Blockchain-Based Lightweight Authentication Protocol for IoT-Enabled Smart Agriculture

Anusha Vangala
International Institute of Information Technology, Hyderabad 500 032, India
anusha.vangala@research.iiit.ac.in

Sandip Roy
Old Dominion University
Suffolk, VA 23435, USA
sroy@odu.edu

Ashok Kumar Das
International Institute of Information Technology
Hyderabad 500 032, India
iitkgp.akdas@gmail.com, ashok.das@iiit.ac.in

2022 International Conference on Cyber-Physical Social Intelligence (ICCSI) | 978-1-6654-9835-7/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICCSI55536.2022.9970603

***Abstract*—Blockchain technology has a significant application in smart farming due to its immutability, decentralization and transparency properties. Data exchanged in an Internet of Things (IoT)-based smart agriculture can be used to remotely monitor the fields and regulate the crop needs for optimal productivity. However, such data is sensitive to several attacks, such as man-in-the-middle attack, replay attack, ephemeral secret leakage attack, impersonation attack and denial of service (DoS) attack. The existing solutions to counter these attacks are either costly or lack significant security features. To mitigate these issues, we design a novel lightweight blockchain based authentication scheme based on a fully decentralized and distributed architecture. The designed scheme is subjected to a rigorous security analysis and also a formal security verification using the widely-used Automated Validation of Internet Security Protocols and Applications (AVISPA) tool, and it is shown that the scheme is robust and secure against various passive and active attacks. A detailed comparative analysis shows that the proposed scheme has the low communication cost and significantly lower computation cost while satisfying all the security and functionality features as compared to those for other existing relevant schemes.**

***Index Terms*—Smart agriculture, authentication, blockchain technology, Internet of Things (IoT), security.**

I. Introduction

In the current world, food systems have been globalized between the world economies leading to vast social and economic impacts. In addition, urbanization has also lead to adoption of foreign dietary lifestyles into the local food culture [1]. Such cultural coalescence is possible with complex interdependencies of food imports and exports. These trade agreements are predisposed to geopolitics and lead to uneven food production and supply, resulting in incompetency of the global food system to meet the rising food demands. This is observed in the shortage of global wheat supply following the recent Russia-Ukraine conflict. Countries such as Africa, Saudi Arabia and Japan import more than 50% of their domestic food needs. India imports 90% of sunflower oil from Russia and Ukraine [2], [3]. Indonesian ban on palm oil export has surged its global price [4]. In addition, wheat, maize and rice are the staple cereals in most cultures and require massive production every year. However, most of the production is concentrated in Russia for wheat, China, India, Bangladesh, Indonesia and Vietnam for rice, and the US, China and Brazil for maize [5].

Deficit in quantity or quality of food may lead to serious health concerns or even malnutrition in the general population. Decline in population health can affect the economic growth of a country as their work contribution will lessen. A solution to such a complex food system in this information age is to have a nation-wide IoT based network with a management system to monitor the information on crop growth, food processing and supply. Every state and nation that maintain such a system will have a nation-wide view of their food status. Only the people who work in the capacity of taking decisions regarding national food security will be authorized to access the data. In addition, the communication should be restricted to occur only among devices that are permitted to mutually access their data.

Blockchain technology can be subsumed in smart agriculture in order to maintain delicate data regarding national food security. Certain data such as the quality, the price, the nutritional value, location of farm production and processing unit, fertilizers used during farming, and preservation chemicals, should be available to all stakeholders, while other data such as the quantity of produce, availability of resources for production, and marketing details should be available to only some stakeholders. Thus, a hybrid blockchain is the most relevant type for the smart agriculture usecase.

II. Related Work

The authentication and key agreement (AKE) is one of the important security services [6], [7], [8], [9]. However, the AKE schemes currently in existence for smart farming systems are very limited. Ali *et al.* [10] is the earliest identified AKE scheme in agriculture network using wireless sensors resistant to device capture attack, deployment of malicious devices, and replay attack. Chen *et al.* [11] improves the above scheme by adding anonymity, untraceability, perfect forward secrecy, and protection against privileged insider attack, distributed denial of service (DDoS) attack, user impersonation attack, and ephemeral secret leakage (ESL) attack.

Chae and Cho [12] propose a weak AKE scheme for a greenhouse Peer-to-Peer (P2P) smart farm that can only resist dictionary and brute force attacks. Rangwani *et al.* [13] propose a stronger scheme using Elliptic Curve Cryptography (ECC), but it fails to provide anonymity, untraceability and

dynamic node addition. However, none of the above schemes support blockchain technology.

Wu and Tsai [14] propose a private blockchain based AKE scheme which is highly costly because of bilinear pairings and is susceptible to offline guessing and ESL attacks. The scheme by Arshad *et al.* [15] suffers from incomplete proofs. The AKE scheme by Bothe *et al.* [16] polls secret credentials using an ODiL framework while Alyahya *et al.* [17] uses CoAP framework, where ODiL is an "open and decentralized platform for agriculture services" and CoAP means a constrained application protocol.

Previous surveys in smart agriculture on security using blockchain [18] and security without blockchain technology [19] reveal that insufficient support for required security features. Vangala *et al.*'s scheme [20] allows a user to authenticate the smart IoT sensors in order to access data from those smart IoT sensors via a gateway node. Furthermore, another scheme proposed by Vangala *et al.* [21] allows a smart IoT sensor in a field and drones to be authenticated to send and store sensor data on a private blockchain. However, it has a central failure point in the ground station server and cannot store public data that should be transparently available to all stakeholders. In addition, Vangala *et al.* [22] proposed a scheme which allows the smart IoT devices to send their data to a gateway server on a hop-by-hop basis and be stored on a hybrid blockchain via edge servers. However, if an edge server gets disconnected from the network, all the smart farms associated with it are also disconnected from the network.

