Journal of Systems Architecture 160 (2025) 103365

Contents lists available at ScienceDirect

Journal of Systems Architecture

journal homepage: www.elsevier.com/locate/sysarc

# Designing secure blockchain-based authentication and key management mechanism for Internet of Drones applications

Mohammad Wazid [a], Saksham Mittal [a,b], Ashok Kumar Das [c,d,*], SK Hafizul Islam [e,**], Mohammed J.F. Alenazi [f], Athanasios V. Vasilakos [g]

[a] Department of Computer Science and Engineering, Graphic Era Deemed to be University, Dehradun 248 002, India
[b] Department of Computer Science and Engineering, Graphic Era Hill University, Dehradun 248 002, India
[c] Center for Security, Theory and Algorithmic Research, International Institute of Information Technology, Hyderabad 500 032, India
[d] Department of Computer Science and Engineering, College of Informatics, Korea University, 145 Anam-ro, Seongbuk-gu, Seoul 02841, South Korea
[e] Department of Computer Science and Engineering, Indian Institute of Information Technology Kalyani, West Bengal 741 235, India
[f] Department of Computer Engineering, College of Computer and Information Sciences (CCIS), King Saud University, Riyadh 11451, Saudi Arabia
[g] Center for AI Research (CAIR), University of Agder (UiA), 4879 Grimstad, Norway

ARTICLE INFO

*Keywords:*
Internet of Drones
Blockchain
Authentication
Key agreement
Session key
Security

ABSTRACT

Due to advancement in Information and Communications Technology (ICT) and Internet of Things (IoT), the Internet of Drones (IoD) can be employed in numerous applications, facilitating the daily lives of diverse users, including civilians and others. Wireless communication nature leads to an IoD environment to be vulnerable to various potential attack risks, such as data breaches, man-in-the-middle, impersonation, replay, and data leaking attacks. As a result, the security of the IoD environment becomes crucial. To safeguard the data and devices (such as IoT-enabled drones and servers) integral to IoD networks, a security solution is essential. It is imperative to implement targeted security measures, such as intrusion detection, access control, and authentication, in order to establish a security scheme that is both reliable and efficient. In this article, we mainly focus on developing a secure authentication and key management scheme that leverages blockchain technology. Most existing authentication techniques proposed in IoT and IoD environments are either inefficient in communication and computation, or they are insecure against various attacks. To mitigate these issues, this study proposes a secure blockchain-based authentication and key management scheme for IoD applications (in short BAKMM-IoD). The blockchain is applied here as a secure data storage purpose. After performing a detailed security analysis and formal security verification with the widely-recognized Scyther tool, the proposed BAKMM-IoD has exhibited resilience against different potential attacks. BAKMM-IoD also surpasses other contemporary existing schemes in terms of security and functionality features, including computational costs, and communication costs. Moreover, the blockchain simulation shows that the influence of the proposed BAKMM-IoD on critical performance metrics in real-world scenarios.

## 1. Introduction

Drones refer to unmanned aerial vehicles (UAVs) capable of autonomous flight without the physical presence of a pilot or aviator. The term "unmanned aerial vehicles" (UAVs) specifically denotes drones. Drones are commonly battery-operated devices. In addition, their information processing and storage capabilities are finite. The creation of energy-efficient and economical micro-controller designs has accelerated the progress of drone-based monitoring and control systems. This is a consequence of the accelerated pace at which technology is advancing. Drones are employed in various sectors, including environmental monitoring, search and rescue operations during natural disasters, and the oversight of ecologically sensitive regions, including agricultural lands and forest fires [1]. The Internet of Drones (IoD) is a novel framework founded on the principles of the Internet of Things (IoT). Drones serve as replacements for physical objects inside this framework.

* Corresponding author at: Center for Security, Theory and Algorithmic Research, International Institute of Information Technology, Hyderabad 500 032, India.

** Corresponding author.
*E-mail addresses:* wazidkec2005@gmail.com (M. Wazid), mittalsaksham07@gmail.com (S. Mittal), iitkgp.akdas@gmail.com, ashok.das@iiit.ac.in (A.K. Das), hafi786@gmail.com, hafi786@iiitkalyani.ac.in (SKH Islam), mjalenazi@ksu.edu.sa (M.J.F. Alenazi), thanos.vasilakos@uia.no (A.V. Vasilakos).

https://doi.org/10.1016/j.sysarc.2025.103365
Received 13 November 2024; Received in revised form 12 January 2025; Accepted 6 February 2025
Available online 15 February 2025

IoT has enhanced communication and interaction among drones, enabling remote control in scenarios where direct optical transmission is impractical. An additional element of the IoD is the onboard controller, which employs artificial intelligence to make robust decisions [2–4]. IoD has various applications as discussed earlier. Cybersecurity concerns confronting the IoD. Some of them are as follows. Instances of data theft occur when adversaries illicitly intercept conversations and pilfer data, including control and command signals that are utilized to guide the drone [5]. Further, by exploiting vulnerabilities in drone software, adversaries can remotely seize control of drones and hijack them for their own objectives. Moreover, the faking of GPS signals by drones is facilitated by malicious software, therefore enabling their use for harmful purposes. Apart from that unauthorized access to the IoD systems can also be possible. An antagonistic user, such as an attacker or hacker, can intercept the IoD network, enabling them to bypass it and execute man-in-the-middle (MiTM) attacks. Intercepting the collected drone data is also feasible [5,6].

### 1.1. Potential ethical concerns belong to IoD communication

Here, we discuss the key ethical concerns that belong to the IoD communication. It includes data sovereignty problems, because drones operate across different borders may be from different countries or states of a country, which potentially violates local laws (for example, the laws on data storage and its processing). Another potential challenge is "General Data Protection Regulation (GDPR)". It is the European Union (EU) law that regulates how organizations handle personal data [7]. It complies with the risk of unauthorized personal data collection and excessive data processing [8]. IoD communication also faces concerns of data privacy, surveillance and accountability. To address these issues and challenges, some of the strategies, such as data localization, privacy-by-design, use of strong encryption and global regulatory standards, are necessarily needed [9].

### 1.2. Research motivation

While IoD fulfills various functions, enhancing the daily lives of a wide range of users and citizens, its communication framework is also vulnerable to numerous risks, including data leakage, impersonation, replay, drone physical capture, stolen verifier attack, credentials/secret keys/session keys leakage, Ephemeral Secret Leakage (ESL), malware injection and cross-site scripting attacks. The security of the IoD becomes vital, as it safeguards against numerous threats, including data breaches, privacy infringements, and other security issues [10]. Preventive security measures can be adopted to alleviate these risks. Drones lacking robust cybersecurity protections are susceptible to numerous risks. Therefore, to safeguard information and devices (including drones and servers) within IoD networks, a security mechanism is essential. Establishing a resilient security architecture requires the deployment of particular security measures, such as authentication, intrusion detection, and access control [11,12]. Moreover, the adoption of blockchain technology can bolster security against various potential threats and attacks [3].

The security of blockchain technology stems from its decentralized architecture and the application of encryption. Blockchains are decentralized networks that utilize a consensus (agreement) mechanism. Consequently, any effort to alter data can be identified by other nodes within the network. Blockchains employ cryptographic methods, including public-key cryptography (i.e., "Elliptic Curve Cryptography (ECC)"), to secure data and enable the generation of digital signatures. This method protects data from unauthorized access and ensures its confidentiality and integrity. Each data block in the chain is inherently connected to the preceding and subsequent blocks to create an immutable record of transactions. It is worth noticing that a block is immutable and cannot be modified once it has been integrated into the chain [3]. In this article, we propose a secure blockchain-based authentication and key management scheme that is applicable in various IoD-based real-life applications.

### 1.3. Research contributions

The following list outlines the research contributions made in this article.

- A secure blockchain-based authentication and key management mechanism is proposed for IoD applications (in short, we call it as BAKMM-IoD).
- The proposed BAKMM-IoD has demonstrated to be secured against a wide range of potential threats after an extensive security analysis and formal verification utilizing the widely recognized Scyther tool.
- The BAKMM-IoD has been shown to surpass other similar contemporary methods for functionality, security, computational overheads, and communication overheads.
- A functional illustration of the proposed BAKMM-IoD is subsequently shown to demonstrate its applicability to real-world settings.

## 2. Literature review

Authentication is one of the very important security services that can be applied in various networking domains [13–19].

The safe authentication mechanism utilizing blockchain technology was proposed by Yazdinejad et al. [20]. Drones were designed to execute the planned deployment of the strategy in smart cities. At every stage of the process, this approach guaranteed the least amount of delays. A zone-based architecture was devised for a drone network, and a decentralized consensus mechanism tailored for remote drone use in smart cities was deployed.

Bera et al. [21] introduced ACSUD-IoD, an innovative access control system designed to identify and thwart unwanted unmanned aerial vehicles (UAVs) within the IoD. The storing of transactional data within a private blockchain framework was enabled by the integration of a blockchain-based solution with ACSUD-IoD. This encompassed the delivery of secure, standardized data from an UAV to the ground station server. Consequently, the transactional data on the blockchain is verifiable. A formal security verification was performed utilizing the "Automated Validation of Internet Security Protocols and Applications (AVISPA) tool", alongside a comprehensive security evaluation. It illustrated that their method was adequately protected against several possible threats.

Feng et al. [22] proposed a "cross-domain authentication protocol grounded in blockchain technology". This system was designed to use 5G technology for diverse IoD applications. The aforementioned limits were duly acknowledged during the formulation of this plan with the aim of transcending them. Their methodology was based on a varied collection of signatures, all produced via threshold sharing. As a result, they successfully established a productive identity federation for collaborative domains.

Cho et al. [23] developed an authentication mechanism for unmanned aerial vehicles (UAVs) to reduce security threats linked to unauthorized drones utilizing the IoD concept. Although their methodology reduced communication and computational requirements, yet their architecture was vulnerable to the "Ephemeral Secret Leakage (ESL) attack under the CK-adversary model". The method insufficiently protected the anonymity and untraceability of the participants. Another element that contributed to this issue was the absence of blockchain technology in their proposed strategy.

