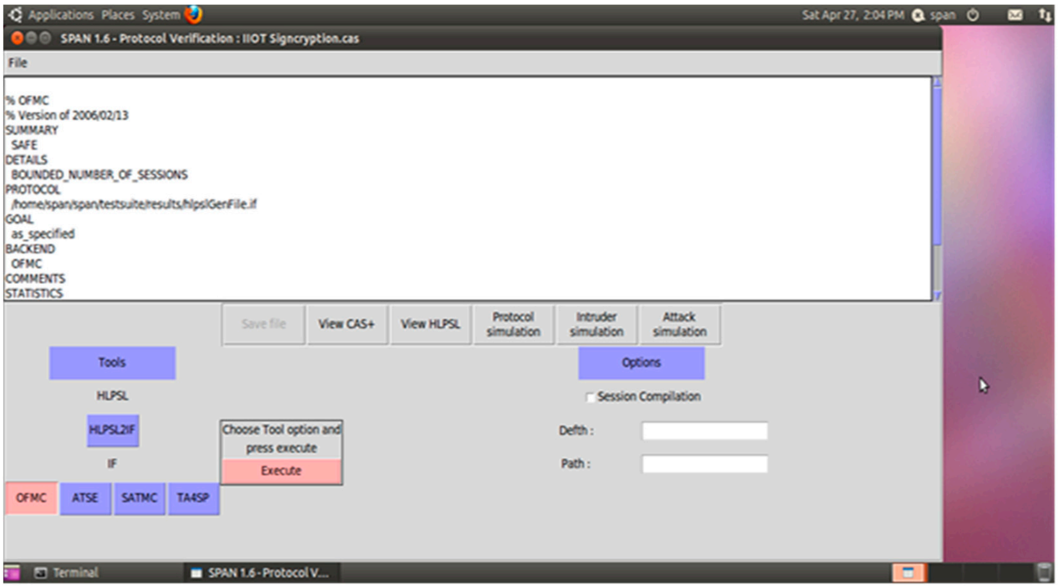
Table A3. Se sion role.

```
role session1(Clsn:agent,Clus:agent,
Ys:public_key,Yr:public_key)
def=
local
SND2,RCV2,SND1,RCV1:channel(dy)
composition
role_Clus(Clsn,Clus,Ys,Yr,SND2,RCV2) /\ role_Clsn(Clsn,Clus,Ys,Yr,SND1,RCV1)
end role

role session2(Clsn:agent,Clus:agent,
Ys:public_key,Yr:public_key)
def=
local
SND1,RCV1:channel(dy)
composition
role_Clsn(Clsn,Clus,Ys,Yr,SND1,RCV1)
end role
```

Table A4. Environment role.

```
role environment()
def=
const

hash_0:hash_func,ys:public_key,alice:agent,bob:agent,yr: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,ys,yr)
end role
```