The proposed work aims to increase decentralization in the system models by removing any intermediate hops for the sensor data from the smart IoT devices to the blockchain storage. This creates a fully distributed system.

III. System Models

A. Network Model

The network architecture model for smart agriculture consists of farming arenas, which could be natural agricultural lands, artificial glasshouses, hydroponic farming centers, or subsistence farming. Smart IoT devices are embedded into the each of these agrosystems to perceive the conditions of its territory. These data are sent to the nearest gateway server in a Peer-to-Peer (P2P) blockchain server farm. The received sensor data are then added into blocks on the blockchain after consensus with the other gateway servers in the farm. A Trusted Admin (TA) registers the smart IoT devices and the gateway servers.

B. Threat Model

The Dolev-Yao threat model [23] is considered the most appropriate threat model for the network model in Section III-A. In this model, an adversary can capture, re-transit, modify, delete messages in public channel between the sensors and the gateway server apart from inserting fake messages. It may impersonate the gateway server and smart IoT devices. Consideration of the Canetti-Krawczyk model [24] allows the adversary to extract secret credentials by hijacking session state or from memory contents of physically captured devices with power analysis and timing attacks.

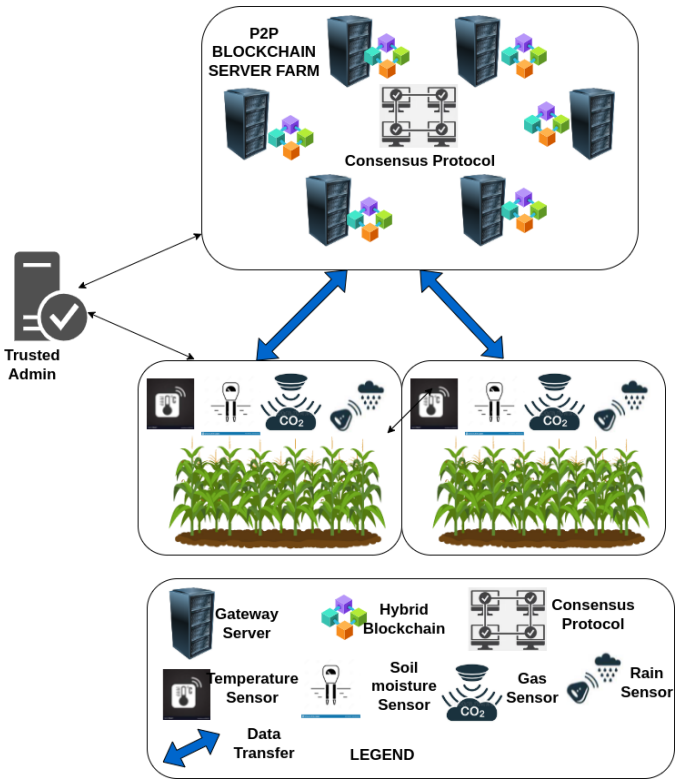

Fig. 1. Lightweight Decentralized Architecture Model for Smart Agriculture

IV. Proposed Scheme

The proposed scheme consists of the following phases.

A. System Initialization

The TA uses this phase to initialize all the parameters required for the working of the proposed system.

- *Step* $SInit_1$*:* A non-singular elliptic curve $E_q(a,b)$: $y^2 = x^3 + ax^2 + b \pmod q$ is chosen over the Galois field $GF(q)$ along with a "point at infinity (zero point)" $\mathcal{O}$ where constants $a, b \in Z_q$ such that $4a^3 + 47b^2 \neq 0$ and q is a large prime. A base point $G \in E_q(a,b)$ is chosen with order n_G as large as q, meaning $n_G \cdot G = G + G + \cdots + G(n_G \text{ times}) = \mathcal{O}$.
- *Step* $SInit_2$*:* A cryptographic one-way hash function $H(\cdot)$ and the "Practical Byzantine Fault Tolerance (PBFT)" algorithm for achieving consensus in the P2P blockchain server farm are chosen.
- *Step* $SInit_3$*:* The TA chooses its secret master key as k_{TA} and publishes the parameters $\{E_q(a,b),\ G,\ H(\cdot)\}$.

B. Registration Phase

1) Registration of IoT Smart Devices: This phase is used by TA to register the IoT smart devices SN in the smart farms.

- *Step* SNR_1*:* The TA picks its true identity ID_S, temporary identity TID_S, device random $n_s \in Z_q^*$ and timestamp t_s. It computes the pseudo-identity $RID_S = H(ID_S || n_S || k_{TA} || t_s)$.

• *Step* SNR_2*:* The TA picks its private key $prv_S \in Z_q^*$ and computes its corresponding public key as $Publ_S = prv_S \cdot G$ and makes it publicly available.
• *Step* SNR_3*:* The TA preloads SN with $\{(RID_S, TID_S, n_S), H(\cdot), E_q(a,b), G, (prv_S, Publ_S)\}$.

Fig. 2 summarizes the steps in the registration of IoT smart device.

