Hindawi
Security and Communication Networks
Volume 2023, Article ID 9759520, 1 page
https://doi.org/10.1155/2023/9759520

Retraction

Retracted: ECC-Based Authenticated Key Exchange Protocol for Fog-Based IoT Networks

Security and Communication Networks

Received 11 July 2023; Accepted 11 July 2023; Published 12 July 2023

This article has been retracted by Hindawi following an investigation undertaken by the publisher [1]. This investigation has uncovered evidence of one or more of the following indicators of systematic manipulation of the publication process:

(1) Discrepancies in scope
(2) Discrepancies in the description of the research reported
(3) Discrepancies between the availability of data and the research described
(4) Inappropriate citations
(5) Incoherent, meaningless and/or irrelevant content included in the article
(6) Peer-review manipulation

The presence of these indicators undermines our confidence in the integrity of the article's content and we cannot, therefore, vouch for its reliability. Please note that this notice is intended solely to alert readers that the content of this article is unreliable. We have not investigated whether authors were aware of or involved in the systematic manipulation of the publication process.

Wiley and Hindawi regrets that the usual quality checks did not identify these issues before publication and have since put additional measures in place to safeguard research integrity.

We wish to credit our own Research Integrity and Research Publishing teams and anonymous and named external researchers and research integrity experts for contributing to this investigation.

The corresponding author, as the representative of all authors, has been given the opportunity to register their agreement or disagreement to this retraction. We have kept a record of any response received.

References

[1] U. Iqbal, J. Bhola, M. Jayasudha et al., "ECC-Based Authenticated Key Exchange Protocol for Fog-Based IoT Networks," *Security and Communication Networks*, vol. 2022, Article ID 7264803, 15 pages, 2022.

Hindawi
Security and Communication Networks
Volume 2022, Article ID 7264803, 15 pages
https://doi.org/10.1155/2022/7264803

Research Article

ECC-Based Authenticated Key Exchange Protocol for Fog-Based IoT Networks

Ummer Iqbal[1], **Jyoti Bhola**[2], **M. Jayasudha**[3], **Mohd Wazih Ahmad**[4], **Rahul Neware**[5], **Arvind R. Yadav**[6], and **Fraol Waldamichael Gelana**[7]

[1]*National Institute of Technology Srinagar, Srinagar, J&K, India*
[2]*Electronics & Communication Engineering Department, National Institute of Technology, Hamirpur, India*
[3]*VIT University Chennai Campus, Chennai, India*
[4]*ASTU Adama, Adama, Ethiopia*
[5]*Department of Computing, Mathematics and Physics, Høgskulen På Vestlandet, Bergen, Norway*
[6]*Parul Institute of Engineering & Technology, Parul University, Vadodara, India*
[7]*Ethiopian Artificial Intelligence Institute, Addis Ababa, Ethiopia*

Correspondence should be addressed to Ummer Iqbal; drkhaniqbalummer@gmail.com and Fraol Waldamichael Gelana; fraol.gelana@aic.et

Received 11 March 2022; Revised 25 April 2022; Accepted 10 May 2022; Published 22 June 2022

Academic Editor: Mukesh Soni

Fog computing is one of the prominent technology that bridges the gap between IoT nodes and cloud servers. For increasing the efficiency at the fog level, a fog federation can be employed. Fog federation at the fog level can be controlled by the fog coordinator. However, the information exchange between the fog coordinator and IoT nodes needs to be secured. Recently, a lightweight secure key exchange (LKSE) protocol for secure key exchange for fog federation was proposed. In this paper, the cryptanalysis of the LKSE is carried out. The cryptanalysis indicates that LKSE is vulnerable to spoofing and man in the middle attacks. To overcome the limitation of the LKSE, a design of an ECC-based secure key exchange protocol for IoT devices and fog coordinators is proposed. The security strength of the designed method has been evaluated using BAN logic and the random oracle model. Simulations on AVISPA have been performed for automatic security verification of the proposed method. A detailed security and functional comparison of the proposed scheme with LKSE have also been carried out.

1. Introduction

IoT-based smart city applications have acquired significant attraction over the years [1–4]. The various IoT-based smart applications include smart water, smart health, smart grid, etc. The introduction of the Internet of Things (IoT) has resulted in an unprecedented creation of massive and diverse amounts of data, referred to as data explosions [5]. On the other hand, while cloud computing has been an effective means to process and store this data, difficulties such as real-time access, latency, and network capacity limitations need to be handled if cloud computing is employed. To solve this issue, a new computing paradigm called fog computing has been proposed [6]. Fog computing brings cloud services to the network's edge, thus improving low-latency, mobility, network bandwidth, security, and privacy.

A typical fog computing block diagram is shown in Figure 1 [7]. The architecture comprises the end device layer, fog layer, and cloud computing layer. In the end device layer, smart devices are deployed to monitor and sense various attributes depending upon the context of the application. The end device layer typically involves resource constraint devices. Because of the resource constraint nature, the security with in the device layer is an emerging research area. The fog layer comprises of fog nodes. The fog node [6] is the core component of the fog layer. Fog nodes are strongly associated with smart end devices. Fog nodes can be set as stand-alone fog nodes that interact among themselves to

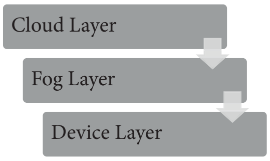


Figure 1: Fog computing architecture.

supply the service or can be federated to build clusters to implement a certain fog computing capability. The cloud layer comprises of different servers which can be utilized for online/offline analysis, etc. Some of the applications of fog computing include linked vehicles, smart grids and smart cities, and real-time analytics.

The security of the sensed data sent from the end device layer to fog nodes is of paramount importance and is an active area of research. The primary security requirements of the data communicated between smart end-devices and fog nodes include confidentiality, integrity, data freshness, and authentication of the sensed data. Confidentiality of the sensor data ensures that the data in the legitimate form are accessible only to the intended receiver [8]. If an attacker can eavesdrop on the message exchanges, confidentiality must ensure that the eavesdropped messages cannot be deciphered. Confidentially is enforced using encryption and decryption techniques. The encryption technique scrambles the sensed data in such a way that intended receiver with proper decryption process can recover the sensed data. The integrity authentication of the sensed data ensures that the sensed data messages are not altered in transit by an adversary [9]. An authentication mechanism is required to validate and verify whether the legitimate network entities are communicating with each other or not. Nonrepudiation guarantees the responsibility of action. Any security protocol targeting low power nodes must oblige to its constraints [10, 11] and must be formally verified [12].

The bedrock of the security requirements discussed above is a secure authenticated key exchange between end devices and the fog nodes. Schemes have been presented in the literature for secure key establishment. Sun et al. [13], Jia et al. [14], Wahid et al. [15], Chen et al. [16], Zheng and Chang [17], and Chen et al. [18] proposed some of the schemes which are reported to be safe and support authenticated key exchange. However, all these schemes are not suitable for the fog federation environment [19, 20]. CE-SKE [19] and LKSE [20] are some of the recent schemes proposed for secure key exchange in the fog federation. CE-SKE claims to support mutual authentication and key exchange; however, this scheme is not lightweight. The second scheme called as LKSE is an improved scheme in terms of efficiency as it is based on elliptical curve cryptography. However, it can be shown that both schemes are vulnerable to spoofing attacks and man the middle attack.

In this paper, a lightweight ECC-based authenticated key exchange scheme has been presented. The proposed scheme is resilient to all major security attacks while being functionally optimal in terms of resource overheads. The paper reviews the LKSE scheme in terms of security limitation and proposes a design of lightweight authenticated key exchange scheme which overcomes the limitations of LKSE.

1.1. Elliptical Curve Cryptography. An Elliptical Curve $E_p(a,b)$ over a finite prime field F_p is defined as (1):

$$E_p(a,b)\colon\ y^2 = x^3 + ax + b, \tag{1}$$

$$\text{Where, } \blacktriangle = \left(4a^3 + 27b^2 ! = 0\right). \tag{2}$$

The computational hardness of the elliptical curve cryptography is based on the elliptical curve discrete log problem (ECDLP). Given two points $P(x,y)$ and $Q(x,y) \in E(a,b)$ such that: $Q(x,y) = n.P(x,y)$ where n is a scalar, ECDLP states that it is computationally infeasible to find n [10, 11].

1.2. Contributions. The contributions of the paper are as follows:

(1) A review and cryptanalysis of the LKSE have been made carried out to indicate that the scheme is vulnerable to various attacks.

(2) ECC-based secure key exchange protocol for IoT devices and fog coordinator is proposed with better specifications as compared to the existing schemes.

(3) The proposed scheme has been formally validated using AVISPA [21, 22]. The verification results indicate that the scheme is safe and is resilient to man in the middle attack and replay attack.

(4) The validity of the proposed protocol has also been evaluated using BAN logic [23].

1.3. Paper Organization. The remainder of the paper is laid out as follows. Section 2 reviews and highlights the weaknesses of the LSKE Scheme. In Section 3, the details of the designed protocol are presented. In Section 4, security analysis of the designed scheme has been presented. AVISPA simulation details are presented in Section 5. BAN logic analysis has been carried out in Section 6. Finally, in Section 7, the comparative analysis of the designed scheme is presented.

2. Review and Weakness of the LSKE Scheme

2.1. Review of the LSKE Scheme. The key exchange steps between the end device node and the fog center in the LSKE scheme is given as below:

Step 1: The node computes $A1$ as in (3) and sends it to the fog center.

$$A1 = (IDA\ , IDB\ , TA, Kas). \tag{3}$$

Step 2: The fog center checks $TA < \blacktriangle_T$, if true, then it performs the following:

(1) Stores the Kas.

(2) Chooses the numbers $a, b, p, R1, R2$, and NB.

(3) Calculate equations $H1, B1, H2, B2, H3$, and PB as (4)–(10):

$$H1 = h(IDA\ , IDB\ , Kbs), \tag{4}$$

$$B1 = (IDA\ , IDB\ , Kbs, H1), \tag{5}$$

$$H2 = h(a, b, p, R1, R2), \tag{6}$$

$$B2 = (a, b, p, R1, R2, H2), \tag{7}$$

$$PB = NB * G(R1, R2), \tag{8}$$

$$H3 = h(PB), \tag{9}$$

$$B3 = (PB, TB, H3, B1, B2)Kas. \tag{10}$$

Step 3: Fog center sends $B3$ to node

Step 4: The node checks: $TB < \blacktriangle_T$, if true then it performs the following:

(1) Compute $H1' = h(B1)$, check $H1' = H1$, if true, then store Kbs.
(2) Compute $H2' = h(B2)$, check $H2' = H2$, if true, then store $a, b, p, R1, R2$.
(3) Compute $H3' = h(B3)$, check $H3' = H3$, if true, then store PB.