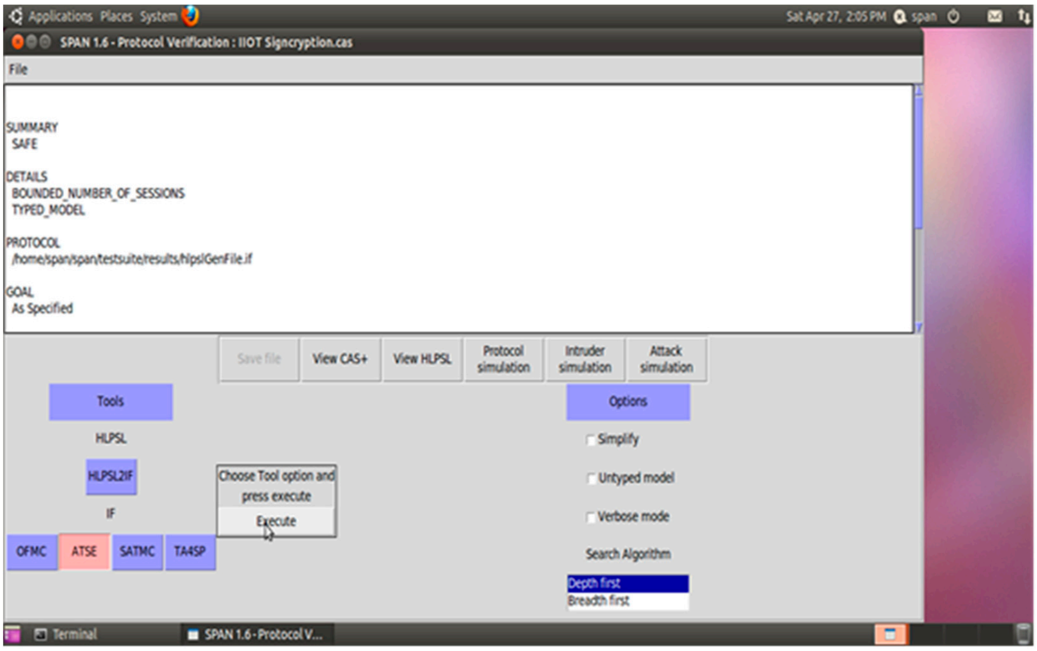
In Figure A2, we provide the simulation result of our proposed scheme under the functionality of the OFMC back-end of the AVISPA tool, which is safe. The OFMC accomplishes schemes misrepresentation and restricted authorization by investigating the change framework portrayed by an IF detail in an interest-driven manner. OFMC actualizes various right and complete representative methods. It strengthens the detail of mathematical properties of cryptographic properties and composed and un-typed scheme models.



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

Figure A2. Simulation results of OFMC.

Also in Figure A3, we offer the simulation result of our proposed scheme under the functionality of the ATSE back-end of AVISPA tool, which is safe. CL-AtSe works in a measured manner and is available to augmentations for taking care of the mathematical properties of cryptographic operators. It supports type-imperfection discovery and handles associativity of message concatenation.



```
SUMMARY
 SAFE

DETAILS
 BOUNDED_NUMBER_OF_SESSIONS
 TYPED_MODEL

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

GOAL
 As Specified
```

□ Simplify
□ Untyped model
□ Verbose mode
Search Algorithm
Depth first
Breadth first

Figure A3. Results of ATSE.

References

1. Shen, L.; Ma, J.; Liu, X.; Wei, F.; Miao, M. A Secure and Efficient ID-Based Aggregate Signature Scheme for Wireless Sensor Networks. *IEEE Internet Things J.* **2017**, *4*, 546–554. [CrossRef]
2. Liao, Y.; Loures, E.D.F.R.; Deschamps, F. Industrial Internet of Things: A Systematic Literature Review and Insights. *IEEE Internet Things J.* **2018**, *5*, 4515–4525. [CrossRef]
3. Cohensius, G.; Ben-Porat, O.; Meir, R.; Amir, O. Efficient Crowdsourcing via Proxy Voting. *arXiv* **2018**, arXiv:1806.06257.