Trusted Admin (TA)	IoT Smart Device (SN)
Pick $ID_S, TID_S, n_s \in Z_q^*$ and timestamp t_s. Compute $RID_S = H(ID_S \| n_S \| k_{TA} \| t_s)$. Pick $prv_S \in Z_q^*$, calculate $Publ_S = prv_S \cdot G$. Preload SN with $\{(RID_S, TID_S, n_S), H(\cdot), E_q(a,b), G, (prv_S, Publ_S)\}$.	Store $\{(RID_S, TID_S, n_S), H(\cdot), E_q(a,b), G, (prv_S, Publ_S)\}$ in its memory.

Fig. 2. Summary of IoT Smart Device Registration Phase

2) Offline Registration of Gateway Servers: This phase is used by the TA to register all the gateways servers GS in the blockchain smart farm in an offline mode.
• *Step* GSR_1*:* The TA picks true identity ID_G, temporary identify TID_G, gateway random $n_G \in Z_q^*$ and timestamp t_g. It computes the pseudo-identity $RID_G = H(ID_G \| n_G \| k_{TA} \| t_g)$.
• *Step* GSR_2*:* The GS picks private key $pr_G \in Z_q^*$ and computes its public key as $Publ_G = prv_G \cdot G$.
• *Step* GSR_3*:* The TA preloads the gateway node GS with $\{(RID_G, TID_G), \{(RID_S, TID_S)\}, n_G, H(\cdot), E_q(a,b), G\}$.

C. Authentication Phase

This phase authenticates SN and GS mutually and establishes a session key to be used for secure data aggregation.
• SGA_1*:* SN picks $r_{sg} \in Z_q^*$ and timestamp t_{sg}. It hides the pseudo-identity as $RID_s^* = H(\ r_{sg} \| \ RID_S \| \ prv_S \| \ t_{sg})$. It computes the TA's contribution to IoT smart device for session key as $N_s = H(n_s \| \ RID_S^* \| \ TID_S \| \ prv_S \| \ t_{sg}) \cdot G$ and IoT smart node's contribution to session key as $R_{sg} = H(r_{sg} \| \ RID_S^* \| \ TID_S \| \ prv_S \| \ t_{sg}) \cdot G$.
• SGA_2*:* SN computes the signature on r_{sg} as $Sig_{sg} = H(r_{sg} \| \ RID_S^* \| \ TID_S \| \ prv_S \| \ t_{sg})$ + $H(N_s \| \ R_{sg} \| \ TID_S \| RID_S^* \| t_{sg})$ * $prv_s \pmod q$ and sends the message $Msg_1 = \langle N_S, R_{sg}, Sig_{sg}, TID_S, t_{sg} \rangle$ to GS via public channel.
• SGA_3*:* GS receives Msg_1 at time t'_{sg} and verifies the timestamp as $|t'_{sg} - t_{sg}| \leq \Delta T$. If it is true, the signature is verified as $Sig_{SG} \cdot G \stackrel{?}{=} R_{sg}$ + $H(N_s \| \ R_{sg} \| \ TID_S \| \ RID_S^* \| \ t_{sg}) \cdot Publ_s$. If it is so, GS selects a random $r_{gs} \in Z_q^*$ and timestamp t_{gs}. It hides its pseudo-identity as $RID_g^* = H(r_{gs} \| \ RID_G \| \ prv_G \| \ t_{gs})$. It computes TA's contribution to GS for session key as $N_g = H(\ n_g \| \ RID_G^* \| \ TID_G \| \ prv_G \| \ t_{gs}) \cdot G$ and GS's contribution to session key as $R_{gs} = H(r_{gs} \| \ RID_G^* \| \ TID_G \| \ prv_G \| \ t_{gs}) \cdot G$.
• SGA_4*:* GS computes the Diffie-Hellman parameter as $DK_{GS} = H(\ r_{gs} \| \ RID_G^* \| \ TID_G \| \ prv_G \| \ t_{gs}) \cdot R_{sg}$ and the session secret key as $SK_{GS} = H(DK_{GS} \| \ N_S \| \ H(\ n_g \| \ RID_G^* \| \ TID_G \| \ prv_G \| \ t_{gs}) \| \ R_S \| \ H(\ r_{gs} \| \ RID_G^* \| \ TID_G \| \ prv_G \| \ t_{gs})$. The signature on r_{gs} is computed as $Sig_{gs} = H(r_{gs} \| \ RID_G \| \ prv_G \| \ t_{gs})$ + $H(SK_{GS} \| \ DK_{GS} \| \ TID_S \| \ TID_G \| \ N_S \| \ N_G)$ * $prv_g \pmod q$. GS selects a new temporary identity TID_S^{new} for smart device and hides it as $TID_S^* = TID_s^{new} \oplus H(\ TID_S \| \ N_S \| \ N_G \| \ t_{gs})$. GS sends the message $Msg_2 = \langle N_G, R_{gs}, Sig_{gs}, TID_G, TID_S^*, t_{gs} \rangle$ to SN via public channel.
• SGA_5*:* SN receives Msg_2 at time t'_{gs} and verifies timestamp as $|t'_{gs} - t_{gs}| \leq \Delta T$. If it is true, the Diffie-Hellman parameter is calculated as $DK_{SG} = H(r_{sg} \| \ RID_S^* \| \ TID_S \| \ prv_S \| \ t_{sg}) \cdot R_{gs}$ and the session key is computed as $SK_{SG} = H(DK_{SG} \| \ H(n_s \| \ RID_S^* \| \ TID_S \| \ prv_S \| \ t_{sg}) \| \ N_G \| \ H(r_{sg} \| \ RID_S^* \| \ TID_S \| \ prv_S \| \ t_{sg}) \| \ R_{gs})$.
• SGA_6*:* SN verifies the signature as $Sig_{gs} \cdot G \stackrel{?}{=} R_{gs}$ + $H(SK_{SG} \| \ DK_{SG} \| \ TID_S \| \ TID_G \| \ N_S \| \ N_G) \cdot Publ_g$. If the signature is valid, SN stores the session key SK_{SG} and GS also stores the session key SK_{GS}. SN extracts its new temporary identity as $TID_s^{new} = TID_S^* \oplus H(\ TID_S \| \ N_S \| \ N_G \| \ t_{gs})$ and updates TID_S in its memory.