Gupta et al. [24] presented a GaRuDa system, which might potentially denoted as the drone-based delivery system that operated on the blockchain technology. The integration of this system into the operations of Healthcare 5.0 applications was feasible. The IoT and blockchain technology were utilized in their approach to enable the swift and accurate distribution of medical supplies, which could be continuously monitored and recorded by many stakeholders. This was achieved by using a 5G-enabled Internet environment.

A pair of unique communication strategies for UAV environments were developed by Rodrigues et al. [25]. Their scheme facilitated the establishment of a direct exchange of messages between two drones. The presented scheme was derived from the existing scheme proposed in [26]. Nevertheless, the main contractual arrangement has been altered within the framework of this strategy. In accordance with the CK-adversary concept, their scheme was not impervious to the possibility of an ESL attack. Moreover, their scheme lack support for the blockchain technology.

Ever [27] proposed an authentication system for IoT applications that used Elliptic Curve Cryptography (ECC). UAVs were considered to be mobile extensions of wireless sensor networks, operating within a hierarchical framework, according to their design. This particular design enabled the effective implementation of one-time user authentication for mobile sinks (UAVs), cluster chiefs, and sensor nodes. In contrast, their system was vulnerable to "ESL attack under the CK-adversary model". Moreover, their scheme did not ensure the maintenance and safeguarding of anonymity and untraceability. Another limitation of their scheme was absence of blockchain technology and it required more communication and computational costs.

Singh et al. [28] examined the evolution and potential applications of the Internet of Drones. The advanced development of this technology has generated several apprehensions, among which the degree of security offered by autonomous robots has always been a prominent issue. Hence, they emphasized the most urgent security vulnerabilities and suggested that the most efficient approach to address these challenges would be to adopt state-of-the-art blockchain technology.

Xiong et al. [29] introduced a secure collaborative computing system that implemented blockchain technology. They initially created a lightweight blockchain framework that was specifically designed for "Unmanned Aerial Vehicle (UAV) Ad-Hoc Networks (UANET)". Further, they introduced an improved "Practical Byzantine Fault Tolerance (PBFT)" consensus algorithm that was based on trust assessment.

Wang et al. [30] introduced a mutual authentication method that was both simple and effective, and it exclusively relied on one-way hash algorithms and bitwise XOR operations. Additionally, the issue of a centralized trusted authority (TA) was mitigated by blockchain technology. The Real-or-Random model-based formal security analysis was employed. Further, an informal security proof was provided to prove the security of their proposed authentication mechanism. Further, Wang et al. [31] introduced, "BSIF: Blockchain-Based Secure, Interactive, and Fair Mobile Crowdsensing" system. It was blockchain-based and was distinguished by its security, interactivity, and impartiality. These attributes were achieved through the integration of smart contracts and mobile devices. Yu et al. [32] presented a "Cross-domain Industrial IoT Based on Consortium Blockchain mechanism (CBDS) for the security of Industrial Internet of Things (IIoT). Further, they introduced consortium blockchain specifically to establish trust across IIoT domains.

Srinivas et al. [33] developed an innovative authentication technique that was anonymous, lightweight, and relied on temporal credentials for Internet of Things (IoT)-based platforms. It was denoted as $TCALAS$. To enhance $TCALAS$, Ali et al. [34] developed an improved version of $TCALAS$, referred to as $iTCALAS$ for the secure communication of IoD.

Mishra et al. [35] represented a framework for managing authentication and session keys using blockchain technology. This framework supported the integration of big data analytics capabilities for drones that operate on networks beyond 5G applications. Through a comprehensive security examination and scyther tool-based formal security verification, they have proven their scheme secured against the wide range of attacks.

In 2024, Algarni and Jan [36] proposed a robust yet lightweight security mechanism utilizing a fuzzy extractor and the MD5 (Message Digest 5) algorithm to authenticate all IoD participants and ensure secure communication. However, the MD5 hash algorithm is widely recognized as less secure compared to more robust alternatives like the Secure Hash Algorithm (SHA-256). Consequently, the overall strength of their scheme is compromised. Moreover, their approach does not incorporate support for blockchain implementation.

**Research gaps and novelty:** Blockchain technology offers powerful solutions to strengthen the security of the IoD environment. By enabling the creation of unique digital identities for individual drones, which are securely stored and managed on the blockchain, it helps mitigate the risk of impersonation attacks [37]. In addition, the data coming securely from the drones to the ground station server is used for the transactions and later, the blocks formed from the authentic and genuine data from the drones are stored in the blockchain network maintained by the cloud servers. Storing data on semi-trusted cloud servers raises serious concerns about data poisoning attacks, which can significantly impact businesses and organizations by corrupting big data analytics, leading to financial losses and reputational damage [38]. Research shows notable improvements in accuracy, recall, precision, and F1-score when data is free from poisoning attacks and is directly sourced from the blockchain. In this context, authentication among drones and other entities in the IoD environment becomes critical to ensure that genuine data is stored on the blockchain.

The literature review highlights that most existing authentication techniques for IoT and IoD environments are either inefficient in terms of communication and computation or vulnerable to various attacks. This underscores the need for a reliable and secure authenticated key agreement protocol to facilitate secure data aggregation at ground station servers in the IoD environment, with blockchain technology providing enhanced secure storage. Therefore, the objective of this work is to develop a novel and secure blockchain-based authentication and key management mechanism for IoD applications that is not only resistant to various attacks but also efficient in communication and computational costs, making it suitable for real-world practical applications.

## 3. System models

The system models which are related to the BAKMM-IoD are explained below. Moreover, the details of the network model and the threat model are given below.

### 3.1. Network model

Fig. 1 illustrates the proposed BAKMM-IoD's network model. This scenario involves several users, cloud servers, ground station servers, and several drones. The significant versatility of this architecture allows its application across various industries, including smart farming, industrial automation and control, intelligent transportation systems (ITS), and healthcare, among others. The drones are connected to the ground station servers, which are in turn connected to the cloud servers through communication channels. The ground station servers can consistently store the necessary data. Drones do not encounter excessive workloads as a substantial portion of computationally expensive tasks are managed by the ground station servers. The data gathered by the drones is relayed to ground station servers for further analysis and use. The partial blocks generated by the ground station servers from the received data are subsequently transmitted to the corresponding cloud-based servers.

Upon receiving partial data blocks, the cloud servers utilize them to reconstruct the complete block. The aforementioned blocks may ultimately be incorporated into the blockchain, contingent upon the successful completion of the consensus procedure. The peer-to-peer cloud server network (P2PCS) is responsible for maintaining the functionality of the blockchain. Due to the implementation of advanced technologies and substantial resources, the P2PCS network's cloud servers have exceptional processing, communication, and storage capabilities. The prevailing opinion is that cloud servers are semi-trusted

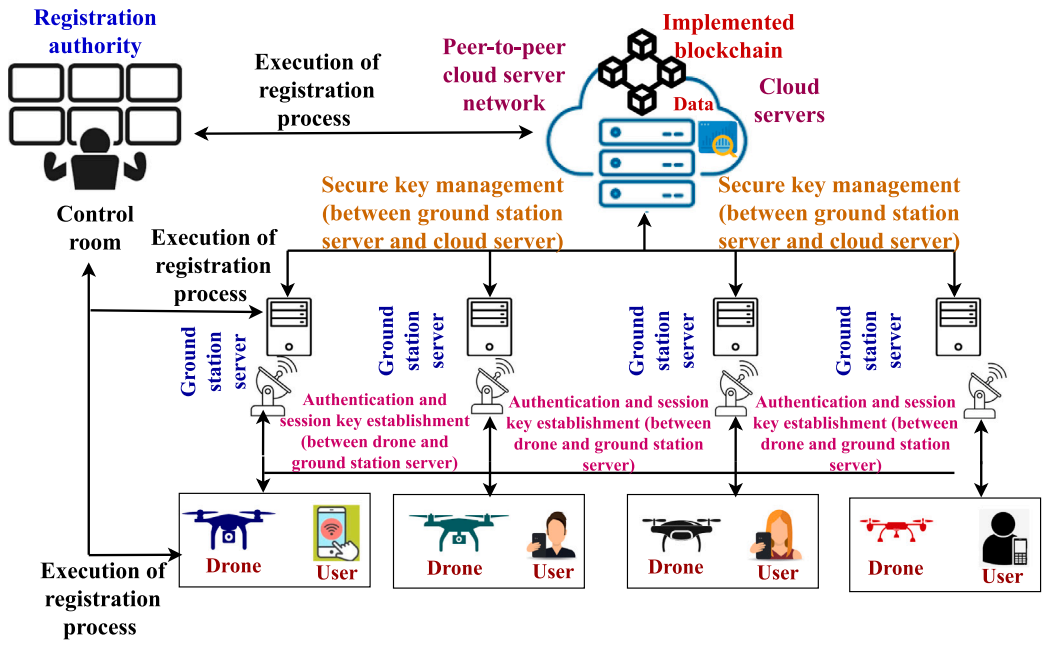


**Fig. 1.** Network model of the BAKMM-IoD.

network entities. Cyberattacks may compromise the communication occurring between drones, cloud servers, and ground station servers. The integrity of these communications may be jeopardized by the potential adversary $\mathcal{A}$. To guarantee system security, it is imperative to employ security measures such as authentication and key management under the present conditions. Insufficient implementation of this security feature may render the devices and servers susceptible to hackers. Potential hazards encompass "malware injection attacks, unauthorized data access, data replay attacks, man-in-the-middle (MiTM) attacks, impersonation attacks, and unauthorized session key estimation attacks".

### 3.2. Threat model

The proposed BAKMM-IoD is constructed based on the following threat models and assumptions.

- The Dolev–Yao (DY) threat model, which is widely acknowledged as the prevailing de-facto standard [39]. DY model states that two unprotected entities can communicate with each other across an open network, such as the Internet. Entities at endpoints that are often deemed untrustworthy comprise drones and ground station servers. Communications transferred across an unsecured network can be accessed, modified, or deleted by an adversary $\mathcal{A}$, irrespective of their active or passive status. The BAKMM-IoD is designed to counter many potential attacks. Examples of these attacks encompass the "physical drone capture attack, the ephemeral secret leakage (ESL) attack, the secret data leakage attack, the impersonation attack, the replay attack, the man-in-the-middle (MiTM) attack, among others".
- The proposed BAKMM-IoD has been designed with the Canetti and Krawczyk (CK) substantial adversary model as a consideration [40]. Currently, $\mathcal{A}$ possesses comprehensive access to all attributes related to the model DY. Furthermore, session states, encompassing session keys and credentials linked to a particular session, are obtained by $\mathcal{A}$.