4. Kazai, G.; Kamps, J.; Koolen, M.; Milic-Frayling, N. Crowdsourcing for book search evaluation: Impact of hit design on comparative system ranking. In Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, Beijing, China, 25–29 July 2011; pp. 205–214.
5. Vuurens, J.; de Vries, A.P.; Eickhoff, C. How much spam can you take? An analysis of crowdsourcing results to increase accuracy. In Proceedings of the ACM SIGIR Workshop on Crowdsourcing for Information Retrieval (CIR11), Beijing, China, 28 July 2011; pp. 21–26.
6. Wais, P.; Lingamneni, S.; Cook, D.; Fennell, J.; Goldenberg, B.; Lubarov, D.; Marin, D.; Simons, H. Towards building a high-quality workforce with mechanical turk. In Proceedings of the Computational Social Science and the Wisdom of Crowds (NIPS), Whistler, BC, Canada, 10 December 2010; pp. 1–5.
7. Quinn, A.J.; Bederson, B.B. Human computation: A survey and taxonomy of a growing field. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Vancouver, BC, Canada, 7–12 May 2011; pp. 1403–1412.
8. Karati, A.; Islam, S.H.; Biswas, G.P.; Bhuiyan, M.Z.A.; Vijayakumar, P.; Karuppiah, M. Provably Secure Identity-Based Signcryption Scheme for Crowdsourced Industrial Internet of Things Environments. *IEEE Internet Things J.* **2018**, *5*, 2904–2914. [CrossRef]
9. Zheng, Y. Digital signcryption or how to achieve cost (signature & encryption)? cost (signature)+ cost (encryption). In *Advances in Cryptology-CRYPTO'97*; Springer: Berlin/Heidelberg, Germany, 1997; pp. 165–179.
10. Chen, H.; Chen, S.; Xu, H.; Hu, B. A Security Scheme of 5G Ultradense Network Based on the Implicit Certificate. *Wirel. Commun. Mob. Comput.* **2018**, *2018*, 8562904. [CrossRef]
11. Kumar, N.C.; Basit, A.; Singh, P.; Venkaiah, V.C. Lightweight Cryptography for Distributed PKI Based MANETS. *arXiv* **2018**, arXiv:1804.06313.
12. Ullah, S.; Marcenaro, L.; Rinner, B. Secure Smart Cameras by Aggregate-Signcryption with Decryption Fairness for Multi-Receiver IoT Applications. *Sensors* **2019**, *19*, 327. [CrossRef]