Step 5: The node selects a random number NA and obtains PA as follows:

$$PA = NA * G(R1, R2). \tag{11}$$

Step 6: The node calculates the common key as follows:

$$K = NA * PB. \tag{12}$$

Step 7: The node calculates $A2$ and $H4$ and sends it to fog center:

$$\begin{gathered} H4 = h(PA), \\ A2 = (PA, TA, H4)Kbs. \end{gathered} \tag{13}$$

Step 8: Fog center computes $H4' = h(PA)$, check $4' = H4$, check $TA < \blacktriangle_T$, then calculate

$$K = NB * PA. \tag{14}$$

2.2. Cryptanalysis of the LSKE Scheme. In this section, the cryptanalysis of the LSKE scheme has been carried out. Considering an active adversary α in the middle, α can spoof the messages and subsequently launch man in the middle attack as given below:

Step 1: The node computes $A1$ as (15) and sends it to the fog center.

$$A1 = (IDA\ , IDB\ , TA, Kas). \tag{15}$$

Step 2: Adversary α, intercepts the message and performs the following steps:

(1) Selects a public key Kas^{α}.
(2) Computes $A1^{\alpha} = (IDA\ , IDB\ , TA^{\alpha}, Kas^{\alpha})$.
(3) Sends $A1^{\alpha}$ to fog center

Step 3: The fog center checks $TA^{\alpha} < \blacktriangle_T$, which evaluates to be true. The fog center then performs the following:

(1) Stores the Kas^{α} key.
(2) Chooses the numbers $a, b, p, R1, R2$, and NB.
(3) Calculates equations $H1, B1, H2, B2, H3$, and PB as (16)–(22):

$$H1 = h(IDA\ , IDB\ , Kbs), \tag{16}$$

$$B1 = (IDA\ , IDB\ , Kbs, H1), \tag{17}$$

$$H2 = h(a, b, p, R1, R2), \tag{18}$$

$$B2 = (a, b, p, R1, R2, H2), \tag{19}$$

$$PB = NB * G(R1, R2), \tag{20}$$

$$H3 = h(PB), \tag{21}$$

$$B3 = (PB, TB, H3, B1, B2)Kas^{\alpha}. \tag{22}$$

Step 4: Fog center sends $B3$ to the fog node

Step 5: Adversary α intercepts the message and performs the following steps:

(1) Decrypts: $(PB, TB, H3, B1, B2)Kas^{\alpha}$ using the private key.
(2) Adversary α selects a random number NA^{α} and obtains PA^{α}as follows:

$$PA^{\alpha} = NA^{\alpha} * G(R1, R2). \tag{23}$$

(3) Adversary α calculates the common key with the fog center as (24):

$$K^{\alpha}_{FC} = NA^{\alpha} * PB. \tag{24}$$

Step 6: Adversary α calculates $A2^{\alpha}$ and $H4^{\alpha}$ and sends it to the fog center:

$$\begin{gathered} H4^{\alpha} = h\left(PA^{\alpha}\right), \\ A2^{\alpha} = \left(PA^{\alpha}, TA^{\alpha}, H4^{\alpha}\right)Kbs. \end{gathered} \tag{25}$$

Step 7: Fog center decrypts $A2^{\alpha}$ and compute $H4^{\alpha'} = h(PA^{\alpha})$, check $4' = H4$, check $, TA^{\alpha} < \blacktriangle_T$, calculate (26)

$$K^{\alpha}_{FC} = NB * PA^{\alpha}. \tag{26}$$

Step 8: Adversary α, further performs the following functions

(1) Adversary α selects a random number NB^α and obtains PB^α as follows:

$$PB^\alpha = NB^\alpha * G(R1, R2). \tag{27}$$

(2) Selects a public key Kbs^α.

(3) Chooses the numbers $a, b, p, R1^\alpha, R2^\alpha$.

(4) Calculate equations $H1^\alpha, B1^\alpha, H2^\alpha, B2^\alpha, H3^\alpha$ and PB^α as follows:

$$H1^\alpha = h(IDA\ , IDB\ , Kbs^\alpha), \tag{28}$$

$$B1^\alpha = (IDA\ , IDB\ , Kbs^\alpha, H1^\alpha), \tag{29}$$

$$\begin{aligned} H2^\alpha &= h(a, b, p, R1^\alpha, R2^\alpha),\\ B2^\alpha &= (a, b, p, R1^\alpha, R2^\alpha, H2^\alpha),\\ PB^\alpha &= NB^\alpha * G(R1^\alpha, R2^\alpha),\\ H3^\alpha &= h(PB^\alpha),\\ B3^\alpha &= (PB^\alpha, TB^\alpha, H3^\alpha, B1^\alpha, B2)Kas. \end{aligned} \tag{30}$$

Step 9: The node decrypts $B3^\alpha$, first checks the time stamp with $TB^\alpha < \blacktriangle_T$, if true then it performs the following:

(1) Compute $H1' = hB1^\alpha)$, check $H1' = H1$, if true, store Kbs.

(2) Compute $H2' = h(B2^\alpha)$, check $H2' = H2$, if true, store $a, b, p, R1^\alpha, R2^\alpha$.

(3) Compute $H3' = h(B3^\alpha)$, check $H3' = H3$, if true, store PB^α.

Step 10: The node selects a random number NA and obtains PA as follows:

$$PA = NA * G(R1, R2). \tag{31}$$

Step 11: The node calculates the common key as follows:

$$K_{FN}^\alpha = NA * PB^\alpha. \tag{32}$$

Step 12: The node calculates $A2$ and $H4$ and sends it to the fog center:

$$\begin{aligned} H4 &= h(PA),\\ A2 &= (PA, TA, H4)Kbs^\alpha. \end{aligned} \tag{33}$$

Step 13: Adversary α intercepts the message and performs the following steps:

(1) Decrypt $A2$ using Kbs^α

(2) Compute $H4' = h(PA)$, check $H4' = H4$, check $TA < \blacktriangle_T$, calculate (34):

$$K_{FN}^\alpha = NB^\alpha * PA. \tag{34}$$

From the above cryptanalysis, we understand that an adversary α, by spoofing the message exchange can execute a Man-in-the-Middle-Attack. Attacker α forms a shared key K_{FC}^α with the fog center, wherein the fog center believes that K_{FC}^α is key formed with the fog node and forms a shared key K_{FC}^α with the fog node, wherein the fog node believes that K_{FC}^α is key formed with the node fog center. The genesis of this attack originates from the fact that there is no complete integrity check on the messages being exchanged as such an adversary α was able to manipulate and spoof the messages.

3. Proposed Scheme

In this section, the ECC-based scheme for secure key exchange protocol for Iota devices and fog coordinator is proposed. The design of the protocol is based on elliptical curve cryptography. The notations used are listed in Table 1. The various phases in the proposed access control protocol include the setup and initialization phase, fog node registration phase, fog center registration phase, and authentication and key establishment Phase.

3.1. Setup and Initialization Phase. The certification authority G_N^{CA} performs the system setup phase. The various steps undertaken in this phase are as follows:

(i) G_N^{CA} chooses an elliptical curve; $E_P(a, b)$ defined as $y^2 = x^3 + ax + b(\text{mod}\, p)$ is chosen where a and $b \in Z_P$ and P is a large prime number.

(ii) The G_N^{CA} chooses G_K^{Pr} and computes $G_K^{Pu}(x, y)$, where $G_K^{Pu}(x, y) = G_K^{Pr}.G(x, y)$

3.2. Fog Node Registration Phase

(i) For each IoT node, Node_I, G_N^{CA} chooses N_K^{Pr} and calculates $N_K^{Pu}(x, y) = N_K^{Pr}.G(x, y)$

(ii) G_N^{CA} creates a signature point $SP_I(x, y)$ for each Node_I as (35):

$$SP_I(x, y) = \left[N_K^{Pr} + G_K^{Pr}\right] * V_{SP_I}^K * H[\text{Node}_I] * G(x, y), \tag{35}$$

where $V_{SP_I}^K$ is the version of the signature $SP_I(x, y)$ and guards its freshness. Initially, the $V_{SP_I}^K = 1.$ for each redeployment of Node_I the $V_{SP_I}^K$ is incremented by 1.

(iii) G_N^{CA} computes the ECDSA signature $[(r_i, s_i)]$ for each Node_I as (36):

$$[(r_i, s_i)]\ = ECDSA - SIG\left(SP_I(x, y)\middle\|N_K^{Pu}(x, y)\middle\|V_{SP_I}^K \text{Node}_I, G_K^{Pr}, E(a, b)\right). \tag{36}$$

Table 1: Symbols and their description.

Symbol	Description
$H()$	Hash function
Node_I	Node with identity i
Fog_{Cen}	Identity of the fog center
G_K^{Pr}	Private key of certification authority
$G_K^{Pu}(x, y)$	Public key of certification authority
N_K^{Pr}	PrivateKey of Node_I
$N_K^{Pu}(x, y)$	Public Key of Node_I
N_F^{Pr}	Private key of Fog_{Cen}
$N_F^{Pu}(x, y)$	Public key of Fog_{Cen}
$[(r_i, s_i)]$	ECDSA signature pair of Node_I
K_{IJ}	Shared key between Node_I and Node_J
$V_{SP_I}^K$	Certificate version of Node_I
$V_{SP_K}^K$	Certificate version of Fog_{Cen}
LSD_I	Last seen certificate version of Node_I in Fog_{Cen}
$G(x, y)$	Generator point of $E_p(a, b)$
$K * G(x, y)$	Scalar point multiplication between K and $G(x, y)$
$P(x, y) + Q(x, y)$	Point addition between $P(x, y)$ and $Q(x, y)$
G_N^{CA}	Certification authority

The ECDSA signature $[(r_i, s_i)]$ computed using the private key of the G_N^{CA} is to thwart any spoofing or malicious manipulation of authentication and key establishment request and response messages between the deployed node and its neighbors. The evaluation of $[(r_i, s_i)]$ during the authentication and key establishment phase ensures that messages exchanged are authentic and their integrity is maintained.

(iv) G_N^{CA} preloads each IoT node, Node_I with the following:

$E_p(a,b), H, SP_I(x, y), G_K^{Pu}(x, y), \text{Node}_I, N_K^{Pr},\ N_K^{Pu}(x, y)$, $[(r_i, s_i)]$ and $V_{SP_I}^K$

3.3. Fog Centre Registration Phase

(i) G_N^{CA} chooses F_K^{Pr} and calculates $F_K^{Pu}(x, y) = F_K^{Pr}.G(x, y)$ for Fog central node-Fog_{Cen}

(ii) G_N^{CA} creates a signature point $(SP_F(x, y)$ for Fog_{Cen} as (37):

$$SP_F(x, y) = \left[F_K^{Pr} + G_K^{Pr}\right] * V_{SP_F}^K * H\left[\text{Fog}_{\text{Cen}}\right] * G(x, y), \tag{37}$$

where $V_{SP_F}^K$ is the version of the signature $SP_F(x, y)$ and guards its freshness.

(iii) G_N^{CA} computes the ECDSA signature $[(r_f, s_f)]$ for Fog_{Cen} as (38):

$$\left[\left(r_f, s_f\right)\right] = ECDSA - SIG\left(SP_F(x, y) \| F_K^{Pu}(x, y) \| V_{SP_F}^K \text{Fog}_{\text{Cen}}, G_K^{Pr}, E(a, b)\right). \tag{38}$$

(iv) Fog_{Cen} stores the following:

$E_p(a,b), H, SP_F(x, y), G_K^{Pu}(x, y), \text{Fog}_{\text{Cen}}, F_K^{Pr}, F_K^{Pu}(x, y)$, $(r_f,\ s_f)$ and $V_{SP_F}^K$.