The DY threat model and the CK adversary model focus on defending against those adversaries who possess the ability to alter communication channels while the cryptographic primitives remain intact. In the DY model, the adversary can intercept, modify, and inject messages, and then secure protocols need to use strong encryption and authenticity mechanisms to ensure confidentiality and integrity. Replay attacks must be prevented by the use of current timestamp values, and mutual authentication should be done using digital signatures or certificates which helps in establishing credibility between entities. In the case of the CK-adversary model, mitigation focuses primarily on ephemeral key exchanges to derive session keys even if the short-term secrets are compromised, since it extends the DY-model assumptions and supports forward secrecy and session independence. Both models call for formal validation of the protocols with the automated validation tools, like Scyther, to ensure that security properties are met. Following these strategies, cryptographic protocols will survive in environments against the DY and CK adversaries.

$\mathcal{A}$ may also physically capture a certain number of drones and extract data from their memory using an advanced power analysis method [41]. The collected information can be used to launch associated attacks and formulate additional malevolent acts, including impersonation efforts. The use of disguised session keys and credentials, together with privileged insider attacks, may be implemented in these attacks. Cloud servers are regarded as semi-trusted entities within the network because of their role in maintaining and storing system data. The registration authority (RA) of the control room, tasked with the registration of network entities, concurrently serves as the registration authority for the network. Moreover, it is expected that the system's security would be compromised if $RA$ were compromised, so undermining the system's overall integrity.

## 4. BAKMM-IoD: The proposed BAKMM-IoD

The proposed BAKMM-IoD is comprehensively described in this section. The BAKMM-IoD is a multifaceted process that includes registration, authentication and key establishment, key management, dynamic device integration, and blockchain implementation.

In the proposed BAKMM-IoD, the drones are communicating entities, which collect various data through their inbuilt units, i.e., sensors. After this data collection they send their data to the connected ground station servers in a secure way with the help of the proposed "authentication and key establishment phase". The ground station servers create partial blocks from this received data and then send it to the connected cloud servers in a secure way with the help of the given

**Table 1**
Notations used in BAKMM-IoD.

| Notation | Meaning |
|---|---|
| BAKMM-IoD | Short name of the proposed mechanism |
| $\mathcal{A}$ | An adversary |
| $DE_i$, $ID_{DE_i}$, $RID_{DE_i}$ | $i$th deployed drone, its identity and pseudo-identity, respectively |
| $ES_j$, $ID_{ES_j}$, $RID_{ES_j}$ | $j$th ground station server, its identity and pseudo-identity, respectively |
| $CS_k$, $ID_{CS_k}$, $RID_{CS_k}$ | $k$th cloud server, its identity and pseudo-identity, respectively |
| $RA$, $k_{RA}$ | The registration authority (trusted entity), its secret key and its pseudo-identity, respectively |
| $k_{DE_i}$, $k_{ES_j}$ | Private keys $DE_i$ and $ES_j$ |
| $SN_{RA}$, $SN_{DE_i}$ and $SN_{ES_j}$ | The secret numbers of $RA$, $DE_i$ and $ES_j$, respectively |
| $MS_{DE_i-ES_j}$ | primary secret key of both $DE_i$ and $ES_j$ |
| $MS_{ES_j-CS_k}$ | primary secret key of both $ES_j$ and $CS_k$ |
| $T_x$ | Different timestamp values used |
| $rs_x$ | Different random secret values used |
| $\Delta T$ | The allowed delay value to mitigate replay attack |
| $h(\cdot)$ | Cryptographic one-way hash function utilized |
| $SK_{a_i,b_j}$ | The session key obtained and established in between entities $a_i$ and $b_j$ |
| $\|$ | A concatenation computation |
| $\oplus$ | A bitwise exclusive-OR ($XOR$) computation |

"key management phase". The cloud servers are the part of peer-to-peer server network and does the task of blockchain implementation. Some of the cloud servers are also the miner nodes of the network and perform the task of blockchain mining with the help of the consensus algorithm.

The details of the used notations are provided in Table 1 The following is a concise overview of the phases.

### 4.1. Registration phase

In this phase, the registration authority ($RA$) is tasked with registering the entities, which comprise the drone ($DE_i$), the ground station server ($ES_j$), and the cloud server ($CS_k$). Comprehensive information is provided here.

#### 4.1.1. Registration of drone $DE_i$

The drone $DE_i$'s registration is performed as follows.

- *RSDI1:* Initially, $RA$ designates $SN_{RA}$ as its confidential (secret) number and $k_{RA}$ as its confidential key. The pseudo identity is subsequently computed as follows: $RID_{RA} = h(ID_{RA} \| SN_{RA} \| k_{RA})$. Subsequently, it designates $ID_{DE_i}$ as the identifier for $DE_i$, $k_{DE_i}$ as the confidential key, and $SN_{DE_i}$ as the confidential number. The pseudo identity of $DE_i$ is then calculated by $RA$ as $RID_{DE_i} = h(RID_{RA} \| ID_{DE_i} \| k_{RA} \| k_{DE_i} \| SN_{DE_i})$. It calculates the temporal credentials parameter of $DE_i$ using the formula $TC_{DE_i} = h(RID_{RA} \| ID_{DE_i} \| k_{RA} \| k_{DE_i} \| SN_{DE_i} \| RTS_{DE_i})$, where $RTS_{DE_i}$ is the registration timestamp value of $DE_i$. It generates $TID_{DE_i}$ as a provisional temporary identity for $DE_i$. The registration data has subsequently been stored in the memory of $DE_i$.
- *RSDI2:* Finally, $DE_i$ stores values $\{TID_{DE_i}, RID_{DE_i}, TC_{DE_i}, MS_{DE_i-ES_j}, h(.)\}$. Here, it is important to mention that $MS_{DE_i-ES_j}$ is the primary secret key of both $DE_i$ and $ES_j$, this key distinct for different drones. As we have different deployed $DE_i$, where $i = 1, 2, \ldots, n_{DE}$, and $n_{DE}$ is the number of deployed drones.

The above drone registration phase is also given in Table 2.

#### 4.1.2. Registration of ground station server $ES_j$

The registration of ground station server $ES_j$ is performed as follows.

- *RSES1:* First $RA$ chooses the secret key and secret number of $ES_j$ as $k_{ES_j}$ and $SN_{ES_j}$. Then $RA$ chooses its identity as $ID_{ES_j}$. Further, it computes pseudo identity number of $ES_j$ as $RID_{ES_j} = h(RID_{RA} \| ID_{ES_j} \| k_{RA} \| k_{ES_j} \| SN_{ES_j})$ and temporal credentials parameter as $TC_{ES_j} = h(RID_{RA} \| ID_{ES_j} \| k_{RA} \| k_{ES_j} \| SN_{ES_j} \| RTS_{ES_j})$, where $RTS_{ES_j}$ is the registration timestamp value of $ES_j$. $RA$ also generates a provisional temporary identification number for $ES_j$ as $TIN_{ES_j}$, and a secret primary key for $ES_j$ and cloud server $CS_k$ as $MS_{ES_j-CS_k}$. Here, it is important to mention that $MS_{ES_j-CS_k}$ are distinct for different ground station servers and cloud server. Then $RA$ stores the registration information of registered $DE_i$ and its own information in its database/memory.
- *RSES2:* Finally, $ES_j$ contains $\{\{(TID_{DE_i}, RID_{DE_i}) \mid i = 1, 2, \ldots, n_{DE}\}, TIN_{ES_j}, RID_{ES_j}, TC_{ES_j}, (MS_{DE_1-ES_j}, MS_{DE_2-ES_j} \ldots, MS_{DE_{n_{DE}}-ES_j}), MS_{ES_j-CS_k}, h(\cdot)\}$ in the region of its secured database, where $n_{DE}$ represents the entire quantity of drones deployed under ground station server $ES_j$.

The registration phase of ground station server $ES_j$ is given in Table 3.

#### 4.1.3. Registration of $CS_k$

The subsequent process is employed to register cloud server $CS_k$.

- *RSCS1:* First $RA$ chooses the secret key and secret number of $CS_k$ as $k_{CS_k}$ and $SN_{CS_k}$. Then $RA$ chooses its identity as $ID_{CS_k}$. Further, it calculates the pseudo identity of $CS_k$ as $RID_{CS_k} = h(RID_{RA} \| ID_{CS_k} \| k_{RA} \| k_{CS_k} \| SN_{CS_k})$ and temporal credentials parameter as $TC_{CS_k} = h(RID_{RA} \| ID_{CS_k} \| k_{RA} \| k_{CS_k} \| SN_{CS_k} \| RTS_{CS_k})$, where $RTS_{CS_k}$ is the registration timestamp value of $CS_k$.
- *RSCS2:* Finally, $CS_k$ contains $\{\{(TIN_{ES_j}, RID_{ES_j}) \mid j = 1, 2, \ldots, n_{ES}\}, RID_{CS_k}, TC_{CS_k}, (MS_{ES_1-CS_k}, MS_{ES_2-CS_k}, \ldots, MS_{ES_{n_{ES}}-CS_k}), h(\cdot)\}$ in its secured database, where $n_{ES}$ is the total number of ground station servers deployed under cloud server $CS_k$.

The registration phase of cloud server $CS_k$ is provided in Table 4.