13. Shamir, A. Identity-based cryptosystems and signature schemes. In Proceedings of the Advances in Cryptology-CRYPTO'84, LNCS 196, Santa Barbara, CA, USA, 19–22 August 1984; pp. 47–53.
14. Boneh, D.; Franklin, M. Identity-based encryption from the weil pairing. In Proceedings of the Advances in Cryptology-CRYPTO'01, LNCS 2139, Santa Barbara, CA, USA, 19–23 August 2001; pp. 213–229.
15. Malone-Lee, J. Identity Based Signcryption. In *Cryptology ePrint Archive Report*; IACR: Lyon, France, 2002.
16. Al-Riyami, S.; Paterson, K. Certificateless Public Key Cryptography. In Proceedings of the International Conference on the Theory and Application of Cryptology and Information Security, Berlin, Germany, 30 November–4 December 2003; pp. 452–473.
17. Barbosa, M.; Farshim, P. Certificateless Signcryption. In *Proceedings of the ASICC, Tokyo, Japan, 18–20 March 2008*; ACM: New York, NY, USA, 2008; pp. 18–20.
18. Suárez-Albela, M.; Fraga-Lamas, P.; Fernández-Caramés, T.M. A Practical Evaluation on RSA and ECC-Based Cipher Suites for IoT High-Security Energy-Efficient Fog and Mist Computing Devices. *Sensors* **2018**, *18*, 3868. [CrossRef]
19. Yu1, M.; Zhang, J.; Wang, J.; Gao1, J.; Xu1, T.; Deng, R.; Zhang, Y.; Yu, R. Internet of Things security and privacy-preserving method through nodes differentiation, concrete cluster centers, multi-signature, and blockchain. *Int. J. Distrib. Sens. Netw.* **2018**, *14*, 1–15.
20. Braeken, A. PUF Based Authentication Protocol for IoT. *Symmetry* **2018**, *10*, 352. [CrossRef]
21. Zhou, C.; Zhao, Z.; Zhou, W.; Mei, Y. Certificateless Key-Insulated Generalized Signcryption Scheme without Bilinear Pairings. *Secur. Commun. Netw.* **2017**, *2017*, 8405879. [CrossRef]
22. Kumari, S.; Karuppiah, M.; Das, A.K.; Li, X.; Wu, F.; Kumar, N. A secure authentication scheme based on elliptic curve cryptography for IoT and cloud servers. *J. Supercomput.* **2017**, *74*, 6428–6453. [CrossRef]
23. Omala, A.; Mbandu, A.; Mutiria, K.; Jin, C.; Li, F. Provably Secure Heterogeneous Access Control Scheme for Wireless Body Area Network. *J. Med Syst.* **2018**, *42*. [CrossRef] [PubMed]
24. Tamizhselvan, C.; Vijayalakshmi, V. An Energy Efficient Secure Distributed Naming Service for IoT. *Int. J. Adv. Stud. Sci. Res.* **2019**, *3*.
25. Naresh, V.; Sivaranjani, R.; Murthy, N.V.E.S. Provable secure lightweight hyper elliptic curve-based communication system for wireless sensor networks. *Int. J. Commun. Syst.* **2018**, *31*, e3763. [CrossRef]
26. Rahman, A.; Ullah, I.; Naeem, M.; Anwar, R.; 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.* **2018**, *9*. [CrossRef]
27. Karati, A.; Islam, S.H.; Karuppiah, M. Provably Secure and Lightweight Certificateless Signature Scheme for IIoT Environments. *IEEE Trans. Ind. Inform.* **2018**, *14*, 3701–3711. [CrossRef]
28. Mehmood, A.; Noor-Ul-Amin, I.; Umar, A.I. Public Verifiable Generalized Authenticated Encryption based on Hyper Elliptic Curve. *J. Appl. Environ. Biol. Sci.* **2017**, *7*, 194–200.
29. Ming, Y.; Wang, Y. Cryptanalysis of an identity based signcryption scheme in the standard model. *Int. J. Netw. Secur. Appl.* **2016**, *18*, 165–171.
30. Nayak, B. A secure ID-based signcryption scheme based on elliptic curve cryptography. *Int. J. Comput. Intell. Stud.* **2017**, *6*, 150. [CrossRef]
31. Ashibani, Y.; Mahmoud, Q.H. An efficient and secure scheme for smart home communication using identity-based signcryption. In Proceedings of the IEEE 36th International Performance Computing and Communications Conference (IPCCC), Phoenix, AZ, USA, 10–12 December 2017.
32. Huang, Y.; Yang, J. A Novel Identity-Based Signcryption Scheme in the Standard Model. *Information* **2017**, *8*, 58. [CrossRef]
33. Tsai, T.-T.; Huang, S.-S.; Tseng, Y.-M. SIBSC: Separable Identity-Based Signcryption for Resource-Constrained Devices. *Informatica* **2017**, *28*, 193–214. [CrossRef]
34. Yu, H.; Wang, Z.; Li, J.; Gao, X. Identity-Based Proxy Signcryption Protocol with Universal Composability. *Secur. Commun. Netw.* **2018**, *2018*, 9531784. [CrossRef]
35. Zhou, C.; Zhou, W.; Dong, X. Provable certificateless generalized signcryption scheme. *Des. Codes Cryptogr.* **2012**, *71*, 331–346. [CrossRef]
36. Shi, W.; Kumar, N.; Gong, P.; Zhang, Z. Cryptanalysis and improvement of a certificateless signcryption scheme without bilinear pairing. *Front. Comput. Sci.* **2014**, *8*, 656–666. [CrossRef]
37. Liu, Z.; Hu, Y.; Zhang, X.; Ma, H. Certificateless signcryption scheme in the standard model. *Inf. Sci.* **2010**, *180*, 452–464. [CrossRef]