3.4. Authentication and Key Establishment Phase. The authentication and key establishment phases undertaken between Node_I and the Fog_{Cen} are detailed below:

(i) Node_I sends the authentication and key establishment request A_{NI}^{RR} to Fog_{Cen}

$$\text{Node}_I \longrightarrow \text{Fog}_{\text{Cen}}\colon A_{NI}^{RR} = SP_I(x, y) \| N_K^{Pu}(x, y) \| V_{SP_I}^K \| \text{Node}_I \| [(r_i, s_i)]. \tag{39}$$

(ii) Fog_{Cen} verifies the integrity and the authenticity of A_{NI}^{RR} by computing:

$$EC\ DS\ A - \text{VERIFY}\left(A_{NI}^{RR}, G_K^{Pu}(x, y), E(a, b)\right] \tag{40}$$

If the verification check evaluates to be false, no processing is done, and the request is rejected. However, if the verification check evaluates to be true, Step iii is performed.

(iii) Fog_{Cen} authenticates Node_I by performing the following computational steps:

(a) Fog_{Cen} calculates $V = [V^K_{SP_I} * H[\text{Node}_I]]^{-1}$ where $\text{Node}_I, V^K_{SP_I}$ are received through A^{RR}_{NI}

(b) Fog_{Cen} performs the scalar multiplication of $SP_I(x, y)$ and V as (41):

$$\begin{aligned} R_I(x, y) &= SP_I(x, y) * V, \\ R_I(x, y) &= \left[N^{Pr}_{KI} + G^{Pr}_K\right] * V^K_{SP_I} * H[\text{Node}_I] \\ &\quad * \left[V^K_{SP_I} * H[\text{Node}_I]\right]^{-1} * G(x, y), \\ R_I(x, y) &= \left[N^{Pr}_{KI} + G^{Pr}_K\right] * G(x, y). \end{aligned} \tag{41}$$

(c) Fog_{Cen} calculates the authentication point as (42):

$$\begin{aligned} AP_I(x, y) &= R_I(x, y) + G^{Pu}_K(x, -y), \\ AP_I(x, y) &= N^{Pu}_{KI}(x, y) + G^{Pu}_K(x, y) + G^{Pu}_K(x, -y), \\ AP_I(x, y) &= N^{Pu}_{KI}(x, y). \end{aligned} \tag{42}$$

(d) Fog_{Cen} compares $AP_I(x, y) == N^{Pu}_{KI}(x, y)$ where $N^{Pu}_K(x, y)$ is received through A^{RR}_{NI}. If true, then A^{RR}_{NI} from Node_I is validated and step iv is performed; otherwise, the phase is aborted the phase is aborted.

(iv) Fog_{Cen} computes key with Node_I as follows:

$$\begin{aligned} K_{JI} &= H\left[F^{Pr}_K * V^K_{SP_I} * AP_I(x, y)\right], \\ K_{JI} &= H\left[F^{Pr}_K * N^{Pr}_{KI} * V^K_{SP_I} * G(x, y)\right]. \end{aligned} \tag{43}$$

(v) Fog_{Cen} sends authentication and key establishment response A^{RE}_{Fog} as follows::

$$\begin{aligned} \text{Fog}_{\text{Cen}} &\longrightarrow \text{Node}_I\colon A^{RE}_{\text{Fog}} \\ &= SP_F(x, y) \| F^{Pu}_K(x, y) \| V^K_{SP_F} \| \text{Fog}_{\text{Cen}} \| \left[(r_f, s_f)\right]. \end{aligned} \tag{44}$$

(vi) Node_I verifies the integrity and the authenticity of A^{RE}_{NJ} by computing the following:

$$EC\ DS\ A\_\text{VERIFY}\left(A^{RE}_{\text{Fog}}, G^{Pu}_K(x, y), E(a, b)\right]. \tag{45}$$

If the verification check evaluates to be false, no processing is done, and the request is rejected. However, if the verification check evaluates to be true, Step viii is performed.

(vii) Node_I is authenticates Fog_{Cen} by performing the following computational steps:

(a) Node_I calculates $V = [V^K_{SP_F} * H[\text{Fog}_{\text{Cen}}]]^{-1}$ where $\text{Fog}_{\text{Cen}}, V^K_{SP_F}$ are received through A^{RE}_{Fog}

(b) Node_I performs the scalar multiplication of $SP_F(x, y)$ and V as (46):

$$\begin{aligned} R_F(x, y) &= SP_F(x, y) * V, \\ R_F(x, y) &= \left[F^{Pr}_K + G^{Pr}_K\right] * V^K_{SP_F} * H[\text{Fog}_{\text{Cen}}] \\ &\quad * \left[V^K_{SP_F} * H[\text{Fog}_{\text{Cen}}]\right]^{-1} * G(x, y), \\ R_F(x, y) &= \left[F^{Pr}_K + G^{Pr}_K\right] * G(x, y). \end{aligned} \tag{46}$$

(c) Node_I calculates the authentication point as (47):

$$\begin{aligned} AP_F(x, y) &= R_F(x, y) + G^{Pu}_K(x, -y), \\ AP_F(x, y) &= F^{Pu}_K(x, y) + G^{Pu}_K(x, y) + G^{Pu}_K(x, -y), \\ AP_F(x, y) &= F^{Pu}_K(x, y). \end{aligned} \tag{47}$$

(d) Node_I compares $AP_F(x, y) == F^{Pu}_K(x, y)$ where $F^{Pu}_K(x, y)$ is received through A^{RE}_{Fog} . If true, then A^{RE}_{Fog} from Fog_{Cen} is validated and step v is performed; otherwise, the phase is aborted.

(viii) Node_I computes key with Fog_{Cen} as 60

$$\begin{aligned} K_{JI} &= H\left[N^{Pr}_{KI} * V^K_{SP_F} * AP_F(x, y)\right], \\ K_{JI} &= H\left[F^{Pr}_K * N^{Pr}_{KI} * V^K_{SP_F} * G(x, y)\right]. \end{aligned} \tag{48}$$

(ix) Node_I chooses a nonce $N1$, computes $E_{KJI}[N1H(N1)]$ and sends the following to the Fog_{Cen} :

$$\text{Node}_I \longrightarrow \text{Fog}_{\text{Cen}}\colon E_{KJI}[N1H(N1)]. \tag{49}$$

(x) Fog_{Cen} receives $E_{KJI}[N1H(N1)]$ and decrypts $E_{KJI}[N1H(N1)]$ as $D_{KIJ}[E_{KJI}[N1H(N1)]]$. Fog_{Cen} further calculates $H^1(N1)$ using the $N1$ obtained by decrypting $E_{KJI}[N1H(N1)]$ and verifies $H(N1) == H^1(N1)$. If $H(N1) == H^1(N1)$ is true, the authentication and key exchange process is completed.

4. Security Analysis

4.1. Informal Security Analysis. In this section, the proposed protocol has been evaluated on some of the major security requirements as indicated in [13–20]

(a) Eavesdropping and false injection attacks: To prevent the eavesdropping and the false injection of sensed data, a shared key is established between the and the Fog_{Cen} as follows:

$$K_{JI} = H\left[F^{Pr}_K * N^{Pr}_{KI} * V^K_{SP_I} * G(x, y)\right]. \tag{50}$$

The key K_{JI} can be used with any lightweight cipher to provide basic security primitives of confidentiality, integrity, and authentication of the sensed data.

(b) Impersonation attack: During the setup, and the initialization phase, each node Node_I is preloaded with the following key material:

$$\begin{gathered} E_p(a,b),\ H, SP_I(x,y), G_K^{Pu}(x,y), \text{Node}_I, \\ N_K^{Pr}, N_K^{Pu}(x,y), [(r_i,s_i)],\ DS_I^K. \end{gathered} \tag{51}$$

Let us assume that the Node_I is captured by an adversary α. α has access to all the preloaded material of the Node_I. The complete network security will get compromised if the private key of G_N^{CA} is extracted. The private key G_K^{Pr} is used in the $SP_I(x,y)G_K^{Pu}(x,y)$ and $[(r_i,s_i)]$. However, the adversary α cannot extract the private of G_N^{CA} from $SP_I(x,y)G_K^{Pu}(x,y)$ and $[(r_i,s_i)]$ due to the computational hardness of the elliptical discrete logarithm problem [24–26].

(c) Man-in-the-Middle-Attack: Suppose attacker α wants to undertake a MITM Attack between a Node_I and Fog_{Cen}. To accomplish so, α must fabricate $[SP_{MAL-I}(x,y),\ [(r_{MAL-I},s_{MAL-I})]]$ and $[\mathbb{F},\ [(r_{MAL-F},s_{MAL-F})]]$ so that Node_I and Fog_{Cen} recognize them as authentic signatures. Due to ECDLP [24–26], it is computationally impossible for α to fake $[SP_{MAL-I}(x,y),\ [(r_{MAL-I},s_{MAL-I})]]$ and $[SP_{MAL-F}(x,y),\ [(r_{MAL-F},s_{MAL-F})]]$; hence, MIMA is prevented in the proposed protocol.

(d) Replay attacks: Let us say A_{NI}^{RR} is an old authentication request of Node_I .The scheme design causes the request to be refused if replayed later as the signature version is maintained. Let Fog_{Cen} gets the replayed request A_{NI}^{RR} . Fog_{Cen} checks to see if $V_{SP_F}^{K} \le LSD_I$. If true, the request is rejected else, it is accepted.

(e) Spoofing attack: The resistance against spoofing attacks is provided using ECDSA verification. The ECDSA [25] signature pair $[(r,s)]$ pair sent along with request and response authentication messages between the new node and the neighboring nodes ensures the integrity authentication of the messages exchanged. In the proposed protocol, Node_I broadcasts the authentication request A_{NI}^{RR} to become part of the network. Any neighbor node who receives the message, before processing further to determine the legitimacy of the node and subsequently to form the shared key, verifies the authenticity and the integrity of the received broadcast using

$$\text{ECDSA} - \text{VERIFY}\big(A_{NI}^{RR}, [(r_i,s_i)], G_K^{Pu}(x,y), E(a,b)\big]. \tag{52}$$

Any spoofing or modification of the broadcast A_{NI}^{RR} would be detected by the neighboring nodes which in turn would result in the rejection of the broadcast before any further processing is done. Thus, the use of ECDSA signature to ensure the integrity and the authenticity of the messages exchanged in the proposed protocol provides a strong resilience against spoofing attacks.

4.2. Security Proof

Theorem 1. *The design of the proposed scheme is resilient to impersonation attack malicious node deployment, man in the middle attack, and spoofing attack: under the ECDLP assumption.*

Proof. The proof is based on [27–29]. Let us define the following oracles for the adversary α:

(i) Reveal $- G_K^{\text{Pr}}$: outputs the G_K^{Pr} using $E(a,b)$ and $G_K^{Pu}(x,y)$ as input.

(ii) Reveal $- F_K^{\text{Pr}}$: outputs the F_K^{Pr} using $E(a,b)$ and $F_K^{Pu}(x,y)$ as input.

(iii) Reveal $- N_K^{\text{Pr}}$: outputs the N_K^{Pr} using $E(a,b)$ and $N_K^{Pu}(x,y)$ as input.