Fig. 3 summarizes the steps in the authentication phase between the IoT smart device and the gateway server.

IoT Smart Device (SN)	Gateway Server (GS)
Pick $r_{sg} \in Z_q^*$ and timestamp t_{sg}. Compute $RID_s^* = H(r_{sg} \| \ RIDS \| \ prvs \| \ t_{sg})$, $N_s = H(n_s \| \ RID_S^* \| \ TIDS \| \ prvs \| \ t_{sg}) \cdot G$, $R_{sg} = H(r_{sg} \| \ RID_S^* \| \ TIDS \| \ prvs \| \ t_{sg}) \cdot G$, $Sig_{sg} = H(r_{sg} \| \ RID_S^* \| \ TIDS \| \ prvs \| \ t_{sg})$ + $H(N_s \| \ R_{sg} \| \ TIDS \| RIDS^* \| t_{sg})$ * $prv_s \pmod q$. $Msg_1 = \langle N_S, R_{sg}, Sig_{sg}, TIDS, t_{sg} \rangle$ via public channel	Check if $\|t'_{gs} - t_{sg}\| \leq \Delta T$ and $Sig_{SG} \cdot G \stackrel{?}{=} R_{sg} + H(N_s \| \ R_{sg} \| \ TIDS \| \ RID_S^* \| \ t_{sg}) \cdot Publ_s$ Select random $r_{gs} \in Z_q^*$ and timestamp t_{gs}. Compute $RID_g^* = H(r_{gs} \| \ RID_G \| \ prv_G \| \ t_{gs})$, $N_g = H(\ n_g \| \ RID_G^* \| \ TID_G \| \ prv_G \| \ t_{gs}) \cdot G$, $R_{gs} = H(r_{gs} \| \ RID_G^* \| \ TID_G \| \ prv_G \| \ t_{gs}) \cdot G$, $DK_{GS} = H(r_{gs} \| \ RID_G^* \| \ TID_G \| \ prv_G \| \ t_{gs}) \cdot R_{sg}$, $SK_{GS} = H(DK_{GS} \| \ N_S \| \ H(\ n_g \| \ RID_G^* \| \ TID_G \| \ prv_G \| \ t_{gs}) \| \ R_S \| \ H(\ r_{gs} \| \ RID_G^* \| \ TID_G \| \ prv_G \| \ t_{gs})$, $Sig_{gs} = H(r_{gs} \| \ RID_G \| \ prv_G \| \ t_{gs})$ + $H(SK_{GS} \| \ DK_{GS} \| \ TID_S \| \ TID_G \| \ N_S \| \ N_G)$ * $prv_g \pmod q$. Select TID_S^{new} and compute $TID_s^* = TID_s^{new} \oplus H(\ TID_S \| \ N_S \| \ N_G \| \ t_{gs})$. $Msg_2 = \langle N_G, R_{gs}, Sig_{gs}, TIDG, TIDS^*, t_{gs} \rangle$ via public channel Store session key SK_{GS}.
Check if $\|t'_{gs} - t_{sg}\| \leq \Delta T$? Compute $DK_{SG} = H(r_{sg} \| \ RID_S^* \| \ TIDS \| \ prvs \| \ t_{sg}) \cdot R_{gs}$, $SK_{SG} = H(DK_{SG} \| \ H(n_s \| \ RID_S^* \| \ TIDS \| \ prvs \| \ t_{sg}) \| \ N_G \| \ H(r_{sg} \| \ RID_S^* \| \ TIDS \| \ prvs \| \ t_{sg}) \| R_{gs})$. Verify: $Sig_{gs} \cdot G \stackrel{?}{=} R_{GS} + H(SK_{SG} \| \ DK_{SG} \| \ TIDS \| \ TIDG \| \ N_S \| \ N_G) \cdot Publ_g$ If true, store session key SK_{SG} and compute $TID_s^{new} = TIDS \oplus H(\ TIDS \| \ N_S \| \ N_G \| \ t_{gs})$. Update TID_S in database.	
Both SN and GS share the same secret key $SK_{SG} = SK_{GS}$.	

Fig. 3. Summary of Authentication Phase

D. Secure Data Aggregation

The smart IoT devices use this phase to send their data to their registered gateway server. The readings ($Readings$) from the environmental surroundings are collected by the smart IoT sensor devices. These $Readings$ are then encrypted using the session key SK_{SG} established in the authentication phase in Section IV-C and sent to the gateway server GS securely. GS decrypts the $Readings$ using the session key SK_{GS}. GS also notes the time at which the sensor data $Readings$ is received as T_{data}.