### 4.2. Authentication phase

This section provides a detailed description of the mutual authentication and key establishment mechanism between a drone ($DE_i$) and its associated ground station server ($ES_j$). The following steps need to be executed:

- *AKDDE1:* The drone $DE_i$ produces a new timestamp value represented as $T_1$ and a random secret value denoted as $rs_1$. Further, it estimates some values as $M_1 = h(RID_{DE_i} \| MS_{DE_i-ES_j} \| T_1) \oplus h(TC_{DE_i} \| rs_1 \| MS_{DE_i-ES_j} \| T_1)$ and $M_2 = h(h(TC_{DE_i} \| rs_1 \| MS_{DE_i-ES_j} \| T_1) \| RID_{DE_i} \| T_1)$. It then sends message $MSG_1 = \{TID_{DE_i}, M_1, M_2, T_1\}$ to $ES_j$ through open channel, which is insecure in nature.
- *AKDDE2:* At the arrival of $MSG_1$, $ES_j$ checks condition $|T_1 - T_1^*| \leq \Delta T$, where the "maximum transmission delay" is given by $\Delta T$ and $T_1^*$ is receiving time of $MSG_1$. Here, it is important to say that $\Delta T$ also denotes the expected time interval for the transmission delay/preset acceptable delay threshold value. If the condition holds, $ES_j$ then fetches the values of $RID_{DE_i}$ and $MS_{DE_i-ES_j}$ from its memory which is corresponding to the received $TID_{DE_i}$. After that $ES_j$ computes $h(TC_{DE_i} \| rs_1 \| MS_{DE_i-ES_j} \| T_1) = M_1 \oplus h(RID_{DE_i} \| MS_{DE_i-ES_j} \| T_1)$. After $ES_j$ computes $M_2' = h(h(TC_{DE_i} \| rs_1 \| MS_{DE_i-ES_j} \| T_1) \| RID_{DE_i} \| T_1)$. Then it checks if $M_2' = M_2$? If it matches then $DE_i$ is

**Table 2**
Registration phase of drone $DE_i$.

| $RA$ | $DE_i$ |
|---|---|
| Generate $SN_{RA}$&$k_{RA}$.<br>Compute $RID_{RA} = h(ID_{RA} \| SN_{RA} \| k_{RA})$.<br>Generate $ID_{DE_i}$ for $DE_i$,<br>Generate $k_{DE_i}$&$SN_{DE_i}$ for $DE_i$<br>Compute $RID_{DE_i} = h(RID_{RA} \| ID_{DE_i} \| k_{RA} \| k_{DE_i} \| SN_{DE_i})$,<br>$TC_{DE_i} = h(RID_{RA} \| ID_{DE_i} \| k_{RA} \| k_{DE_i} \| SN_{DE_i} \| RTS_{DE_i})$.<br>Generate $TID_{DE_i}$ | |
| | Store $\{TID_{DE_i}, RID_{DE_i}, TC_{DE_i}, MS_{DE_i-ES_j}, h(.)\}$. |

**Table 3**
Registration phase of ground station server $ES_j$.

| $RA$ | $ES_j$ |
|---|---|
| Generate $k_{ES_j}$, $SN_{ES_j}$&$ID_{ES_j}$ for $ES_j$.<br>Compute $RID_{ES_j} = h(RID_{RA} \| ID_{ES_j} \| k_{RA} \| k_{ES_j} \| SN_{ES_j})$,<br>$TC_{ES_j} = h(RID_{RA} \| ID_{ES_j} \| k_{RA} \| k_{ES_j} \| SN_{ES_j} \| RTS_{ES_j})$.<br>Generate $TIN_{ES_j}$&$MS_{ES_j-CS_k}$. | |
| | Store $\{\{(TID_{DE_i}, RID_{DE_i}) \| i = 1, 2, \ldots, n_{DE}\}, TIN_{ES_j}, RID_{ES_j}, TC_{ES_j}, (MS_{DE_1-ES_j}, MS_{DE_2-ES_j} \cdots, MS_{DE_{n_{DE}}-ES_j}), MS_{ES_j-CS_k}, h(\cdot)\}$ |

**Table 4**
Registration phase of cloud server $CS_k$.

| $RA$ | $CS_k$ |
|---|---|
| Generate $k_{CS_k}$, $SN_{CS_k}$&$ID_{CS_k}$ for $CS_k$.<br>Compute $RID_{CS_k} = h(RID_{RA} \| ID_{CS_k} \| k_{RA} \| k_{CS_k} \| SN_{CS_k})$,<br>$TC_{CS_k} = h(RID_{RA} \| ID_{CS_k} \| k_{RA} \| k_{CS_k} \| SN_{CS_k} \| RTS_{CS_k})$. | |
| | Store $\{\{(TIN_{ES_j}, RID_{ES_j}) \| j = 1, 2, \ldots, n_{ES}\}, RID_{CS_k}, TC_{CS_k}, (MS_{ES_1-CS_k}, MS_{ES_2-CS_k}, \ldots, MS_{ES_{n_{ES}}-CS_k}), h(\cdot)\}$ |

authenticated with $ES_j$. Further, $ES_j$ produces a new timestamp value represented as $T_2$ and a random secret value denoted as $rs_2$. It then computes $M_3 = h(RID_{DE_i} \| MS_{DE_i-ES_j} \| T_2) \oplus h(RID_{ES_j} \| TC_{ES_j} \| rs_2 \| MS_{DE_i-ES_j} \| T_2)$ and a session key $SK_{ES_j,DE_i} = h(h(TC_{DE_i} \| rs_1 \| MS_{DE_i-ES_j} \| T_1) \| h(RID_{ES_j} \| TC_{ES_j} \| rs_2 \| MS_{DE_i-ES_j} \| T_2) \| T_1 \| T_2 \| RID_{DE_i} \| MS_{DE_i-ES_j})$. After that it computes $M_4 = h(SK_{ES_j,DE_i} \| T_1 \| T_2 \| RID_{DE_i})$. It generates a new temporary identity for $ES_j$ as $TID^{new}_{DE_i}$ and computes $M_5 = TID^{new}_{DE_i} \oplus h(h(TC_{DE_i} \| rs_1 \| MS_{DE_i-ES_j} \| T_1) \| RID_{DE_i} \| T_2)$. $ES_j$ then sends message $MSG_2 = \{M_3, M_4, M_5, T_2\}$ to $DE_i$ through open channel.

- *AKDDE3:* At the arrival of $MSG_2$, $DE_i$ checks condition $|T_2 - T_2^*| \leq \Delta T$, where $T_2^*$ is receiving time of $MSG_2$. If it matches then $DE_i$ compute $h(RID_{ES_j} \| TC_{ES_j} \| rs_2 \| MS_{DE_i-ES_j} \| T_2) = M_3 \oplus h(RID_{DE_i} \| MS_{DE_i-ES_j} \| T_2)$. After that $DE_i$ calculates the session key as $SK_{DE_i,ES_j} = h(h(TC_{DE_i} \| rs_1 \| MS_{DE_i-ES_j} \| T_1) \| h(RID_{ES_j} \| TC_{ES_j} \| rs_2 \| MS_{DE_i-ES_j} \| T_2) \| RID_{DE_i} \| T_1 \| T_2 \| MS_{DE_i-ES_j})$ and $M_4' = h(SK_{ES_j,DE_i} \| T_1 \| T_2 \| RID_{DE_i})$. It then checks condition $M_4' = M_4$? If it matches, $ES_j$ is authenticated with $DE_i$ and computed session key by $DE_i$ is correct. $DE_i$ then computes its new temporary identity as $TID^{new}_{DE_i} = M_5 \oplus h(h(TC_{DE_i} \| rs_1 \| MS_{DE_i-ES_j} \| T_1) \| RID_{DE_i} \| T_2)$. Further, it computes a session key verifier by generating another fresh timestamp value $T_3$, which is $M_6 = h(SK_{DE_i,ES_j} \| T_3)$. Here it is important to mention that $M_6$ is a session key verifier, with the help of $M_6$, $ES_j$ can check whether $DE_i$ has computed the correct session key or not. After that $DE_i$ sends message $MSG_3 = \{M_6, T_3\}$ to $ES_j$ through open channel.
- *AKDDE4:* At the arrival of $MSG_3$, $ES_j$ checks condition $|T_3 - T_3^*| \leq \Delta T$, where $T_3^*$ is receiving time of $MSG_3$. If it holds $ES_j$ computes $M_6' = h(SK_{ES_j,DE_i} \| T_3)$ and checks a condition $M_6' = M_6$? In the event of a match, $ES_j$ presumes that the session key generated by $DE_i$ is correct. In the following phase, both $DE_i$ and $ES_j$ establish the session key $SK_{DE_i,ES_j}$ $(= SK_{ES_j,DE_i})$ to facilitate the secure transmission of their data.

Table 5 offers a succinct overview of the authentication and key establishment mechanism. The above employed method provides the protection of the communication channel between drones and ground stations from external influences and interception of information. This is because the initially the channel between $DE_i$ and $ES_j$ is insecure. However, after the mutual authentication between $DE_i$ and $ES_j$, both the entities $DE_i$ and $ES_j$ are able to establish a common session key $SK_{DE_i,ES_j}$ $(= SK_{ES_j,DE_i})$ which can now be used for encrypting the data exchanged between them. In that way, no adversaries will be able to tamper with the data because the data is already being encrypted with the established session key which is unknown to the adversary. For protecting a communication channel from unauthorized access, we use the "Advanced Encryption Standard (AES-256) symmetric encryption" for reducing the computational time required for a drone.

### 4.3. Key management phase

This procedure is conducted to manage the keys shared by $ES_j$ and $CS_k$. Upon the successful conclusion of this process, $ES_j$ and $CS_k$ will securely transmit their data using the specifically generated session key $SK_{ES_j,CS_k}$.