(iv) *Create* $- SP_{MAL-I}(x,y)\&(r_{MAL-I},s_{MAL-I})$: generate the $SP_{MAL-I}(x,y)\&(r_{MAL-I},s_{MAL-I})$ for Node_I

(v) *Create* $- SP_{MAL-F}(x,y)\&(r_{MAL-F},s_{MAL-F})$: generate the $SP_{MAL-F}(x,y)\&(r_{MAL-F},s_{MAL-F})$ for Fog_{Cen}

α runs the experiment $EXP_{E(a,b)}^{MAL}$ as shown in Figure 2. The success of the experiment is defined as follows:

$$\text{Success}_{MAL}^{ECDLP} = 2P\big[EXP_{E(a,b)}^{MAL} = 1\big] - 1. \tag{53}$$

Accordingly, the advantage is defined as follows:

$$\begin{aligned} &ADV_{MAL}^{ECDLP}(t, Q_{CA}, Q_{\text{FOG}}, Q_{\text{NODE}}, Q_{SIG-N}, Q_{SIG-F}) \\ &= \text{Max}_A\big\{\text{Success}_{MAL}^{ECDLP}\big\}. \end{aligned} \tag{54}$$

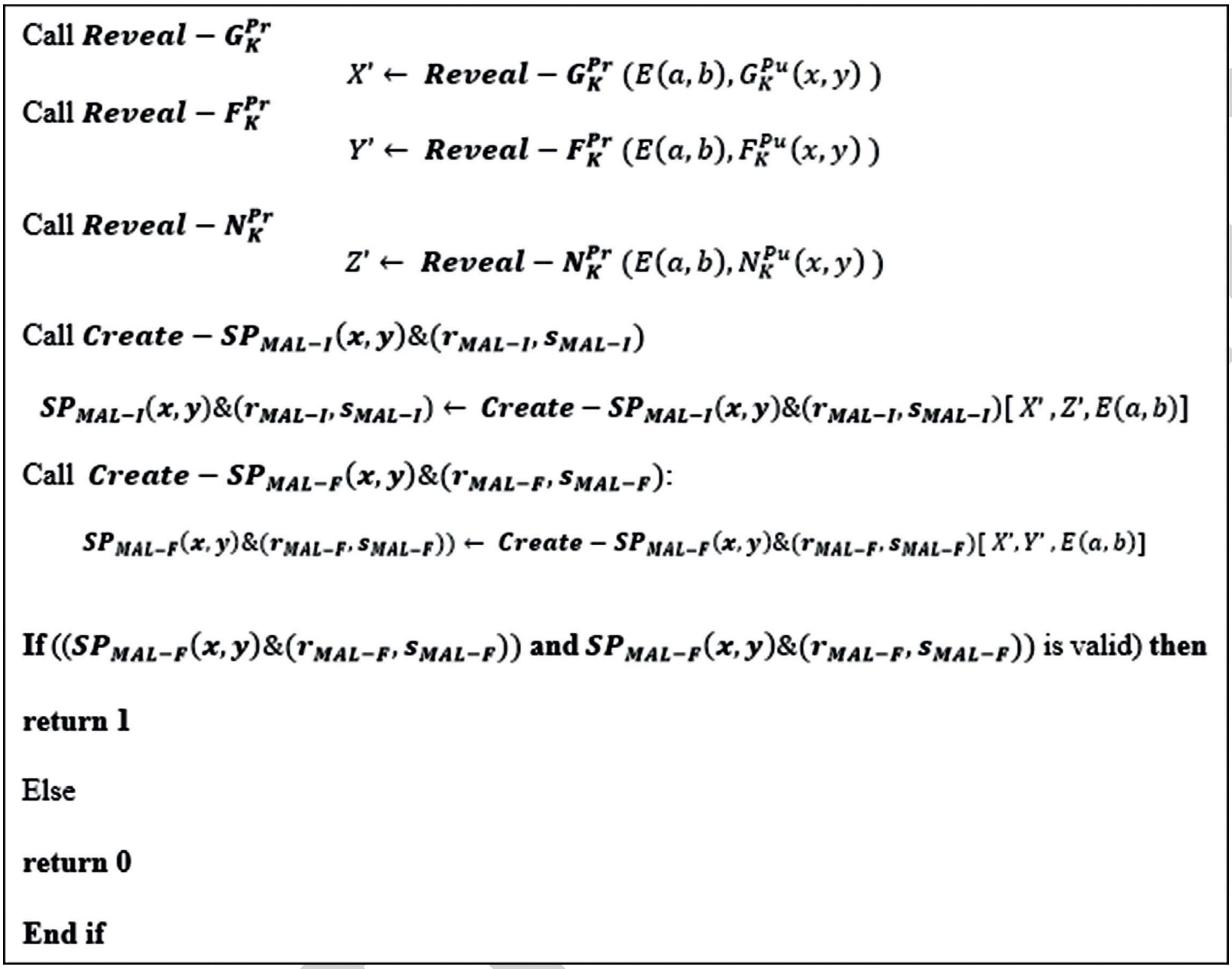
where in maximum is taken over all execution t, Q_{CA} is the number of queries to the Reveal $- G_K^{Pr}$, Q_{FOG} is the number of queries to the Reveal $- F_K^{Pr}$, Q_{NODE} is the number of queries to the Reveal $- N_K^{Pr}$, Q_{SIG-N} is the number of queries to the Create– $SP_{MAL}(x,y)\&(r_{MAL},s_{MAL})$, Q_{SIG-F} is the number of queries to Create $- SP_{MAL-F}(x,y)\&(r_{MAL-F},s_{MAL-F})$. The proposed protocol would be secure against malicious node deployment attacks if:

$$ADV_{MAL}^{ECDLP}(t, Q_{CA}, Q_{\text{FOG}}, Q_{\text{NODE}}, Q_{SIG-N}, Q_{SIG-F}) \le \varepsilon \text{ where } \varepsilon > 0. \tag{55}$$

Based on the experiment shown in Figure 2, α can extract the private key of G_K^{Pr} and N_K^{Pr}. Subsequently, the adversary generates $SP_{MAL-I}(x,y)\&(r_{MAL-I},s_{MAL-I})$ and $SP_{MAL-F}(x,y)\&(r_{MAL-F},s_{MAL-F})$. However, as per the ECDLP definition, extracting $\mathbf{G_K^{Pr}}$ and $\mathbf{N_K^{Pr}}$ is a computationally infeasible problem. Thus, we can conclude the following:

$$ADV_{ACLFS}^{ECDLP}\big(t, Q_{CA}, Q_{\text{FOG}}, Q_{\text{NODE}}, Q_{SIG-N,} Q_{SIG-F}\big) \le \varepsilon \text{ where } \varepsilon > 0. \tag{56}$$

The proposed scheme provides a strong resilience to malicious node deployment. □



```
Call $\boldsymbol{Reveal - G_K^{Pr}}$
        $X' \leftarrow \boldsymbol{Reveal - G_K^{Pr}}\ (E(a,b), G_K^{Pu}(x,y))$
Call $\boldsymbol{Reveal - F_K^{Pr}}$
        $Y' \leftarrow \boldsymbol{Reveal - F_K^{Pr}}\ (E(a,b), F_K^{Pu}(x,y))$

Call $\boldsymbol{Reveal - N_K^{Pr}}$
        $Z' \leftarrow \boldsymbol{Reveal - N_K^{Pr}}\ (E(a,b), N_K^{Pu}(x,y))$

Call $\boldsymbol{Create - SP_{MAL-I}(x,y)\&(r_{MAL-I}, s_{MAL-I})}$

  $\boldsymbol{SP_{MAL-I}(x,y)\&(r_{MAL-I}, s_{MAL-I}) \leftarrow Create - SP_{MAL-I}(x,y)\&(r_{MAL-I}, s_{MAL-I})[X', Z', E(a,b)]}$

Call $\boldsymbol{Create - SP_{MAL-F}(x,y)\&(r_{MAL-F}, s_{MAL-F})}$:

    $\boldsymbol{SP_{MAL-F}(x,y)\&(r_{MAL-F}, s_{MAL-F})) \leftarrow Create - SP_{MAL-F}(x,y)\&(r_{MAL-F}, s_{MAL-F})[X', Y', E(a,b)]}$


If (($\boldsymbol{SP_{MAL-F}(x,y)\&(r_{MAL-F}, s_{MAL-F})}$) and $\boldsymbol{SP_{MAL-F}(x,y)\&(r_{MAL-F}, s_{MAL-F})}$) is valid) then

return 1

Else

return 0

End if
```

Figure 2: Experiment $EXP_{E(a,b)}^{MAL}$ run by the adversary.

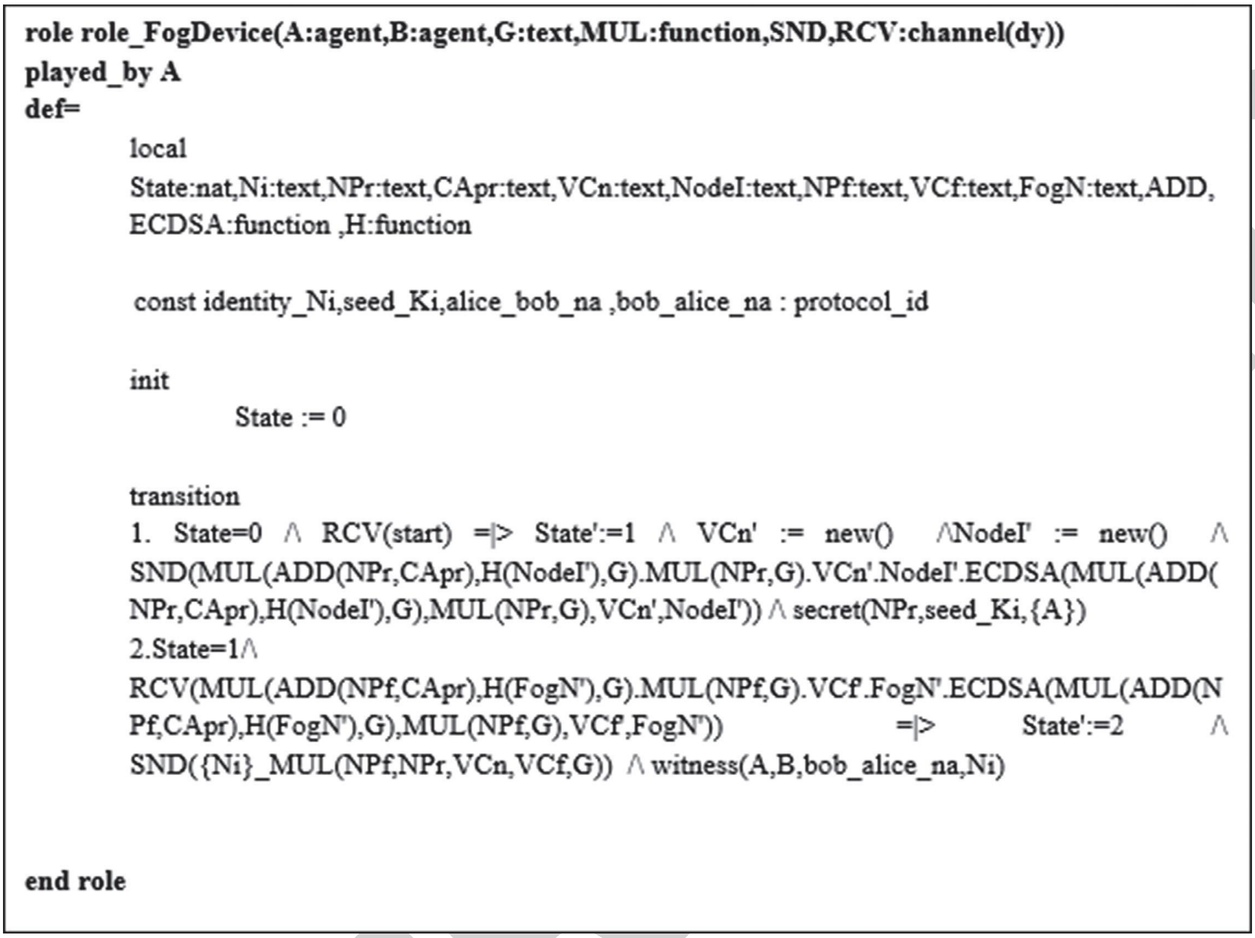
5. AVISPA Simulation

With the help of AVISPA simulation, we prove that the proposed scheme is resistant to man-in-the-middle and replay attacks.