E. Blockchain Storage

This phase is used by the GS to create the transactions from sensor readings, create blocks of the transactions and

add them to the blockchain.

• BS_1: GS creates a transaction $Tr = \langle\ Readings,\ T_{data},\ Sign[H(Readings\ ||\ T_{data})]\ \rangle$ where "Elliptic Curve Digital Signature Algorithm (ECDSA)" is used for signature Sign[·].

• BS_2: Once $num_{transact}$ number of transactions are collected by a gateway server GS, a block is created as shown in Fig. 4.

• BS_3: One of the gateway servers in the P2P blockchain server farm will be elected as a leader. The "Practical Byzantine Fault Tolerance (PBFT)" consensus algorithm ensues among the gateway servers in the farm and the block proposed by the leader is added to the blockchain if the consensus is reached.

F. Big Data Analytics

This phase allows the use of sensor data stored on the blockchain to be used for analytics to obtain meaningful results.

• *Step BDA_1 (Business Case Data Identification):* This involves identifying the data appropriate for remote monitoring, supply chain, precision farming, automation and other relevant business use cases.

• *Step BDA_2 (Data Filtration and Extraction):* The received data is organized for logical partitioning and extracted for querying.

• *Step BDA_3 (Data Cleaning):* Errors and incompatibilities in the data are removed.

• *Step BDA_4 (Data Aggregation):* The data is summarized and aggregated for easy storage and retrieval.

• *Step BDA_5 (Data Analysis and Modelling):* Techniques such data data mining, learning and visualization are applied for meaningful and usable output.

V. Security Analysis

1) Replay Attack: In a session s_1, let SN generate $r_{sg}^{s_1}$, $t_{sg}^{s_1}$ to send a message $Msg_1^{s_1} = \langle N_S^{s_1}, R_{sg}^{s_1}, Sig_{sg}^{s_1}, TID_S^{s_1}, t_{sg}^{s_1}\rangle$. Suppose an adversary Adv captures this message in s_1 session and replays it in session s_2. For session s_2, SN generates $r_{sg}^{s_2}$, $t_{sg}^{s_2}$ and computes $N_S^{s_2}, R_{sg}^{s_2}$. For the adversary to be successful in replay attack, the hashes $N_S^{s_1} = H(n_s||\ RID_S^* s_1||\ TID_S||\ prv_S||\ t_{sg}^{s_1}) \stackrel{?}{=} N_S^{s_2} = H(n_s||\ RID_S^* s_2||\ TID_S||\ prv_S||\ t_{sg}^{s_2})$ and $R_{sg}^{s_1} = H(r_{sg}^{s_1}||\ RID_S^* s_1||\ TID_S||\ prv_S||\ t_{sg}^{s_1}) \stackrel{?}{=} R_{sg}^{s_2} = H(r_{sg}^{s_2}||\ RID_S^* s_2||\ TID_S^{s_2}||\ prv_S^{s_2}||\ t_{sg}^{s_2})$. According to birthday attack, a collision for each hash with q bits occurs with a probability $1/2^{q/2}$. When $q \geq 160$, this probability is negligible. Mismatch of the timestamps $t_{sg}^{s_1} \neq t_{sg}^{s_2}$ will fail signature verification at GS. Due to this, the replayed message $Msg_1^{s_2}$ will be identified and discarded. By similar logic, the replayed message $Msg_2^{s_2}$ will be identified and discarded. Thus, the scheme is secure against replay attack.

2) Man-in-the-Middle (MiTM) Attack: The adversary Adv may capture the message $Msg_1 = \langle N_S,\ R_{sg},\ Sig_{sg},\ TID_S,\ t_{sg}\rangle$ and modify it to $Msg_1^{Adv} = \langle N_S^{Adv},\ R_{sg}^{Adv},\ Sig_{sg}^{Adv},\ TID_S^{Adv},\ t_{sg}^{Adv}\rangle$ before forwarding it to GS. However, Adv does not have access to private parameters n_s, r_{sg}, identities ID_s, RID_S, RID_S^* and timestamp t_{sg}. It is computationally hard to generate the correct combination of all these parameters that verify the signature Sig_{sg} correctly at GS. Hence, MiTM is easily detected on Msg_1. By a similar logic, Msg_2 is also safe from MiTM attacks.

Block Header	
Block Version ($Block\_Version$)	Unique block version number
Previous Block Hash ($Previous\_Block\_Hash$)	Hash value of previous block
Merkle Tree Root ($MTRoot_{TX}$)	Merkle tree root on transactions
Timestamp (T_{GS})	Block creation time
Owner of Block	Gateway server (GS)
Public key of transactions verification	$Publ_G$
Public key of block signer	$Publ_G$
Block Payload (Encrypted Transactions)	
MV Transactions Tx_i	$\{Tr_i \mid i = 1, 2, \cdots, num_{transact}\}$
ECDSA signature on Block	Sig_{Block}
Current Block Hash ($Current\_Block\_Hash$)	Hash value of current block

Fig. 4. Structure of a block in the blockchain

3) Impersonation Attacks: Here, we consider the following two types of impersonation attacks.

• *Smart Device Impersonation Attack:* Consider that Adv captures $Msg_1 = \langle N_S, R_{sg}, Sig_{sg}, TID_S, t_{sg}\rangle$ and modify it to $Msg_1^{Adv} = \langle N_S^{Adv}, R_{sg}^{Adv}, Sig_{sg}^{Adv}, TID_S^{Adv}, t_{sg}^{Adv}\rangle$. However, this requires Adv to know the secrets n_s, r_{sg} that are not available in public channels.