- *AKDEC1:* $ES_j$ starts communication and produces a new timestamp value represented as $TS_1$ and a random secret value denoted as $RS_1$. Then, it computes $m_1 = h(TC_{ES_j} \| RS_1 \| MS_{ES_j-CS_k} \| TS_1) \oplus h(RID_{ES_j} \| MS_{ES_j-CS_k} \| TS_1)$ and $m_2 = h(h(RS_1 \| TC_{ES_j} \| MS_{ES_j-CS_k} \| TS_1) \| RID_{ES_j} \| MS_{ES_j-CS_k} \| TS_1)$. After these many computations $ES_j$ sends message $msg_1 = \{TIN_{ES_j}, m_1, m_2, TS_1\}$ to $CS_k$ through the open channel.
- *AKDEC2:* At the arrival of $msg_1$, $CS_k$ checks condition $|TS_1 - TS_1^*| \leq \Delta T$, where $TS_1^*$ is receiving time of $msg_1$. If it satisfies, then $CS_k$ fetches $RID_{ES_j}$, and $MS_{ES_j-CS_k}$ corresponding to received $TIN_{ES_j}$. Then, $CS_k$ computes $h(RS_1 \| TC_{ES_j} \| MS_{ES_j-CS_k} \| TS_1) = m_1 \oplus h(RID_{ES_j} \| MS_{ES_j-CS_k} \| TS_1)$ and $m_2' = h(h(rs_1 \| TC_{ES_j} \| MS_{ES_j-CS_k} \| TS_1) \| RID_{ES_j} \| MS_{ES_j-CS_k} \|$

**Table 5**
Authentication and key establishment between $DE_i$ and $ES_j$.

| $DE_i$ | $ES_j$ |
|---|---|
| Generate $rs_1$&$T_1$.<br>Compute<br>$M_1 = h(RID_{DE_i} \| MS_{DE_i-ES_j} \| T_1) \oplus h(TC_{DE_i} \|$<br>$rs_1 \| MS_{DE_i-ES_j} \| T_1)$<br>$M_2 = h(h(TC_{DE_i} \| rs_1 \| MS_{DE_i-ES_j} \| T_1) \| RID_{DE_i} \| T_1)$.<br>$MSG_1 = \{TID_{DE_i}, M_1, M_2, T_1\}$<br>(via open channel) | |
| | Check if $\|T_1 - T_1^*\| \leq \Delta T$? If so<br>Fetch $RID_{DE_i}$&$MS_{DE_i-ES_j}$<br>Compute<br>$h(TC_{DE_i} \| rs_1 \| MS_{DE_i-ES_j} \| T_1)$<br>$= M_1 \oplus h(RID_{DE_i} \| MS_{DE_i-ES_j} \| T_1)$.<br>$M_2' = h(h(TC_{DE_i} \| rs_1 \| MS_{DE_i-ES_j} \| T_1) \| RID_{DE_i} \| T_1)$.<br>Check if $M_2' = M_2$? If so,<br>generate $T_2$&$rs_2$<br>Compute<br>$M_3 = h(RID_{DE_i} \| MS_{DE_i-ES_j} \| T_2)$<br>$\oplus h(RID_{ES_j} \| TC_{ES_j} \| rs_2 \| MS_{DE_i-ES_j} \| T_2)$<br>$SK_{ES_j,DE_i} = h(h(TC_{DE_i} \| rs_1 \| MS_{DE_i-ES_j} \| T_1) \|$<br>$h(RID_{ES_j} \| TC_{ES_j} \|$<br>$rs_2 \| MS_{DE_i-ES_j} \| T_2) \| T_1 \| T_2 \| RID_{DE_i} \| MS_{DE_i-ES_j})$.<br>$M_4 = h(SK_{ES_j,DE_i} \| T_1 \| T_2 \| RID_{DE_i})$.<br>Generate $TID_{DE_i}^{new}$<br>Compute<br>$M_5 = TID_{DE_i}^{new} \oplus h(h(TC_{DE_i} \| rs_1 \|$<br>$MS_{DE_i-ES_j} \| T_1) \| RID_{DE_i} \| T_2)$.<br>$MSG_2 = \{M_3, M_4, M_5, T_2\}$<br>(via open channel) |
| Check $\|T_2 - T_2^*\| \leq \Delta T$? If so,compute<br>$h(RID_{ES_j} \| TC_{ES_j} \| rs_2 \| MS_{DE_i-ES_j} \| T_2)$<br>$= M_3 \oplus h(RID_{DE_i} \| MS_{DE_i-ES_j} \| T_2)$,<br>$SK_{DE_i,ES_j} = h(h(TC_{DE_i} \| rs_1 \| MS_{DE_i-ES_j} \| T_1) \| h(RID_{ES_j} \|$<br>$TC_{ES_j} \| rs_2 \| MS_{DE_i-ES_j} \| T_2) \| RID_{DE_i} \| T_1 \| T_2 \| MS_{DE_i-ES_j})$,<br>$M_4' = h(SK_{ES_j,DE_i} \| T_1 \| T_2 \| RID_{DE_i})$.<br>Check if $M_4' = M_4$? If so, compute<br>$TID_{DE_i}^{new} = M_5 \oplus h(h(TC_{DE_i} \| rs_1 \|$<br>$MS_{DE_i-ES_j} \| T_1) \| RID_{DE_i} \| T_2)$.<br>Generate $T_3$ & compute<br>$M_6 = h(SK_{DE_i,ES_j} \| T_3)$.<br>$MSG_3 = \{M_6, T_3\}$<br>(via open channel) | |
| | Check $\|T_3 - T_3^*\| \leq \Delta T$? If so,<br>compute $M_6' = h(SK_{ES_j,DE_i} \| T_3)$<br>Check $M_6' = M_6$? If so,<br>store session key $SK_{ES_j,DE_i}$ |
| Store session key $SK_{DE_i,ES_j}$ | |

$TS_1$). Next, it checks $m_2' = m_2$? In case, if it holds, $CS_k$ produces a new timestamp value represented as $TS_2$ and a random secret value denoted as $RS_2$. After that, it computes $m_3 = h(RS_2 \| TC_{CS_k} \| MS_{ES_j-ES_j} \| TS_2) \oplus h(RID_{ES_j} \| MS_{ES_j-CS_k} \| TS_1 \| TS_2)$ and a session key as $SK_{CS_k,ES_j} = h(h(RS_2 \| TC_{CS_k} \| MS_{ES_j-CS_k} \| TS_2) \| h(RS_1 \| TC_{ES_j} \| MS_{ES_j-CS_k} \| TS_1) \| RID_{ES_j} \| MS_{ES_j-CS_k} \| TS_1 \| TS_2)$. Again, it computes $m_4 = h(SK_{CS_k,ES_j} \| RID_{ES_j} \| MS_{ES_j-CS_k} \| TS_2)$ and generates a new temporary identification number for $ES_j$ as $TIN_{ES_k}^{new}$. After that $CS_k$ computes $m_5 = TIN_{ES_j}^{new} \oplus h(RID_{ES_j} \| h(RS_1 \| TC_{ES_j} \| MS_{ES_j-CS_k} \| TS_1) \| TS_2)$. After these many computations, $CS_k$ sends message $msg_2 = \{m_3, m_4, m_5, TS_2\}$ to $ES_j$ through the open channel.

- *AKDEC3:* At the arrival of $msg_2$, $ES_j$ checks condition $|TS_2 - TS_2^*| \leq \Delta T$, where $TS_2^*$ is receiving time of $msg_2$, if it holds then $ES_j$ compute $h(RS_2 \| TC_{CS_k} \| MS_{ES_j-CS_k} \| TS_2) = m_3 \oplus h(RID_{ES_j} \| MS_{ES_j-CS_k} \| TS_1 \| TS_2)$ and the session key $SK_{ES_j,CS_k} = h(h(RS_2 \| TC_{CS_k} \| MS_{ES_j-CS_k} \| TS_2) \| h(RS_1 \| TC_{ES_j} \| MS_{ES_j-CS_k} \| TS_1) \| RID_{ES_j} \| MS_{ES_j-CS_k} \| TS_1 \| TS_2)$. It again computes $m_4' = h(SK_{ES_j,CS_k} \| RID_{ES_j} \| MS_{ES_j-CS_k} \| TS_2)$. Then, it checks if $m_4' = m_4$? If it matches, the computed session key by $ES_j$ is considered to be correct. Further, $ES_j$ computes $TIN_{CS_k}^{new} = m_5 \oplus h(RID_{ES_j} \| h(RS_2 \| TC_{ES_j} \| MS_{ES_j-CS_k} \| TS_1) \| TS_2)$ and updates $TIN_{ES_j}^{new}$ with odd $TIN_{ES_j}$ in its database for future use. Then, it generates another fresh timestamp value as $TS_3$ and computes $m_6 = h(SK_{ES_j,CS_k} \| TS_3)$ and sends message $msg_3 = \{m_6, TS_3\}$ to $CS_k$ via open channel.
- *AKDEC4:* At the arrival of $msg_3$, $CS_k$ checks condition $|TS_3 - TS_3^*| \leq \Delta T$, where $TS_3^*$ is receiving time of $msg_3$, if it holds then $CS_k$ computes $m_6' = h(SK_{ES_j,CS_k} \| TS_3)$ and checks $m_6' = m_6$? If it matches $CS_k$ assumes that $ES_j$ has computed the correct session key. After that, both $ES_j$ and $CS_k$ establish session key $SK_{ES_j,CS_k}$ $(= SK_{CS_k,ES_j})$ for their secure data transmission.

### 4.4. Dynamic device addition phase

In this phase, we provide the facility of addition of a new drone to the network. If we do not provide this phase, a new device (i.e., drone) cannot be added to the network. However, this procedure is essentially needed especially when we do the expansion of the network or the requirements of the users increase even in the case of physical drones capture attack by an adversary. It can be done using the following steps.

- *DDA1: RA* chooses identity for $DE_i^v$ as $ID_{DE_j}^v$, its secret key as $k_{DE_i}^v$ and its secret number as $SN_{DE_i}^v$. *RA* further computes

the pseudo identity of $DE_i^v$ as $RID_{DE_i}^v = h(RID_{RA} \| ID_{DE_i}^v \| k_{RA} \| k_{DE_i}^v \| SN_{DE_i}^v)$. It again computes the temporal credentials value of $DE_i^v$ as $TC_{DE_i}^v = h(RID_{RA} \| ID_{DE_i}^v \| k_{RA} \| k_{DE_i}^v \| SN_{DE_i}^v \| RTS_{DE_i}^v)$, where $RTS_{DE_i}^v$ is the registration timestamp value of $DE_i^v$. It again generates a temporary identity for $DE_i$ as $TID_{DE_i}^v$. Then, the registration information has been stored in the memory of $DE_i^v$.

- *DDA2:* Finally, $DE_i^v$ stores values $\{TID_{DE_i}^v, RID_{DE_i}^v, TC_{DE_i}^v, MS_{DE_i-ES_j}^v, h(.)\}$. Here, it is important to mention that $MS_{DE_i-ES_j}^v$ is the primary secret key of both $DE_i^v$ and $ES_j$, this key distinct for different drones. $RA$ also shares the registration information of $DE_i^v$ with the deployed $ES_j$s in a secure way.