5.1. HLPSL Specification of the Proposed Scheme. In this section, the HLPSL model of the proposed access control scheme is discussed. The authentication and the key exchange between the Node_I and the Fog_{Cen} are modeled by defining their corresponding HLPSL roles. The HLPSL model of the Node_I is given in Figure 3. The *role_FogDevice* is played by agent A. The RCV (start) in state 0 of the *role_FogDevice* initiates the simulation. On receiving the start, agent A sends the $A_{NI}^{RR} = SP_I(x,y)\|N_K^{Pu}(x,y)\| V_{SP_I}^K\|\text{Node}_I\|, [(r_i, s_i)]$ using the SND() operation. SND and RCV are defined as a channel (dy). Channel (dy) defines the Dolev and Yoa threat model in which the communication channel is completely insecure. In-state 0, N_K^{Pr} is specified to be a secrecy goal identified by protocol_id type *seed_Ki*. The *roleNewNode* in state 1, on receiving the response $A_{\text{Fog}}^{RE} = SP_F(x,y)\|F_K^{Pu}(x,y)\|V_{SP_F}^K\|\text{Fog}_{\text{Cen}}\|[(r_f, s_f)]$ using the RCV() from Fog_{Cen} , Node_I sends $E_{KJI}[N1H(N1)]$ and the conjunction, *witness(A,B,bob_alice_na,Ni))* is validated. Witness (A, B, bob_alice_na, Ni) demands a weak authentication of Node_I by Fog_{Cen} , where Fog_{Cen} is witness to the information given by Node_I, i.e., Ni'. Bob_alice_na identifies this property in the goal section defined in the environment role.

The HLPSL model of the Fog_{Cen} is given in Figure 4. The *role_FogCentre* is played by agent B. On receiving $A_{NI}^{RR} = SP_I(x,y)\|N_K^{Pu}(x,y)\|V_{SP_I}^K\|\text{Node}_I\|, [(r_i, s_i)]$ using RCV(), agent B, sends $A_{NI}^{RR} = SP_I(x,y)\|N_K^{Pu}(x,y)\|V_{SP_I}^K\|\text{Node}_I\|, [(r_i, s_i)]$ using SND() operation. The F_K^{Pr} is specified to be a secrecy goal identified by protocol_id type *seed_KJ*. *request*(*B*,*A*,*bob_alice_na*,*Ni*) is a strong authentication where Fog_{Cen} is a witness of the *Ni* for Node_I and is identified by *bob_alice_na* in the goal section. The role *session* and *environment* are shown in Figure 5. A session is a composing role instantiating one or more basic roles. The composed role does not have a transition section. /\ is used to indicate the basic role that runs in parallel. Role A and B are initiated in parallel as shown in Figure 5.

5.2. Simulation Results. The HLPSL code of the proposed protocol was simulated on SPAN, which is the simulation



```
role role_FogDevice(A:agent,B:agent,G:text,MUL:function,SND,RCV:channel(dy))
played_by A
def=
        local
        State:nat,Ni:text,NPr:text,CApr:text,VCn:text,NodeI:text,NPf:text,VCf:text,FogN:text,ADD,
        ECDSA:function ,H:function

         const identity_Ni,seed_Ki,alice_bob_na ,bob_alice_na : protocol_id

        init
                State := 0

        transition
        1. State=0 /\ RCV(start) =|> State':=1 /\ VCn' := new() /\NodeI' := new() /\
        SND(MUL(ADD(NPr,CApr),H(NodeI'),G).MUL(NPr,G).VCn'.NodeI'.ECDSA(MUL(ADD(
        NPr,CApr),H(NodeI'),G),MUL(NPr,G),VCn',NodeI')) /\ secret(NPr,seed_Ki,{A})
        2.State=1/\
        RCV(MUL(ADD(NPf,CApr),H(FogN'),G).MUL(NPf,G).VCf'.FogN'.ECDSA(MUL(ADD(N
        Pf,CApr),H(FogN'),G),MUL(NPf,G),VCf',FogN'))          =|>        State':=2      /\
        SND({Ni}_MUL(NPf,NPr,VCn,VCf,G)) /\ witness(A,B,bob_alice_na,Ni)


end role
```

Figure 3: HLPSL role for new node Node_I.

animator for AVISPA. The corresponding message sequence chart on SPAN depicts 02 messages being exchanged, as shown in Figure 6. The HLPSL model of the proposed protocol has been verified on the OFMC backend. OFMC backend employs symbolic techniques to create on-the-fly state representation. OFMC provides fast detection of attacks in a bounded number of sessions. To verify the replay attack in the proposed scheme, the backend performs a search of a passive intruder. The simulation results on the OFMC backend are shown in Figure 7. Thus, the AVISPA verification of the scheme indicates that the scheme is SAFE. The search time is 0.25 sec and the number of nodes visited is 3 with a depth of 2.

6. BAN Logic Analysis

Node_I and Fog_{Cen} represent the communicating parties, where N_K^{Pr} and F_K^{Pr} denote their private keys, respectively. The BAN notations are given in Table 2 [30], and the BAN postulates are tabulated in Table 3. Synthesis rules are tabulated in Table 4 [31].

6.1. Assumptions. The assumptions are listed below:

(AS1) $\text{Node}_I| \equiv\ \longrightarrow^{N_{KI}^{Pu}}(x, y)\ \text{Node}_I$

(AS2) $\text{Fog}_{\text{Cen}}| \equiv\ \longrightarrow^{N_{KI}^{Pu}}(x, y)\ \text{Node}_I$

(AS3) $\text{Fog}_{\text{Cen}}| \equiv\ \longrightarrow^{F_K^{Pu}}(x, y)\ \text{Fog}_{\text{Cen}}$

(AS4) $\text{Node}_I| \equiv\ \longrightarrow^{F_K^{Pu}}(x, y)\ \text{Fog}_{\text{Cen}}$

(AS5) $\text{Node}_I| \equiv \#(V_{SP_I}^K)$

(AS6) $\text{Fog}_{\text{Cen}}| \equiv \#(V_{SP_F}^K)$

6.2. Idealized Form

$$\begin{aligned}&\text{Node}_I \longrightarrow \text{Fog}_{\text{Cen}}; \{SP_I(x, y)\}_{\mathbf{N_K^{Pr}}},\\ &\text{Fog}_{\text{Cen}} \longrightarrow \text{Node}_I; \{SP_F(x, y)\}_{\mathbf{F_K^{Pr}}}.\end{aligned} \tag{57}$$

6.3. Goals. (G1) $\text{Fog}_{\text{Cen}}| \equiv \text{Node}_I \longrightarrow^K IJ\ \text{Fog}_{\text{Cen}}$.

(G2) $\text{Fog}_{\text{Cen}}| \equiv \text{Node}_I| \equiv \text{Node}_I \longrightarrow^K IJ\ \text{Fog}_{\text{Cen}}$

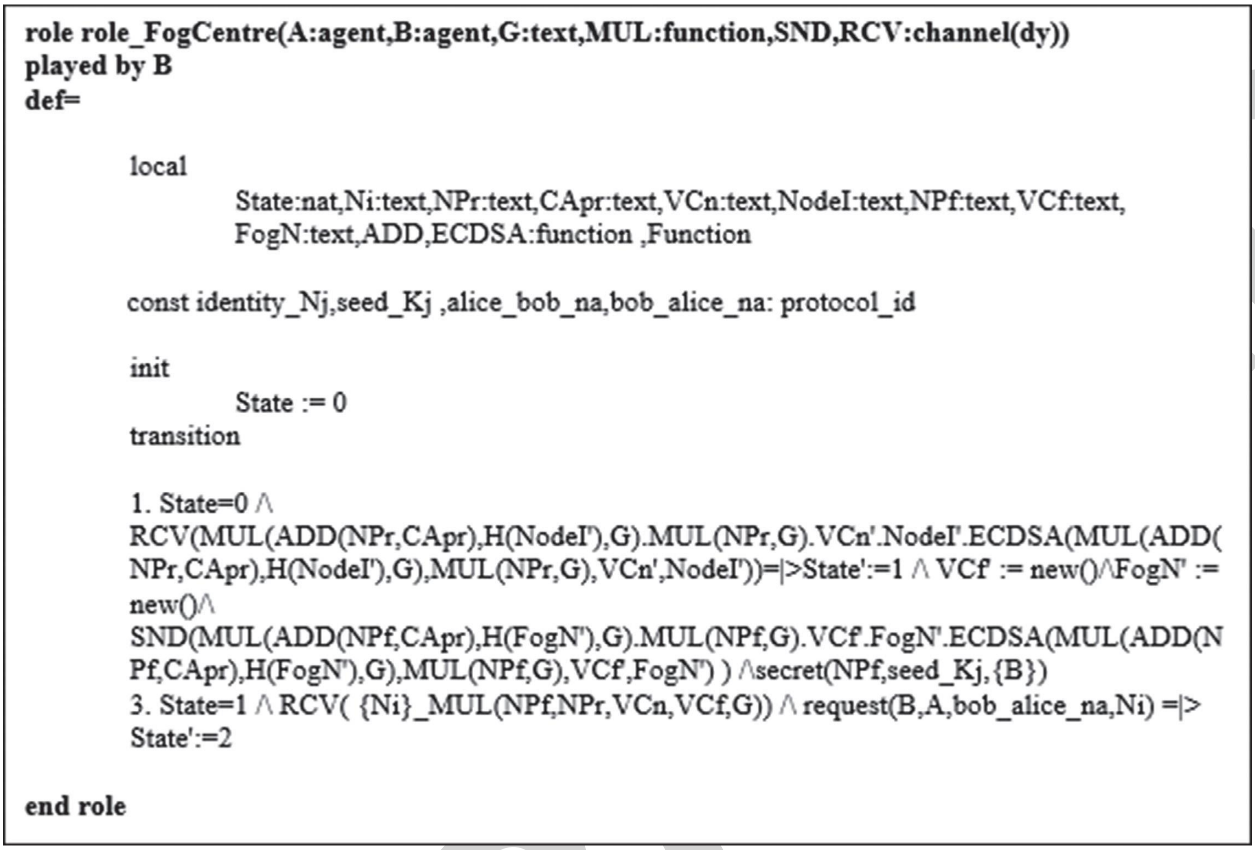
(G3) $\text{Node}_I| \equiv \text{Node}_I \longrightarrow^K IJ\ \text{Fog}_{\text{Cen}}$.