• *Gateway Server Impersonation Attack:* Consider that Adv captures $Msg_2 = \langle N_G, R_{sg}, Sig_{sg}, TID_S, t_{sg}\rangle$ and modify it to $Msg_1^{Adv} = \langle N_S^{Adv}, R_{sg}^{Adv}, Sig_{sg}^{Adv}, TID_S^{Adv}, t_{sg}^{Adv}\rangle$. However, this requires Adv to know the secrets n_s, r_{sg} that are not available in public channels.

4) Privileged Insider Attack: Let a privileged insider with legitimate credentials may attempt to access critical credentials and apply them for malicious use. This is not possible in the proposed system as these critical credentials have been pre-loaded in the IoT smart device memory and via offline registration in the gateway server. Thus, the proposed system is resistant to privileged insider attacks.

5) Physical Device Node Capture attack: If Adv captures the IoT smart device physically and attempts to extract credentials from its memory, the attack will be unsuccessful as the available information in SN's memory $\{(RID_S,\ TID_S,\ n_S), H(\cdot), E_q(a,b), G, (prv_s, Publ_S)\}$ does not compromise communication between uncompromised nodes and gateway server.

6) Ephemeral Secret Leakage attack: The computed session key $SK_{GS} = H(DK_{GS}||\ N_S||\ H(\ n_g||\ RID_G^*||\ TID_G||\ prv_G||\ t_{gs})||\ R_S||\ H(\ r_{gs}||\ RID_G^*||\ TID_G||\ prv_G||\ t_{gs}) = SK_{SG} = H(DK_{SG}||\ H(n_s||\ RID_S^*||\ TID_S||\ prv_S||\ t_{sg})||\ N_G||\ H(r_{sg}||\ RID_S^*||\ TID_S||\ prv_S||\ t_{sg})||\ R_{gs})$ depends on both long-term secrets n_S, n_G, prv_S, prv_G and short-term secrets r_{sg}, r_{gs}. Compromise of session key with knowledge of only short-term secrets r_{sg}, r_{gs} is impossible without possessing long-term secrets. Similarly, compromise of session

key with knowledge of only long-term secrets n_S, n_G, prv_S, prv_G is impossible without possessing short-term secrets.

7) Denial of Service (DoS) Attack: A denial of service attack on the proposed system would rely on replaying the earlier messages multiple times to the gateway server to overwhelm it with requests. However, both the messages Msg_1 and Msg_2 contain timestamps that are to be verified for freshness at the receiver end before ay processing of requests. Multiple requests sent in DoS attack would fail the freshness test and thus are rejected. Hence, the proposed system is secure against DoS and DDoS attacks.

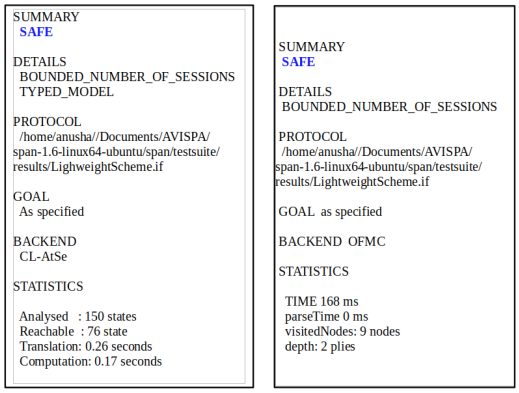

Fig. 5. AVISPA simulation results of Proposed Scheme on Cl-ATSe and OFMC backends

8) Forward Secrecy: Forward secrecy is maintained if the secret key established in a session cannot be reused in the subsequent sessions. The session key $SK_{SG}(= SK_{GS})$ consists of the random secrets r_{sg}, r_{gs} and timestamps t_{sg}, t_{gs} generated uniquely for every session and discarded after the session. Hence, the proposed system maintains forward secrecy.

9) Anonymity and Untraceability: The true identities ID_S and ID_G are never stored or revealed. The pseudo-identities RID_S and RID_G are hidden as RID_S^* and RID_G^*. Neither the pseudo-identities no the hidden pseudo-identities are shared in any of the messages. In addition, the messages are all distinct because of the uniqueness of random secrets nd timestamps.hence, none of the messages are traceable to their sender.

A. Formal Verification using AVISPA: Simulation Study

A widely known validation tool called "Automated Validation of Internet Security Protocols and Applications (AVISPA)" is used to analyze the protocol and "Security Protocol ANimator for AVISPA (SPAN)" is used to simulate the protocol. The protocol is coded in "High Level Protocol Specification Language (HLPSL)" with three roles tadmin (TA), sensor (SN) and gateway (GS). Authentication goal is applied on the parameters r_{sg}, r_{gs}, t_{sg}, t_{gs} using the request and witness types for checking weak and strong authentication properties, respectively. Fig. 5 shows that the proposed system is safe against replay and MiTM attacks.