### 4.5. Blockchain implementation phase

During this step, we present the specifics of the blockchain. It is a significant phase of the proposed mechanism. Note that "Elliptic Curve Cryptography (ECC)" encryption is used to encrypt a transaction in a block with the help of the public key $KU_{ES_j}$ of the respective ground station server ($ES_j$) so that only $ES_j$ can decrypt the data using its own private key. In this case, since block verification involves the verification of signature present in a block using the "Elliptic Curve Digital Signature Algorithm (ECDSA)" for signature verification, we have applied the public-key based ECC encryption for protection of transactions (containing the crucial data in case of sensitive applications such as healthcare and military).

The particulars are delineated using the following steps:

- **BIP1:** As discussed earlier, the ground station server $ES_j$ receives information $Inf_{DE_i}$ from a connected drone $DE_i$ through the established session key $SK_{DE_i,ES_j}$ in a secure way. Then $ES_j$ creates a partial block $PBK_{ES_j}$ from the received information $Inf_{DE_i}$. First, $ES_j$ creates its public and private key pairs as $\{KU_{ES_j}, KS_{ES_j}\}$ through some public key cryptographic systems, i.e., Elliptic Curve Cryptography (ECC) algorithm. It then divides $Inf_{DE_i}$ into some transactions say $tr_x = \{tr_1, tr_2, \ldots, tr_x\}$. Further, $ES_j$ encrypts $tr_x$ with its public key $KU_{ES_j}$ to convert them into encrypted transactions, say $TR_x = E_{KU_{ES_j}}(tr_x)$. The partial block contains fields as follows $PBK_{ES_j} = \{OWI_{ES_j}, KU_{ES_j}, TR_x, MT_{root_{ES_j}}\}$, where $OWI_{ES_j}$ is owner $ES_j$'s identity information and $MT_{root_{ES_j}}$ is the Merkle tree root value, which is generated from all transactions. $ES_j$ then sends partial block $PBK_{ES_j}$ to connected cloud server $CS_l$ with the help of the established session key $SK_{ES_j,CS_k}$ in a secure way.
- **BIP2:** After receiving $PBK_{ES_j}$, $CS_l$ makes full block $FBK_{CS_l}$ from it. $FBK_{CS_l}$ contains fields as $FBK_{CS_l} = \{BID_{FBK_{CS_l}}, RN_{FBK_{CS_l}}, TSV_{FBK_{CS_l}}, Hash_{FBK_{CS_l}}, Hash_{FBK_{CS_{l-1}}}, OWI_{ES_j}, KU_{ES_j}, TR_x, MT_{root_{ES_j}}, Sign_{FBK_{CS_l}}\}$, where $BID_{FBK_{CS_l}}$, $RN_{FBK_{CS_l}}$, $TSV_{FBK_{CS_l}}$, $Hash_{FBK_{CS_l}}$, $Hash_{FBK_{CS_{l-1}}}$, and $Sign_{FBK_{CS_l}}$ are the block's ($FBK_{CS_l}$) identity information, a random nonce value, the timestamp, the hash of the current block, the hash of the preceding block, and the block's signature $FBK_{CS_l}$.
- **BIP3:** Upon the completion of this process, $CS_l$ will disseminate $FBK_{CS_l}$ via its peer-to-peer cloud server network. At this juncture, the appointed leader, referred to as $CS_l''$, will initiate a consensus over the just received block. To achieve this purpose, the server ($CS_l''$) may employ the procedures of the standard "practical Byzantine Fault Tolerance (pBFT) method [21]". The block $FBK_{CS_l}$ is incorporated into the blockchain $BCH_{IoD_i}$ at the successful completion of the consensus process. The formed blockchain $BCH_{IoD_i}$ can be considered like a "consortium blockchain". As it contains some private data, however, at the same time some of the data should be available publicly as per the raised requirements.

For the better understanding of the readers, the proposed BAKMM-IoD is also explained through a process flow diagram, which is depicted in Fig. 2. It provides the details of various activities and processes of the proposed scheme. The activities like registration of drone, registration of ground station server, and registration of cloud server are highlighted. After that, there is the execution of authentication and key establishment between the drone and ground station server. Further, there is the execution of key management between the ground station server and cloud server. After that, there is the execution of the blockchain formation phase.

**Remark 1.** Here, we provide the importance of using the blockchain technology instead of using a strong public-key encryption algorithm, like RSA-2048 or others, for storing the encrypted data in a semi-trusted cloud environment. In fact, Mitra et al. [38] interestingly investigated the "impact on blockchain-based artificial intelligence (AI)/machine learning (ML)-enabled big data analytics for cognitive IoT environment". They argued that data poisoning attacks are a serious concern when the data is simply stored in semi-trusted cloud storage in place of the blockchain, because they can significantly impact businesses and organizations, both financially and in terms of their reputation, particularly when the big data analytics rely on corrupted data. Their comprehensive experimental results illustrate the impact of data poisoning attacks on an ML model when data is stored in cloud storage (i.e., outside of blockchain) versus in a blockchain (i.e., without data poisoning). The findings reveal substantial performance improvements in accuracy, recall, precision, and F1-score when the data remain free from poisoning attacks. This is true because the data residing into the blockchain cannot be tampered when the transactions are added into the blockchain through the consensus mechanism. Hence, though the blockchain implementation becomes little more costly as compared to simply putting encrypted data in semi-trusted cloud storage, we certainly have various advantages not only for strengthening the security of the system, but also for improving substantial performance in terms of accuracy, recall, precision, and F1-score in big data analytics.

**Remark 2.** The identity is the original identity information of an entity (i.e., drone, ground station server and cloud server), whereas to make the communication anonymous we have used pseudo identity, due to this mechanism the original identity of an entity is not revealed to the other entities of the network. The temporary identity is used to make the communication anonymous as well as untraceable. The temporary identity information is changed in each session, because in each session we have the provision of use of a new temporary identity. It helps us to achieve the untraceability property for the exchanged data in every session of the communications.

## 5. Security analysis of BAKMM-IoD

In this section, a security analysis of the proposed scheme (BAKMM-IoD) is provided. The BAKMM-IoD has been subjected to an informal security analysis utilizing mathematical concepts, assumptions and proofs. The BAKMM-IoD has been shown to be secure to "replay attacks, man-in-the-middle (MiTM) attacks, impersonation attacks, privileged insider attacks, stolen verifier attacks, physical drone capture attacks, ephemeral secret leakage (ESL) attacks, secret data leakage attacks, and other similar attacks". These findings were obtained after performing formal security analysis.

**Proposition 1.** *The SBBDA-IoD protocol makes it impossible to execute a replay attack.*

**Proof.** Different freshly generated timestamp values are used and then verified at the other recipient's end. The aforementioned timestamp values encompass values like $T_1$, $T_2$, $T_3$, $TS_1$, $TS_2$ and $TS_3$. Successful

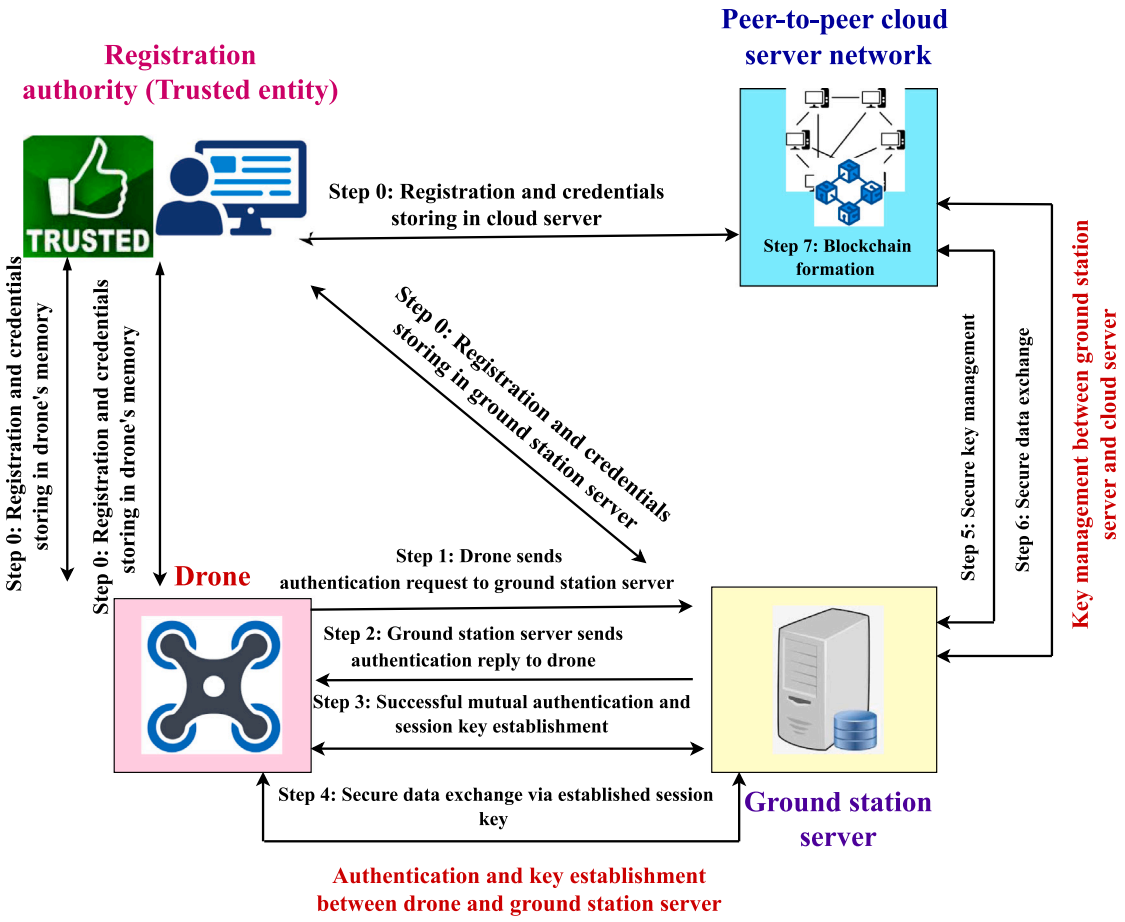


**Fig. 2.** Process flow diagram of the proposed BAKMM-IoD.

completion of the timestamp verification process may result in acceptance of the message by the recipient. Otherwise, it will be returned as undeliverable. By employing condition checking, i.e., $|T_x - T_x^*| \leq \Delta T$, and $|TS_x - TS_x^*| \leq \Delta T$, where $x = 1, 2, 3$, the BAKMM-IoD ensures the prevention of replay attacks. Consequently, the BAKMM-IoD is safeguarded against any replay attacks. □