(G4) $\text{Node}_I| \equiv \text{Fog}_{\text{Cen}}| \equiv \text{Node}_I \longrightarrow^K IJ\ \text{Fog}_{\text{Cen}}$

6.4. BAN Verification of the Proposed Protocol. From (M1), we infer the following:

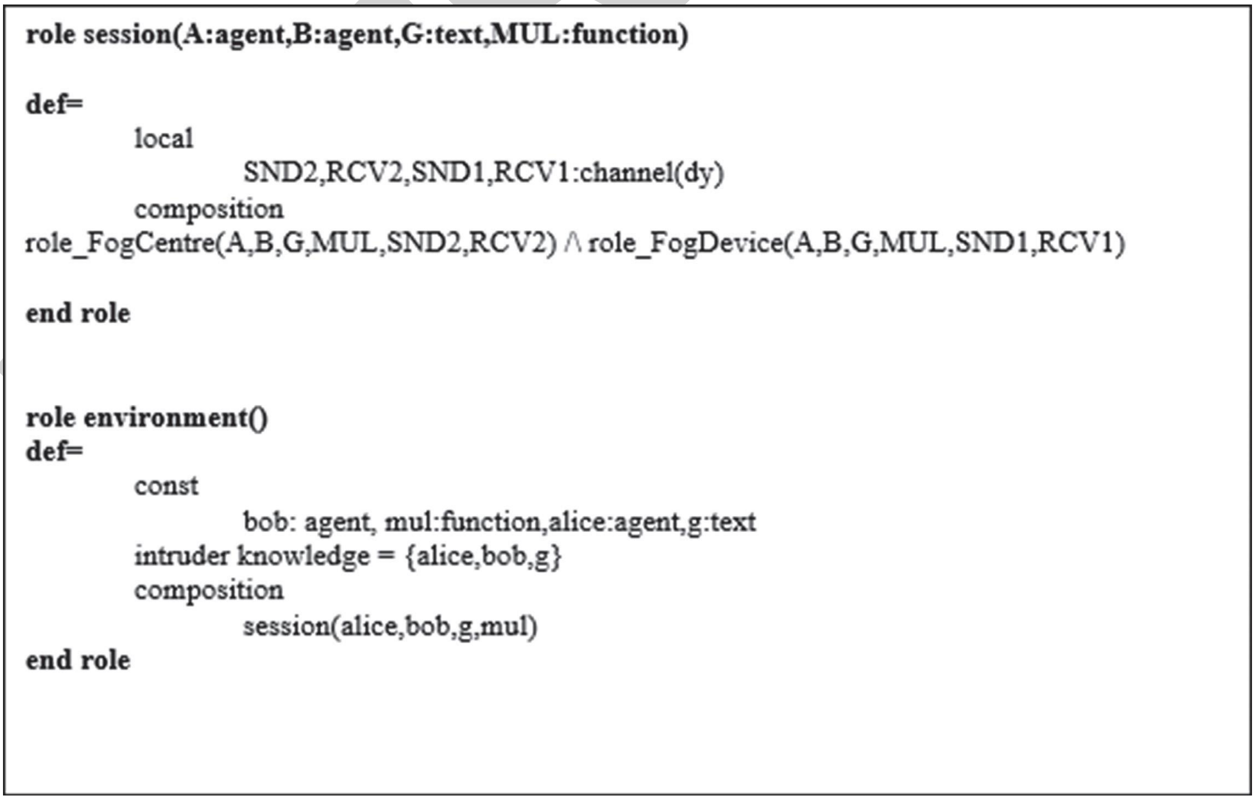
(1) $\text{Node}_I| \equiv \{SP_I(x, y)\}_{\mathbf{N_K^{Pr}}}$

(2) $\text{Fog}_{\text{Cen}} \Leftarrow \{SP_I(x, y)\}_{\mathbf{N_K^{Pr}}}$



```
role role_FogCentre(A:agent,B:agent,G:text,MUL:function,SND,RCV:channel(dy))
played by B
def=

        local
                State:nat,Ni:text,NPr:text,CApr:text,VCn:text,NodeI:text,NPf:text,VCf:text,
                FogN:text,ADD,ECDSA:function ,Function

        const identity_Nj,seed_Kj ,alice_bob_na,bob_alice_na: protocol_id

        init
                State := 0
        transition

        1. State=0 /\
        RCV(MUL(ADD(NPr,CApr),H(NodeI'),G).MUL(NPr,G).VCn'.NodeI'.ECDSA(MUL(ADD(
        NPr,CApr),H(NodeI'),G),MUL(NPr,G),VCn',NodeI'))=|>State':=1 /\ VCf' := new()/\FogN' :=
        new()/\
        SND(MUL(ADD(NPf,CApr),H(FogN'),G).MUL(NPf,G).VCf'.FogN'.ECDSA(MUL(ADD(N
        Pf,CApr),H(FogN'),G),MUL(NPf,G),VCf',FogN') ) /\secret(NPf,seed_Kj,{B})
        3. State=1 /\ RCV( {Ni}_MUL(NPf,NPr,VCn,VCf,G)) /\ request(B,A,bob_alice_na,Ni) =|>
        State':=2

end role
```

Figure 4: HLPSL role for Fog_{Cen}.



```
role session(A:agent,B:agent,G:text,MUL:function)

def=
        local
                SND2,RCV2,SND1,RCV1:channel(dy)
        composition
role_FogCentre(A,B,G,MUL,SND2,RCV2) /\ role_FogDevice(A,B,G,MUL,SND1,RCV1)

end role


role environment()
def=
        const
                bob: agent, mul:function,alice:agent,g:text
        intruder knowledge = {alice,bob,g}
        composition
                session(alice,bob,g,mul)
end role
```

Figure 5: HLPSL role for session and environment.

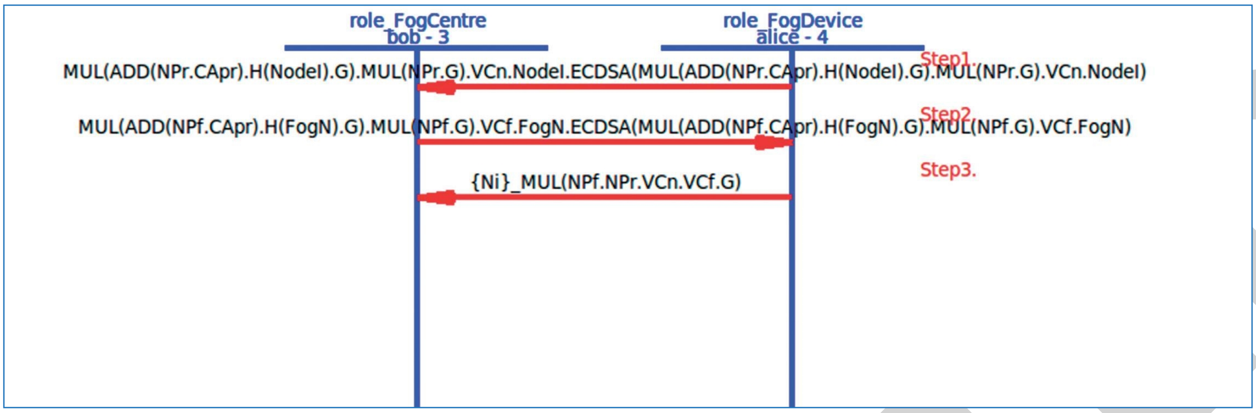
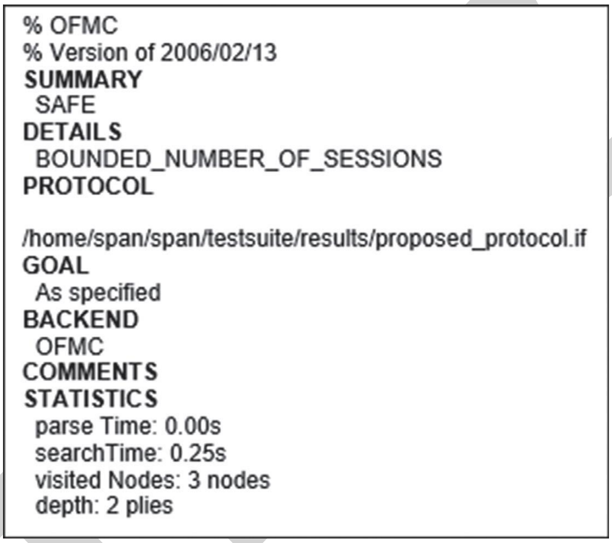

role_FogCentre
bob - 3
role_FogDevice
alice - 4
MUL(ADD(NPr.CApr).H(NodeI).G).MUL(NPr.G).VCn.NodeI.ECDSA(MUL(ADD(NPr.CApr).H(NodeI).G).MUL(NPr.G).VCn.NodeI)
Step1.
MUL(ADD(NPf.CApr).H(FogN).G).MUL(NPf.G).VCf.FogN.ECDSA(MUL(ADD(NPf.CApr).H(FogN).G).MUL(NPf.G).VCf.FogN)
Step2.
{Ni}_MUL(NPf.NPr.VCn.VCf.G)
Step3.

Figure 6: Message sequence chart.



```
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL

/home/span/span/testsuite/results/proposed_protocol.if
GOAL
  As specified
BACKEND
  OFMC
COMMENTS
STATISTICS
  parse Time: 0.00s
  searchTime: 0.25s
  visited Nodes: 3 nodes
  depth: 2 plies
```

Figure 7: AVISPA verification results on OFMC backend.

Table 2: BAN notations.

Notation	Description
$\text{Node}_I \mid\equiv \text{Message}$	Node_I believes Message
$\text{Node}_I \Leftarrow \text{Message}$	Node_I receives Message
$\text{Node}_I \mid\sim \text{Message}$	Node_I sent the Message in past.
$\text{Node}_I \sim \text{Message}$	Node_I sent the Message currently.
$\text{Node}_I \mid\longrightarrow V$	Node_I has jurisdiction over V
$\#(M)$	M is fresh
$\longrightarrow^{N_{KI}^{Pu}}(x, y)\ \text{Node}_I$	$N_K^{Pu}(x, y)$ is the public key of Node_I
$\text{Node}_I \longrightarrow^{K_{JI}} \text{Fog}_{\text{Cen}}$	K_{JI} is the shared key between Node_I and Fog_{Cen}
$\{X\}_{K_{JI}}$	K_{JI} is the key used to encrypt X.
$(EXP1/EXP2)$	If $EXP1$ is true, then $EXP2$ is true

From (2), (AS2) and (R1), we obtain as below:

(3) $\text{Fog}_{\text{Cen}} \mid\equiv \text{Node}_I \mid\sim SP_I(x, y)$

$V_{SP_I}^K$ is a part of $SP_I(x, y)$; from (AS5) and (R6), we obtain as below:

(4) $\text{Node}_I \mid\equiv \#(SP_I(x, y))$

From 3 and 4, we obtain as below:

(5) $\text{Fog}_{\text{Cen}} \mid\equiv \text{Node}_I \sim SP_I(x, y)$

From (5) and (SR4), we obtain as below:

(6) $\text{Fog}_{\text{Cen}} \mid\equiv \#(SP_I(x, y))$

From (3), (6), and (R2), we obtain as below:

(7) $\text{Fog}_{\text{Cen}} \mid\equiv \text{Node}_I \mid\equiv SP_I(x, y)$

$V_{SP_I}^K$ is a part of $SP_I(x, y)$; from (R5), we obtain as below:

(8) $\text{Fog}_{\text{Cen}} \mid\equiv \text{Node}_I \mid\equiv V_{SP_I}^K$

Table 3: Basic postulates.