TABLE I
COMPARISON OF COMPUTATION COSTS

Protocol	Smart device/Drone end	GSS/Server end
Ali *et al.* [10]	$11T_h + T_{fe} + 3T_{senc/sdec}$ $\approx$ 5.738 ms	$8T_h + 5T_{senc/sdec}$ $\approx$ 0.445 ms
Chen *et al.* [11]	20 $T_h \approx$ 6.18 ms	17 $T_h \approx$ 0.935 ms
Chae and Cho [12]	8 T_{ecm} + 8 T_h + 2$Teca$ $\approx$ 20.808 ms	–
Rangwani *et al.* [13]	8 T_h + 5 $T_{ecm} \approx$ 13.912 ms	7 T_h + $T_{ecm} \approx$ 1.059 ms
Wu and Tsai [14]	$2T_{bp} + 2T_{exp} + 2T_{enc/dec} + 1T_h \approx$ 64.965 ms	$2T_{bp} + 2T_{exp} + 2T_{enc/dec} + 1T_h \approx$ 9.407 ms
Proposed	$7T_H$ + $5T_{ecm}$ + $T_{eca} \approx$ 13.619 ms	$7T_H$ + $5T_{ecm}$ + $T_{eca} \approx$ 3.757 ms

TABLE II
COMPARISON OF COMMUNICATION COSTS

Protocol	No. of messages	Total cost (in bits)
Ali *et al.* [10]	5	5504
Chen *et al.* [11]	4	4960
Chae and Cho [12]	4	12896
Rangwani *et al.* [13]	5	4128
Wu and Tsai [14]	10	$1344 + 256n$
Proposed	2	2240

Note: n: "number of agricultural equipment nodes (sensor devices) in Wu and Tsai's scheme [14]"

VI. COMPARATIVE STUDY

This section compares the proposed system with the existing schemes in Ali *et al.* [10], Chen *et al.* [11], Chae and Cho [12], Rangwani *et al.* [13] and Wu and Tsai [14].

A. Computational Costs

The representations T_H, T_{ecm}, T_{eca}, T_{exp}, T_{bp} and $T_{senc/sdec}$ denote the computational time for executing the operations of "one-way hash function using SHA-256 [25]", "elliptic curve multiplication", "elliptic curve addition", "modular exponentiation", "bilinear pairing" and "symmetric encryption/decryption using AES-128 [26]", respectively. The proposed system takes $7T_H$ + $5T_{ecm}$ + T_{eca} computation time each at the smart IoT device and the gateway server. Table I shows comparison of computation costs.

B. Communication Costs

To compare the proposed scheme with the other schemes, the identities are taken to be 160 bits, the timestamps are taken as 32 bits, the hash output from SHA-256 is taken as 256 bits and the ciphertext from symmetric encryption/decryption using AES-128. The communication cost of Msg_1 is 992 bits and Msg_2 is 1248 bits, making a total of 2240 bits cost for the scheme. Table II shows comparison of communication costs.

C. Security and Functionality Features

In [10], [11], [13], failure at the gateway and base stations can disconnect parts of the network. [12] has a field support center whose failure will collapse the entire network system. The comparison of security and functionality features of the schemes in Table III concludes that the proposed scheme

TABLE III
COMPARISON OF SECURITY AND FUNCTIONALITY FEATURES

Scheme	Anonymity	Untraceability	Dynamic node addition	Device impersonation attack	Stolen mobile device attack	ESL attack	Privileged insider attack	Replay attack	MiTM attack	Mutual authentication	Unauthorized login detection	DoS attack	Offline guessing attack	Blockchain support	Fully distributed (This paper)
Ali *et al.*	×	×	✓	×	×	×	×	✓	✓	✓	✓	×	✓	×	NA
Chen *et al.*	✓	✓	✓	✓	×	×	×	✓	✓	✓	✓	✓	✓	×	NA
Chae and Cho	×	×	×	×	NA	×	×	×	×	✓	✓	×	×	×	NA
Rangwani *et al.*	×	×	×	✓	✓	×	✓	✓	✓	✓	✓	×	✓	×	NA
Wu and Tsai	✓	✓	×	NA	NA	×	✓	✓	✓	✓	✓	✓	×	✓	×
Proposed	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Note: NA: "Not applicable"

attains all the desired security and functionality features with low communication and computation cost and hence is superior to all the other schemes.

VII. CONCLUSION

A novel lightweight blockchain based authentication and key agreement scheme based on a fully distributed architecture has been proposed in this paper. The proposed scheme provides anonymity, untraceability, forward secrecy in addition to robustness against ephemeral secret leakage attack and privileged insider attack that very few existing protocols can guard against, as shown in the security analysis and comparative study with the existing schemes. In addition, the proposed scheme has the lowest communication cost and significantly low computation cost while satisfying all the security and functionality features.

REFERENCES

[1] G. N. Gina Kennedy and P. Shetty, "Globalization of food systems in developing countries," *Globalization of food systems in developing countries: impact on food security and nutrition*, no. 83, p. 1, 2004.

[2] "Ukraine/Russia: As War Continues, Africa Food Crisis Looms," https://african.business/2022/04/apo-newsfeed/ukraine-russia-as-war-continues-africa-food-crisis-looms/.

[3] D. Malpass, "A new global food crisis is building," April 2022, https://blogs.worldbank.org/voices/new-global-food-crisis-building.

[4] "Explainer: How Indonesia's palm oil export ban will impact consumers," https://www.moneycontrol.com/news/business/companies/explainer-how-indonesias-palm-oil-export-ban-will-impact-consumers-8408851.html.

[5] "Three Ruling Crops of the World," https://www.downtoearth.org.in/news/food/three-crops-rule-the-world-what-it-means-for-the-planet-s-wildlife-81781.