**Proposition 2.** *The primary objective of the BAKMM-IoD is to prevent man-in-the-middle and impersonation attacks.*

**Proof.** The computation of exchanged messages involves the utilization of several proprietary factors, including $k_{ES_j}$, $SN_{ES_j}$, $MS_{ES_j-CS_k}$, $k_{CS_k}$, $SN_{CS_k}$, $k_{RA}$, $RTS_{CS_k}$, $RTS_{ES_j}$, $MS_{DE_i-ES_j}$, $k_{DE_i}$, and $SN_{DE_i}$. To the attacker $\mathcal{A}$, these discrete values are unknown. Under the present circumstances, it is not feasible for $\mathcal{A}$ to make any changes in the transmitted messages. Another important consideration is that $\mathcal{A}$ is unable to produce completely fresh messages in the correct way. Hence, the BAKMM-IoD offers protection against attacks, like, impersonation tries and man-in-the-middle attempts. □

**Proposition 3.** *The BAKMM-IoD demonstrates robustness in the face of privileged insider attacks.*

**Proof.** The secret values of the entities from the $RA$'s database, namely $k_{ES_j}$, $SN_{ES_j}$, $MS_{ES_j-CS_k}$, $k_{CS_k}$, $SN_{CS_k}$, $RTS_{CS_k}$, $RTS_{ES_j}$, $MS_{DE_i-ES_j}$, $k_{DE_i}$, and $SN_{DE_i}$ have been removed. It may be deduced from this that the authorized user who possesses insider privileges (i.e., $\mathcal{A}$) and who intends to cause harm to the entities (i.e., through a variety of attacks) is not permitted to access the database [42]. As a consequence of this, BAKMM-IoD has afforded protection against privileged insider attacks and other threats of a similar nature. These risks include attempts to impersonation attempts, and illegal session key computations. Therefore, due to its capabilities, the proposal BAKMM-IoD has the potential to reduce the impact of attacks carried out by privileged insiders. □

**Proposition 4.** *The BAKMM-IoD is effectively safeguarded against the stolen verifier attack.*

**Proof.** A segment of the cloud server's database, safeguarded from unauthorized access, contains information related to parameters collected by various entities, including drones and ground station servers. These traits are said to signify the secret information maintained on ground station servers and devices. To ensure that fact, numerous layers of protection have been established. Access to the confidential values of the entities is unattainable for $\mathcal{A}$ due to imposed restrictions [43]. Although this mechanism remains functional, executing an attack on the BAKMM-IoD via the stolen verifier method or other related techniques seem unfeasible. Consequently, the BAKMM-IoD is safeguarded against the stolen verifier attack. □

**Proposition 5.** *The BAKMM-IoD possesses the capacity to prevent the stolen drone attack.*

**Proof.** The suggested implementation of the BAKMM-IoD safeguards sensitive information by ensuring that it is not stored in an unencrypted state within the drones' memory. Moreover, $\mathcal{A}$ should successfully apprehend a drone and subsequently execute an advanced power analysis attack to get critical data from the drone's memory, it would

```
hashfunction h;
const xor:Function;
const cat:Function;
protocol EIOT(DE,ES)
{
 role DE
 {
   fresh rs1:Nonce;
   const TIDDEi,TIDnDEi,RIDDEi,RIDESj,TCDEi,TCESj,MSDEiESj,T1,T2,T3,rs1,rs2;
   var rs2:Nonce;
   macro M1 = xor(h(cat(RIDDEi,MSDEiESj,T1)),h(cat(TCDEi,rs1,MSDEiESj,T1)));
   macro M2 = h(cat(h(cat(TCDEi,rs1,MSDEiESj,T1)),RIDDEi,T1));
   macro SKDEiESj = h(cat(h(cat(TCDEi,rs1,MSDEiESj,T1)),h(cat(RIDESj,TCESj,
   rs2,MSDEiESj,T2)),T1,T2,RIDDEi,MSDEiESj));
   macro M6 = h(cat(SKDEiESj,T3));
   send_1(DE,ES,cat(TIDDEi,M1,M2,T1));
   recv_2(ES,DE,cat(xor(h(cat(RIDDEi,MSDEiESj,T2)),h(cat(RIDESj,TCESj,rs2,
   MSDEiESj,T2))),h(cat(h(cat(h(cat(TCDEi,rs1,MSDEiESj,T1)),h(cat(RIDESj,
   TCESj,rs2,MSDEiESj,T2)),T1,T2,RIDDEi,MSDEiESj)),T1,T2,RIDDEi)),xor(TIDnDEi
   h(cat(h(cat(TCDEi,rs1,MSDEiESj,T1)),RIDDEi,T2))),T2));
   send_3(DE,ES,cat(M6,T3));
   claim_DE1(DE,Secret,(rs1));
   claim_DE2(DE,Secret,(TCDEi));
   claim_DE3(DE,Secret,(MSDEiESj));
   claim_DE4(DE,Niagree);
   claim_DE5(DE,Nisynch);
   claim_DE6(DE,Secret,(SKDEiESj));
   claim_DE7(DE,Weakagree);
   claim_DE8(DE,Alive);}
```

**Fig. 3.** SPDL snippet for the implemented role of DE in BAKMM-IoD.

constitute one of the most perilous scenarios possible [41]. Assuming these conditions were satisfied, $\mathcal{A}$ would possess solely the session key and registration data of this particular drone, lacking access to any other secret information related to the other drones. Each session key within the BAKMM-IoD is unique and exclusive. Every computation is executed using a distinct set of parameters. The deduced session key cannot be utilized to ascertain the session key for additional drones, as such an action is infeasible. This clearly indicates that unauthorized access to the remaining portions of the communication is severely forbidden. As a result, the BAKMM-IoD is protected against the stolen drone attack. □

**Proposition 6.** *The BAKMM-IoD is designed to provide anonymity and untraceability for the exchanged communications.*

**Proof.** No personally identifiable information (i.e., identities of the communicating entities) is sent in plain text within the BAKMM-IoD's architecture. It ensures the safeguarding of the privacy of every individual thus helps us to achieve the anonymity of each entity during the communication. Freshly generated timestamp values (i.e., "$T_1$, $T_2$, $T_3$, $TS_1$, $TS_2$, $TS_3$, and $rs_1$, $rs_2$, $RS_1$, $RS_2$") and randomly produced secret values (i.e., $k_{DE_i}$, $k_{ES_j}$, $k_{CS_k}$) constitute the entirety of the information that is reciprocally shared. It causes the creation of distinct messages for different entities in distinct sessions. Due to this mechanism, the exchanged messages cannot be traced during the communication. Therefore, it can be considered that the proposed BAKMM-IoD achieves anonymity and untraceability properties during the exchange of the messages. □

**Proposition 7.** *The ephemeral secret leakage (ESL) attack is unable to successfully target the BAKMM-IoD under the CK-adversary model.*

**Proof.** The proposed BAKMM-IoD calculates the session key by combining dynamic information, such as random secret numbers, with persistent information, such as secret keys and identities. In BAKMM-IoD, the session keys are computed as $SK_{ES_j,DE_i} = h(h(TC_{DE_i} \| rs_1 \| MS_{DE_i-ES_j} \| T_1) \| h(RID_{ES_j} \| TC_{ES_j} \| rs_2 \| MS_{DE_i-ES_j} \| T_2) \| T_1 \| T_2 \| RID_{DE_i} \| MS_{DE_i-ES_j})$ and $SK_{CS_k,ES_j} = h(h(RS_2 \| TC_{CS_k} \| MS_{ES_j-CS_k} \| TS_2) \| h(RS_1 \| TC_{ES_j} \| MS_{ES_j-CS_k} \| TS_1) \| RID_{ES_j} \| MS_{ES_j-CS_k} \| TS_1 \| TS_2)$. These session keys are computed through the long-term secret parameters consist of the secret keys (i.e., $RID_{DE_i}$, $RID_{ES_j}$, and $RID_{CS_k}$, $k_{ES_j}$, $SN_{ES_j}$, $MS_{ES_j-CS_k}$, $k_{CS_k}$, $SN_{CS_k}$, $RTS_{CS_k}$, $RTS_{ES_j}$, $MS_{DE_i-ES_j}$, $k_{DE_i}$, and $SN_{DE_i}$), and the short-term secret parameter take the form of random secrets (i.e., $rs_1$, $rs_2$, $RS_1$, $RS_2$). This results in the generation of a new session key for a subsequent session. Furthermore, these concealed values are unknown to $\mathcal{A}$. Consequently, it is impractical for $\mathcal{A}$ to precisely ascertain the session key. This indicates that a $\mathcal{A}$ cannot reliably forecast the session key in any measure. Consequently, the BAKMM-IoD demonstrates adequate integrity to endure the ephemeral secret leaking (ESL) attack within the CK-adversary model. □