Rule No	Rule	Representation
R1	Message meaning	$(\text{Node}_I \mid\equiv\ \xrightarrow{F_K^{Pu}} (x, y)\ \text{Fog}_{\text{Cen}}, \text{Node}_I \Leftarrow \{X\}_{\mathbf{F}_{\mathbf{K}}^{\mathbf{Pr}}} / \text{Node}_I \mid\equiv \text{Fog}_{\text{Cen}} \mid\sim X)$
R2	Nonce verification	$(\text{Node}_I \mid\equiv \#(X), \text{Node}_I \mid\equiv \text{Fog}_{\text{Cen}} \mid\sim X / \text{Node}_I \mid\equiv \text{Fog}_{\text{Cen}} \mid\equiv X)$
R3	Jurisdiction	$(\text{Node}_I \mid \longrightarrow X, \text{Node}_I \mid\equiv \text{Fog}_{\text{Cen}} \mid\equiv X / \text{Node}_I \mid\equiv X)$
R4	Seeing	$(\text{Node}_I \Leftarrow X, \text{Node}_I \Leftarrow Y / \text{Node}_I \Leftarrow (X, Y))$
R5	Belief	$(\text{Node}_I \mid\equiv X, \text{Node}_I \mid\equiv Y / \text{Node}_I \mid\equiv (X, Y))$
R6	Freshness	$(\text{Node}_I \mid\equiv \#(X) / \text{Node}_I \mid\equiv \#(X, Y))$
R7	Session key	$(\text{Node}_I \mid\equiv \#(SK), \text{Node}_I \mid\equiv \text{Fog}_{\text{Cen}} \mid\equiv X / \text{Node}_I \mid\equiv \text{Node}_I \longrightarrow^{S} K\ \text{Fog}_{\text{Cen}})$

Table 4: Synthesis rules.

Rule No	Synthesis rule
S1	$\text{Node}_I \Leftarrow A\ \text{Node}_I \Leftarrow (A, B)$
S2	$\text{Node}_I \mid\equiv \text{Fog}_{\text{Cen}} \mid\sim\ A\ \text{Node}_I \mid\equiv \text{Fog}_{\text{Cen}} \mid\sim\ (A, B)$
S3	$\text{Node}_I \mid\equiv \text{Fog}_{\text{Cen}} \mid\sim\ (A, B)\ \text{Node}_I \mid\equiv \text{Fog}_{\text{Cen}} \mid\sim A$
S4	$\text{Node}_I \mid\equiv \text{Fog}_{\text{Cen}} \sim A\ P \mid\equiv \#(A)$

From (SR3) and (3), we obtain as below:

(9) $\text{Fog}_{\text{Cen}} \mid\equiv \text{Node}_I \mid\sim V_{SP_I}^{K}$

From (AS5) and (9), we obtain as below:

(10) $\text{Fog}_{\text{Cen}} \mid\equiv \text{Node}_I \sim V_{SP_I}^{K}$

From (RS4) and (10), we obtain as below:

(11) $\text{Fog}_{\text{Cen}} \mid\equiv \#(V_{SP_I}^{K})$

$V_{SP_I}^{K}$ is a part of K_{IJ}; from (R6), we obtain as below:

(12) $\text{Fog}_{\text{Cen}} \mid\equiv \#(K_{IJ})$

From (10), (12), and (R7), we obtain as below:

(13) $\text{Fog}_{\text{Cen}} \mid\equiv \text{Node}_I \longrightarrow^{K} IJ\ \text{Fog}_{\text{Cen}}$

Due to the symmetry of the protocol,

(14) $\text{Node}_J \mid\equiv \text{Fog}_{\text{Cen}} \mid\equiv \text{Node}_I \longrightarrow^{K} IJ\ \text{Fog}_{\text{Cen}}$

From (M2), we infer that

(15) $\text{Fog}_{\text{Cen}} \mid\equiv \{SP_F(x, y)\}_{\mathbf{F}_{\mathbf{K}}^{\mathbf{Pr}}}$

(16) $\text{Node}_I \Leftarrow \{SP_J(x, y)\}_{\mathbf{N}_{\mathbf{KJ}}^{\mathbf{Pr}}}$

From (16), (AS2), and (R1), we obtain as below:

(17) $\text{Node}_I \mid\equiv \text{Fog}_{\text{Cen}} \mid\sim SP_F(x, y)$

$V_{SP_F}^{K}$ is a part of $SP_F(x, y)$; from (AS5) and (R6), we obtain as below:

(18) $\text{Fog}_{\text{Cen}} \mid\equiv \#(SP_F(x, y))$

From 17 and 18, we obtain as below:

(19) $\text{Node}_I \mid\equiv \text{Fog}_{\text{Cen}} \sim SP_F(x, y)$

From (19) and (SR4), we obtain as below:

(20) $\text{Node}_I \mid\equiv \#(SP_F(x, y))$

From (17), (20), and (R2), we obtain as below:

(21) $\text{Node}_I \mid\equiv \text{Fog}_{\text{Cen}} \mid\equiv SP_F(x, y)$

$V_{SP_F}^{K}$ is a part of $SP_F(x, y)$; from (R5), we obtain as below:

(22) $\text{Node}_I \mid\equiv \text{Fog}_{\text{Cen}} \mid\equiv V_{SP_F}^{K}$

From (SR3) and (17), we obtain as below:

(23) $\text{Node}_I \mid\equiv \text{Fog}_{\text{Cen}} \mid\sim V_{SP_F}^{K}$

From (AS5) and (23), we obtain as below:

(24) $\text{Node}_I \mid\equiv \text{Fog}_{\text{Cen}} \sim V_{SP_F}^{K}$

From (RS4) and (24), we obtain as below:

(25) $\text{Node}_I \mid\equiv \#(V_{SP_F}^{K})$

$V_{SP_F}^{K}$ is a part of K_{IJ};from (R6), we obtain as below:

(26) $\text{Node}_I \mid\equiv \#(K_{IJ})$

From (25), (26), and (R7), we obtain as below:

(27) $\text{Node}_I \mid\equiv \text{Node}_I \longrightarrow^{K} IJ\ \text{Fog}_{\text{Cen}}$

Due to the symmetry of the protocol,

(28) $\text{Node}_I \mid\equiv \text{Fog}_{\text{Cen}} \mid\equiv \text{Node}_I \longrightarrow^{K} IJ\ \text{Fog}_{\text{Cen}}$

7. Comparison with Other Schemes

To draw a comparison of the computational cost between the LKSE and the proposed scheme, the various computational operations considered include Hash Operation : OP_{HASH} , ECC Point Addition ($OP_{ECC-A\ DD}$), ECC Scalar multiplication ($OP_{ECC-MUL}$) , Public key Encryption (OP_{PK-ENC}), Public Key Decryption ($OP_{PK-DE\ C}$), Symmetric key Encryption (OP_{SK-ENC}), Symmetric Key Decryption ($OP_{SK-DE\ C}$) , ECDSA-Verification: ($OP_{EC\ DS\ A-VER}$) , and Modular Inverse (OP_{INV}) . The comparison of the computational cost in terms of computational operation is shown in Table 5. The total no of operations for the proposed scheme is: $8OP_{\text{HASH}} + 2OP_{ECC-MUL} + 2OP_{PK-ENC} + 2OP_{PK-DE\ C}$ and LKSE is $8OP_{\text{HASH}} + 2OP_{ECC-A\ DD} + 4OP_{ECC-MUL} + OP_{SK-ENC} + OP_{SK-DE\ C} + 2OP_{EC\ DS\ A} + 2OP_{INV}$. From Table 5, we can infer that as the proposed scheme does include any public-key encryption and decryption; thus, the computational cost of the proposed scheme is less than LKSE. The size of each message exchanged is shown in Table 6. The total communication cost in the proposed scheme is 2144 bits. The energy consumed for the computational operators on the MicaZ [32] node is depicted in Table 7 [33]. The time taken for public-key encryption and decryption on MicaZ is 0.79 s and 21.5 s [34]. Thus, on a MicaZ mote, the required energy for public key encryption and decryption is 18.96 mJ and 516 mJ respectively. The communication overhead comparison is shown in Figure 8. The energy overhead comparison is shown in Figure 9. The highest communication and energy overhead is that of CE-SKE with 3072 bits and 1606.56 mJ. The high energy overhead in CE-SKE and LKSE scheme is owing to the use of OP_{PK-ENC} and $OP_{PK-DE\ C}$. From Figures 8 and 9,

Table 5: Computational overhead.

Scheme		OP_{HASH}	$OP_{ECC-A\ DD}$	$OP_{ECC-MUL}$	OP_{PK-ENC}	$OP_{PK-DE\ C}$	OP_{SK-ENC}	$OP_{SK-DE\ C}$	$OP_{EC\ DS\ A}$	OP_{INV}
CE-SKE	Node_I	4	0	0	1	2	0	0	0	0
	Fog_{Cen}	4	0	0	2	1	0	0	0	0
LKSE	Node_I	4	0	1	1	1	0	0	0	0
	Fog_{Cen}	4	0	1	1	1	0	0	0	0
Proposed	Node_I	2	1	2	0	0	1	0	1	1
	Fog_{Cen}	2	1	2	0	0	0	1	1	1

Table 6: Communication overhead in the proposed scheme.

S.No	Message	Size
1	$A_{NI}^{RR} = SP_I(x, y)\|N_K^{Pu}(x, y)\|V_{SP_I}^K\|\text{Node}_I\|, [((r_i, s_i)]$	1008
2	$A_{\text{Fog}}^{RE} = SP_F(x, y)\| F_K^{Pu}(x, y)\|V_{SP_F}^K\|\text{Fog}_{\text{Cen}}\|[(r_f, s_f)]$	1008
3	$E_{KJI}[N1H(N1)$	128
	Total	2144
	$SP_I(x, y) = 320\text{ bits}, SP_F(x, y) = 320\text{ bits} \mid\mid N_K^{Pu}(x, y) = 160\text{bits}, F_K^{Pu}(x, y) = 160\text{bits}[(r_i, s_i)] = 320\text{ bits } [(r_f, s_f)]$ $= 320\text{ bits}, V_{SP_I}^K = 32\text{ bits}, \text{ Node}_I = 16\text{ bits}, V_{SP_F}^K = 32\text{ bits}\text{Fog}_{\text{Cen}} = 16\text{ bits}, E_{KJI} = 128\text{ bits}$	

Table 7: Energy consumed.