[6] A. K. Das and B. Bruhadeshwar, "An Improved and Effective Secure Password-Based Authentication and Key Agreement Scheme Using Smart Cards for the Telecare Medicine Information System," *Journal of Medical Systems*, vol. 37, no. 5, p. 9969, 2013.

[7] D. Mishra, A. K. Das, and S. Mukhopadhyay, "A secure and efficient ECC-based user anonymity-preserving session initiation authentication protocol using smart card," *Peer-to-Peer Networking and Applications*, vol. 9, no. 1, pp. 171–192, Jan 2016.

[8] N. Garg, M. Wazid, A. K. Das, D. P. Singh, J. J. P. C. Rodrigues, and Y. Park, "BAKMP-IoMT: Design of Blockchain Enabled Authenticated Key Management Protocol for Internet of Medical Things Deployment," *IEEE Access*, vol. 8, pp. 95 956–95 977, 2020.

[9] B. Bera, S. Saha, A. K. Das, and A. V. Vasilakos, "Designing Blockchain-Based Access Control Protocol in IoT-Enabled Smart-Grid System," *IEEE Internet of Things Journal*, vol. 8, no. 7, pp. 5744–5761, 2021.

[10] R. Ali, A. K. Pal, S. Kumari, M. Karuppiah, and M. Conti, "A secure user authentication and key-agreement scheme using wireless sensor networks for agriculture monitoring," *Future Generation Computer Systems*, vol. 84, pp. 200–215, 2018.

[11] M. Chen, T.-F. Lee, and J.-I. Pan, "An Enhanced Lightweight Dynamic Pseudonym Identity Based Authentication and Key Agreement Scheme Using Wireless Sensor Networks for Agriculture Monitoring," *Sensors*, vol. 19, no. 5, 2019.

[12] C.-J. Chae and H.-J. Cho, "Enhanced secure device authentication algorithm in P2P-based smart farm system," *Peer-to-peer networking and applications*, vol. 11, no. 6, pp. 1230–1239, 2018.

[13] D. Rangwani, D. Sadhukhan, S. Ray, M. K. Khan, and M. Dasgupta, "An improved privacy preserving remote user authentication scheme for agricultural wireless sensor network," *Transactions on Emerging Telecommunications Technologies*, vol. 32, no. 3, p. e4218, 2021.

[14] H. Wu and C. Tsai, "An intelligent agriculture network security system based on private blockchains," *Journal of Communications and Networks*, vol. 21, no. 5, pp. 503–508, 2019.

[15] J. Arshad, M. A. B. Siddique, Z. Zulfiqar, A. Khokhar, S. Salim, T. Younas, A. U. Rehman, and A. Asad, "A Novel Remote User Authentication Scheme by using Private Blockchain-Based Secure Access Control for Agriculture Monitoring," in *International Conference on Engineering and Emerging Technologies (ICEET)*, 2020, pp. 1–9.

[16] A. Bothe, J. Bauer, and N. Aschenbruck, "RFID-assisted Continuous user authentication for IoT-based smart farming," in *IEEE International Conference on RFID Technology and Applications (RFID-TA)*, 2019, pp. 505–510.

[17] S. Alyahya, W. U. Khan, S. Ahmed, S. N. K. Marwat, and S. Habib, "Cyber Secure Framework for Smart Agriculture: Robust and Tamper-Resistant Authentication Scheme for IoT Devices," *Electronics*, vol. 11, no. 6, 2022.

[18] A. Vangala, A. K. Das, N. Kumar, and M. Alazab, "Smart secure sensing for iot-based agriculture: Blockchain perspective," *IEEE Sensors Journal*, vol. 21, no. 16, pp. 17 591–17 607, 2021.

[19] A. Vangala, A. K. Das, V. Chamola, V. Korotaev, and J. J. Rodrigues, "Security in IoT-enabled Smart Agriculture: Architecture, Security Solutions and Challenges," *Cluster Computing*, April 2022, DOI:10.1007/s10586-022-03566-7.

[20] A. Vangala, A. K. Das, and J.-H. Lee, "Provably secure signature-based anonymous user authentication protocol in an internet of things-enabled intelligent precision agricultural environment," *Concurrency and Computation: Practice and Experience*, p. e6187, March 2021.

[21] B. Bera, A. Vangala, A. K. Das, P. Lorenz, and M. K. Khan, "Private blockchain-envisioned drones-assisted authentication scheme in IoT-enabled agricultural environment," *Computer Standards & Interfaces*, vol. 80, p. 103567, 2022.

[22] A. Vangala, A. K. Sutrala, A. K. Das, and M. Jo, "Smart contract-based blockchain-envisioned authentication scheme for smart farming," *IEEE Internet of Things Journal*, vol. 8, no. 13, pp. 10 792–10 806, 2021.

[23] D. Dolev and A. Yao, "On the security of public key protocols," *IEEE Transactions on Information Theory*, vol. 29, no. 2, pp. 198–208, 1983.

[24] R. Canetti and H. Krawczyk, "Universally Composable Notions of Key Exchange and Secure Channels," in *International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT'02)*, Amsterdam, The Netherlands, 2002, pp. 337–351.

[25] W. E. May, "Secure Hash Standard," 2015, FIPS PUB 180-1, National Institute of Standards and Technology (NIST), U.S. Department of Commerce, April 1995. http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.180-4.pdf. Accessed on January 2020.

[26] "Advanced Encryption Standard," 2001, FIPS PUB 197, National Institute of Standards and Technology (NIST), U.S. Department of Commerce, November 2001. http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf. Accessed on May 2022.