38. Selvi, S.S.D.; Vivek, S.S.; Rangan, C.P. Security weaknesses in two certificateless signcryption schemes. In *IACR Cryptology ePrint Archive*; IACR: Lyon, France, 2010.
39. Jin, Z.P.; Wen, Q.Y.; Zhang, H. A supplement to Liu et al.'s certificateless signcryption scheme in the standard model. In *IACR ePrint Archive*; IACR: Lyon, France, 2010.
40. Weng, J.; Yao, G.; Deng, R.H.; Chen, M.-R.; Li, X. Cryptanalysis of a certificateless signcryption scheme in the standard model. *Inf. Sci.* **2011**, *181*, 661–667. [CrossRef]
41. Miao, S.; Zhang, F.; Li, S.; Mu, Y. On security of a certificateless signcryption scheme. *Inf. Sci.* **2013**, *232*, 475–481. [CrossRef]
42. Xiong, H. Toward certificateless signcryption scheme without random oracle. In *IACR ePrint Archive*; IACR: Lyon, France, 2014.
43. Cheng, L.; Wen, Q.Y. An improved certificateless signcryption in the standard model. *Int. J. Netw. Secur.* **2015**, *17*, 597–606.
44. Wahid, A.; Mambo, M. Implementation of Certificateless Signcryption based on Elliptic Curve Using Javascript. *Int. J. Comput. Inform. (IJCANDI)* **2016**, *1*, 90–100.
45. Zhou, C.; Gao, G.; Cui, Z. Certificateless Signcryption in the Standard Model. *Wirel. Pers. Commun.* **2016**, *92*, 495–513. [CrossRef]
46. Rastegari, P.; Berenjkoub, M. An Efficient Certificateless Signcryption Scheme in the Standard Model. *ISC Int. J. Inf. Secur.* **2017**, *9*, 3–16. [CrossRef]
47. Yu, H.; Yang, B. Pairing-Free and Secure Certificateless Signcryption Scheme. *Comput. J.* **2017**, *60*, 1187–1196. [CrossRef]
48. Lin, X.-J.; Sun, L.; Qu, H.; Liu, D. Cryptanalysis of A Pairing-Free Certificateless Signcryption Scheme. *Comput. J.* **2017**, *61*, 539–544. [CrossRef]
49. Zhou, C. Certificateless Signcryption Scheme without Random Oracles. *Chin. J. Electron.* **2018**, *27*, 1002–1008. [CrossRef]
50. Cao, L.; Ge, W. Analysis of Certificateless Signcryption Schemes and Construction of a Secure and Efficient Pairing-free one based on ECC. *Ksii Trans. Internet Inf. Syst.* **2018**, *12*, 4527–4547.
51. Luo, W.; Ma, W. Secure and Efficient Data Sharing Scheme Based on Certificateless Hybrid Signcryption for Cloud Storage. *Electronics* **2019**, *8*, 590. [CrossRef]
52. Koblitz, N. Hyper elliptic crypto systems. *J. Cryptol.* **1989**, *1*, 139–150. [CrossRef]
53. Wollinger, T.; Pelzl, J.; Paar, C. Cantor versus Harley: Optimization and analysis of explicit formulae for hyperelliptic curve cryptosystems. *IEEE Trans. Comput.* **2005**, *54*, 861–872. [CrossRef]
54. Wollinger, T.; Pelzl, J.; Wittelsberger, V.; Paar, C.; Saldamli, G.; Koç, Ç. Elliptic and hyperelliptic curves on embedded μP. *Acm Trans. Embed. Comput. Syst.* **2004**, *3*, 509–533. [CrossRef]
55. Fan, X.; Wollinger, T.; Gong, G. Efficient explicit formulae for genus 3 hyperelliptic curve cryptosystems over binary fields. *IET Inf. Secur.* **2007**, *1*, 65. [CrossRef]
56. Zhou, C. An improved lightweight certificateless generalized signcryption scheme for mobile-health system. *Int. J. Distrib. Sensor Netw.* **2019**, *15*, 1–16. [CrossRef]
57. Ullah, I.; Amin, N.U.-A.; Naeem, M.; Khattak, H.; Khattak, S.J.; Ali, H. A Novel Provable Secured Signcryption Scheme PSSS: A Hyper-Elliptic Curve-Based Approach. *Mathematics* **2019**, *7*, 686. [CrossRef]
58. Yu, S.; Lee, J.; Lee, K.; Park, K.; Park, Y. Secure Authentication Protocol for Wireless Sensor Networks in Vehicular Communications. *Sensors* **2018**, *18*, 3191. [CrossRef]
59. Qiu, S.; Xu, G.; Ahmad, H.; Guo, Y. An enhanced password authentication scheme for session initiation protocol with perfect forward secrecy. *PLoS ONE* **2018**, *13*, e0194072. [CrossRef]
60. Ali, R.; Pal, A.K. Three-Factor-Based Confidentiality-Preserving Remote User Authentication Scheme in Multi-server Environment. *Arab. J. Sci. Eng.* **2017**, *42*, 3655–3672. [CrossRef]
61. Jung, J.; Kang, D.; Lee, D.; Won, D. An Improved and Secure Anonymous Biometric-Based User Authentication with Key Agreement Scheme for the Integrated EPR Information System. *PLoS ONE* **2017**, *12*, e0169414. [CrossRef]