Symbol	Energy(mJ)
OP_{HASH}	0.21
$OP_{ECC-A\ DD}$	3.84
$OP_{ECC-MUL}$	67.68
OP_{PK-ENC}	18.96
$OP_{PK-DE\ C}$	516
OP_{SK-ENC}	0.00069
OP_{PK-ENC}	0.00069
$OP_{EC\ DS\ A}$	67.68
OP_{INV}	3.36

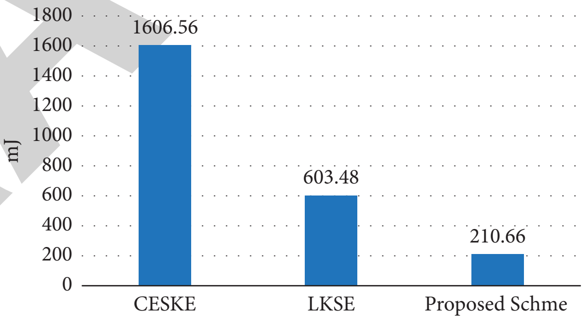


Figure 9: Energy overhead comparison.

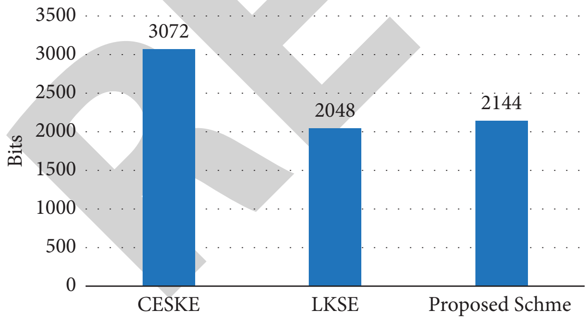


Figure 8: Communication overhead comparison.

Table 8: Security comparison.

Security attack	LSKE	Proposed scheme
Replay attack	✓	✓
Man-in-the-middle attack	X	✓
Insider attack	✓	✓
Impersonation attack	X	✓
Brute force attack	✓	✓
Offline password guessing attack	✓	✓
Mutual authentication	X	✓
Key exchange	✓	✓
Fog federation	✓	✓
Message integrity	X	✓
AVISPA verification	✓	✓

it can be inferred that the proposed protocol has low communication and computational overheads as compared to the CE-SKE and LKSE schemes.

The security comparison is shown in Table 8. The cryptanalysis of LKSE indicates that an adversary can spoof the message exchange and as such can execute a man in the middle attack. The genesis of this attack originates from the fact that there is no complete integrity check on the messages being exchanged as such an adversary was able to manipulate and spoof the messages. As a result of this design flaw, LKSE is not resilient to a man-in-the-middle attack, impersonation attack, and does not support mutual authentication and message integrity. In the proposed protocol, it is computationally impossible for an adversary α to fake $SP_I(x, y), [(r_i, s_i)]$ and $SP_J(x, y), [(r_j, s_j)]$; hence, MIMA and impersonation attack are prevented in the proposed protocol. The design of the proposed scheme also achieves mutual authentication and message integrity using $SP_I(x, y), [(r_i, s_i)]$ and $SP_J(x, y), [(r_j, s_j)]$.

Thus, with the analysis presented, it can be inferred that the proposed scheme with the energy overhead of 210.66 mJ and communication overhead of 2144 bits conforms to all security specifications.

8. Conclusion

The security of sensed data sent from end fog nodes to the fog center is critical and an active area of research. A secure authenticated key exchange between the fog nodes and the fog center is an essential security requirement. Recently, the LKSE scheme for secure key exchange in fog federations was presented. In this paper, a brief review and cryptanalysis of LKSE has been presented. The cryptanalysis indicates that an active adversary can carry out spoofing of the messages, thus resulting in a man in the middle attack. In this paper, a lightweight ECC-based key exchange mechanism for fog federation has been presented. A detailed informal and formal security analysis of the proposed scheme indicates that the scheme is safe from various attacks. The overhead analysis depicts that the proposed scheme requires an energy overhead of 210.66 mJ and communication overhead of 2144 bits while conforming to the desired security specifications.

Data Availability

The data used to support the findings of this study are included in the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] U. Iqbal and A. H. Mir, “Secure and scalable access control protocol for IoT environment,” *Internet of Things*, vol. 12, Article ID 100291, 2020.

[2] P. Ratta, A. Kaur, S. Sharma, M. Shabaz, and G. Dhiman, “Application of Blockchain and internet of things in healthcare and medical sector: applications, challenges, and future perspectives,” *Journal of Food Quality*, vol. 2021, Article ID 7608296, 20 pages, 2021.

[3] F. Ajaz, M. Naseem, S. Sharma, M. Shabaz, and G. Dhiman, “COVID-19: challenges and its technological solutions using IoT,” *Current Medical Imaging Formerly: Current Medical Imaging Reviews*, vol. 17, 2021.

[4] A. Tiwari, V. Dhiman, M. A. M. Iesa, H. Alsarhan, A. Mehbodniya, and M. Shabaz, “Patient Behavioral analysis with smart healthcare and IoT,” *Behavioural Neurology*, vol. 2021, Article ID 4028761, 9 pages, 2021.

[5] B. Zheng, Z. Mei, L. Hou, and S. Qiu, “Application of internet of things and edge computing technology in sports tourism services,” *Security and Communication Networks*, vol. 2021, Article ID 9980375, 10 pages, 2021.

[6] M. Iorga, L. Feldman, R. Barton, M. J. Martin, N. Goren, and C. Mahmoudi, *Fog Computing Conceptual Model*Special Publication (NIST SP), National Institute of Standards and Technology, Gaithersburg, MD, 2018.

[7] O. Akrivopoulos, N. Zhu, D. Amaxilatis, C. Tselios, A. Anagnostopoulos, and I. Chatzigiannakis, “A Fog Computing-Oriented, Highly Scalable IoT Framework for Monitoring Public Educational Buildings,” in *Proceedings of the 2018 IEEE International Conference on Communications (ICC)*, pp. 1–6, Kansas, MO, USA, May 2018.

[8] J. Zheng, A. Jamalipour, and J. Zheng, *Wireless Sensor Networks: A Networking Perspective*, IEEE, Piscataway, NJ, USA, 2009.

[9] J. Bhola and S. Soni, “Information theory-based defense mechanism against DDOS attacks for WSAN,” in *Advances In VLSI, Communication, and Signal Processing*, D. Harvey, H. Kar, S. Verma, and V. Bhadauria, Eds., vol. 683, Singapore, Springer, 2021.

[10] N. Gura, A. Patel, A. Wander, H. Eberle, and S. C. Shantz, *Comparing elliptic curve cryptography and RSA on 8-bit CPUs*, Springer, Berlin, Germany, 2004.

[11] D. J. Malan, M. Welsh, and M. D. Smith, “Implementing public-key infrastructure for sensor networks,” *ACM Transactions on Sensor Networks*, vol. 4, no. 4, pp. 1–23, 2008.

[12] U. Iqbal and S. Shafi, “A Provable and Secure Key Exchange Protocol Based on the Elliptical Curve Diffe–Hellman for WSN,” *Advances in Intelligent Systems and Computing*, Springer Verlag, Berlin, Germany, 2019.

[13] H. Sun, Q. Wen, H. Zhang, and Z. Jin, “A novel remote user authentication and key agreement scheme for mobile client-server environment,” *Applied Mathematics & Information Sciences*, vol. 7, no. 4, pp. 1365–1374, 2013.

[14] X. Jia, D. He, N. Kumar, and K.-K. R. Choo, “Authenticated key agreement scheme for fog-driven IoT healthcare system,” *Wireless Networks*, vol. 25, no. 8, pp. 4737–4750, 2019.

[15] M. Wazid, A. K. Das, N. Kumar, and A. V. Vasilakos, “Design of secure key management and user authentication scheme for fog computing services,” *Future Generation Computer Systems*, vol. 91, pp. 475–492, 2019.

[16] C.-M. Chen, Y. Huang, K.-H. Wang, S. Kumari, and M.-E. Wu, “A secure authenticated and key exchange scheme for fog computing,” *Enterprise Information Systems*, vol. 15, no. 9, pp. 1200–1215, 2020.

[17] Y. Zheng and C.-H. Chang, “Secure mutual authentication and key-exchange protocol between PUF-embedded IoT endpoints,” in *Proceedings of the 2021 IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 1–5, Daegu, South Korea, May 2021.

[18] Y. Chen, J. Yuan, and Y. Zhang, “An improved password-authenticated key exchange protocol for VANET,” *Vehicular Communications*, vol. 27, Article ID 100286, 2021.

[19] Y. Salami, Y. Ebazadeh, and V. Khajehvand, “CE-SKE: LSKE: lightweight secure key exchange scheme in fog federation,” *Complexity*, vol. 2021, no. 3, pp. 1–9, 2021.

[20] Y. Salami and V. Khajehvand, “LSKE: lightweight secure key exchange scheme in fog federation,” *Complexity*, vol. 2021, Article ID 4667586, 9 pages, 2021.

[21] A. Armando, D. Basin, Y. Boichut et al., “The AVISPA tool for the automated validation of internet security protocols and applications,” *Computer Aided Verification*, Springer, Berlin, Germany, 2005.

[22] “The AVISPA project,” 2021, https://www.ercim.eu/publication/Ercim_News/enw64/armando.html.

[23] M. Burrows, M. Abadi, and R. Needham, "A logic of authentication," *ACM Transactions on Computer Systems*, vol. 426, pp. 233–271, 1871.

[24] N. Koblitz, "Elliptic curve cryptosystems," *Mathematics of Computation*, vol. 48, no. 177, pp. 203–209, 1987.

[25] V. S. Miller, "Use of elliptic curves in cryptography," *Lecture Notes in Computer Science Advances in Cryptology — CRYPTO '85 Proceedings*, Springer, Berlin, Germany, 1986.

[26] D. Hankerson and A. Menezes, "Elliptic Curve Cryptography," *Encyclopedia of Cryptography and Security*, p. 397, 2011.

[27] S. Chatterjee and S. Roy, "An efficient dynamic access control scheme for distributed wireless sensor networks," *International Journal of Ad Hoc and Ubiquitous Computing*, vol. 27, no. 1, p. 1, 2018.

[28] Y. H. Chuang and Y. M. Tseng, "An efficient dynamic group key agreement protocol for imbalanced wireless networks," *International Journal of Network Management*, vol. 20, 2010.

[29] A. K. Das, S. Chatterjee, and J. K. Sing, "A novel efficient access control scheme for large-scale distributed wireless sensor networks," *International Journal of Foundations of Computer Science*, vol. 24, no. 5, pp. 625–653, 2013.

[30] S. H. Islam and G. P. Biswas, "A pairing-free identity-based two-party authenticated key agreement protocol for secure and efficient communication," *Journal of King Saud University—Computer and Information Sciences*, vol. 29, no. 1, pp. 63–73, 2017.

[31] L. Buttyan, S. Staamann, and U. Wilhelm, "A simple logic for authentication protocol design," in *Proceedings of the 11th IEEE Computer Security Foundations Workshop*, pp. 153–162, Rockport, MA, USA, June 1998.

[32] "Mote Works.Getting Started Guide, ," MEMSIC, Inc, Milpitas, CA, USA, PN: 7430-0102-02, 2013.

[33] U. Iqbal and A. Hussain Mir, "Secure and Practical Access Control Mechanism for WSN with Node Privacy," *Journal of King Saud University—Computer and Information Sciences*, 2020.

[34] H. Wang and Q. Li, "Efficient implementation of public key cryptosystems on mote sensors (short paper)," *Information and Communications Security*, vol. 4307, pp. 519–528, 2006.