## 6. Formal security verification of presented BAKMM-IoD

This section presents the formal security verification of the BAKMM-IoD. In the context of the BAKMM-IoD's security, the Scyther tool [44, 45], and [46] has been rigorously employed. The tools, like, ProVerif and AVISPA are somewhat less robust than this one in terms of verifying and analyzing the security of a recently developed security protocol. During its operation, the system utilizes the most advanced cryptographic assumptions. The secret key ensures that an opponent $\mathcal{A}$ will be incapable of decrypting the data unless they themselves possess it. The language employed throughout the implementation phase is "Security Protocol Descriptive Language (SPDL)". A unique role is allocated to each communication party or entity in this particular situation. As a consequence of their roles, the entities undertake several other functions, such as the transmission of messages and the reception of replies. The "send" and "recv" methods facilitate the attainment of these objectives. The scyther tool operates on the DY model, with nine other adversarial models, containing the eCK model and the CK model. The system utilizes tests that facilitate the execution of verifications

```
role ES
{
  fresh rs2:Nonce;
  const TIDDEi,TIDnDEi,RIDDEi,RIDESj,TCDEi,TCESj,MSDEiESj,T1,T2,T3,rs1,rs2;
  var rs1:Nonce;

  macro M3 = xor(h(cat(RIDDEi,MSDEiESj,T2)),h(cat(RIDESj,TCESj,rs2,MSDEiESj,
  T2)));
  macro SKESjDEi = h(cat(h(cat(TCDEi,rs1,MSDEiESj,T1)),h(cat(RIDESj,TCESj,
  rs2,MSDEiESj,T2)),T1,T2,RIDDEi,MSDEiESj));
  macro M4 = h(cat(SKESjDEi,T1,T2,RIDDEi));
  macro M5 = xor(TIDnDEi, h(cat(h(cat(TCDEi,rs1,MSDEiESj,T1)),RIDDEi,T2)));
  recv_1(DE,ES,cat(TIDDEi,xor(h(cat(RIDDEi,MSDEiESj,T1)),h(cat(TCDEi,rs1,
  MSDEiESj,T1))),h(cat(h(cat(TCDEi,rs1,MSDEiESj,T1)),RIDDEi,T1)),T1));
  send_2(ES,DE,cat(M3,M4,M5,T2));
  recv_3(DE,ES,cat(h(cat(h(cat(h(cat(TCDEi,rs1,MSDEiESj,T1)),h(cat(RIDESj,
  TCESj,rs2,MSDEiESj,T2)),T1,T2,RIDDEi,MSDEiESj)),T3)),T3));
  claim_ES1(ES,Secret,(rs2));
  claim_ES2(ES,Secret,(TCESj));
  claim_ES3(ES,Secret,(MSDEiESj));
  claim_ES4(ES,Niagree);
  claim_ES5(ES,Nisynch);
  claim_ES6(ES,Secret,(SKESjDEi));
  claim_ES7(ES,Weakagree);
  claim_ES8(ES,Alive);}}
```

**Fig. 4.** SPDL snippet for the implemented role of ES in BAKMM-IoD.

such as agreement, synchronization, weak agreement, and secrecy.

In the Scyther implementation of a cryptographic protocol, metrics such as agreement, synchronization, and secrecy are crucial. These are critical attributes for assessing the security and integrity of the newly designed protocol. These can be described as follows.

- **Agreement:** It guarantees that two parties (e.g., drone and ground station server) recognize their participation in a session for data communication. They both concur on significant aspects, such as keys, identities, and so forth. It mitigates impersonation or man-in-the-middle (MiTM) attacks by ensuring that both parties are authentically communicating as intended. Additionally, it confirms that the protocol accomplishes mutual authentication.
- **Synchronization:** It guarantees that the sequence of message exchanges occurs as anticipated. Messages cannot be replayed, dropped, or modified. It is crucial for a protocol to attain this property, as it depends on the freshness or sequencing of messages (i.e., for the prevention of replay attacks). Moreover, it confirms that both parties are operating in the same session context.
- **Secrecy:** It guarantees that confidential information, such as session keys or random secret nonces/numbers, remains undisclosed. These values must not be disclosed to any unauthorized individuals. It serves to safeguard against eavesdropping and unauthorized data breach attempts.

To securely validate the "authentication and key establishment phase" of the proposed BAKMM-IoD, we analyze the two critical actions associated with DE (for a drone) and ES (for a ground station server). The importance of these roles is substantial. The SPDL code snippets required for simulating the functions of a drone ($DE_i$) and a ground station server ($ES_j$) are presented in Figs. 3 and 4. Further, Fig. 5, located beneath the claim, status, and comments sections, displays the outcomes of the BAKMM-IoD implementation. The obtained data confirmed that the BAKMM-IoD corresponds with the stated assertions. Thus, the BAKMM-IoD provides protection against numerous possible threats.

## 7. Comparative analysis

In this section, the comparisons and analysis have been done for the BAKMM-IoD and other similar schemes of the domain. The comparisons of the computation costs, communication costs and "security and functionality attributes" have been conducted. The details are provided below. The comparisons of different schemes including Ali et al. [34], Cho et al. [23], Rodrigues et al. [25], Ever [27], Bera et al. [21] and Mishra et al. [35] and the BAKMM-IoD are given.

**Table 6**
Execution time (in milliseconds) under a server.

| Primitive | Max. time (ms) | Min. time (ms) | Average time (ms) |
|---|---|---|---|
| $T_h$ | 0.149 | 0.024 | 0.055 |
| $T_{mtp}$ | 0.199 | 0.092 | 0.114 |
| $T_{ecsigg}$ | 3.147 | 0.308 | 0.729 |
| $T_{ecsigv}$ | 6.147 | 0.593 | 1.405 |
| $T_{senc}$ | 0.008 | 0.002 | 0.003 |
| $T_{sdec}$ | 0.005 | 0.002 | 0.003 |
| $T_{ecm}$ | 2.998 | 0.284 | 0.674 |
| $T_{eca}$ | 0.002 | 0.001 | 0.002 |
| $T_{bp}$ | 7.951 | 4.495 | 4.716 |

**Table 7**
Execution time (in milliseconds) under Raspberry PI 3.

| Primitive | Max. time (ms) | Min. time (ms) | Average time (ms) |
|---|---|---|---|
| $T_h$ | 0.643 | 0.274 | 0.309 |
| $T_{mtp}$ | 0.406 | 0.381 | 0.385 |
| $T_{ecsigg}$ | 5.175 | 2.480 | 2.597 |
| $T_{ecsigv}$ | 9.728 | 4.701 | 4.901 |
| $T_{senc}$ | 0.038 | 0.017 | 0.018 |
| $T_{sdec}$ | 0.054 | 0.009 | 0.014 |
| $T_{ecm}$ | 4.532 | 2.206 | 2.288 |
| $T_{eca}$ | 0.021 | 0.015 | 0.016 |
| $T_{bp}$ | 32.79 | 27.606 | 32.084 |

We have taken the results of MIRACL library [21], in which various values of execution time (i.e., computation time) are given. The execution time (in milliseconds) values for a server are given in Table 6. Further, the execution time (in milliseconds) values under Raspberry PI 3 for a device (i.e., smart IoT device, drones) are given in 7. Here it is important to mention that the donations $T_h$, $T_{senc}/T_{sdec}$, $T_{bp}$, $T_{fe}$, $T_{eca}$, $T_{ecm}$, $T_{ecsigg}$, $T_{ecsigv}$, and $T_{mtp}$ are taken for the time needed for the execution a "one-way cryptographic hash function", a "symmetric key encryption/decryption (AES-128)", a "bilinear pairing", a "fuzzy extractor", an "elliptic curve point addition", an "elliptic curve point multiplication", a "ECDSA generation", "ECDSA verification", and a "map to point", respectively. It is considered that $T_{fe}$ ($\approx T_{ecm}$) [47].

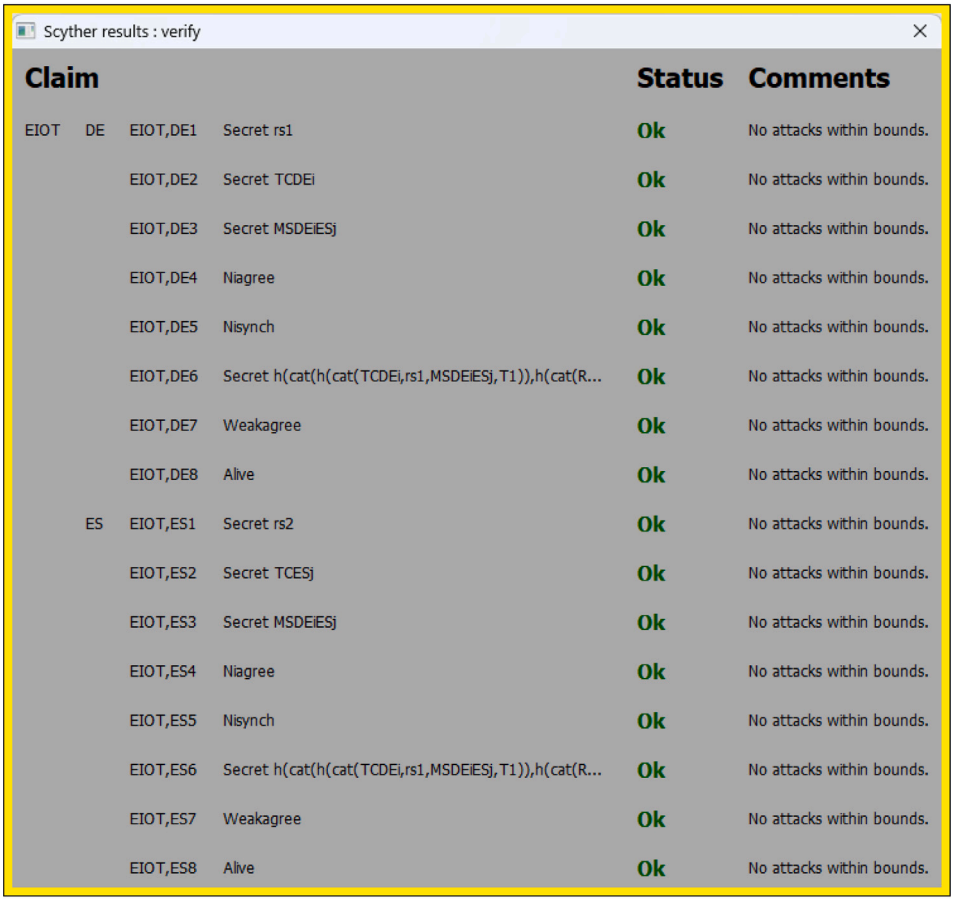


**Fig. 5.** Results of security verification using scyther tool.

### 7.1. Comparison of computation costs

For computation costs assessment, $T_h$, $T_{senc}/T_{sdec}$, $T_{bp}$, $T_{fe}$, $T_{eca}$, $T_{ecm}$, $T_{ecsigg}$, $T_{ecsigv}$, and $T_{mtp}$ are used to signify for the time needed to execute a "one-way cryptographic hash function", a "symmetric key encryption/decryption (AES-128)", a "bilinear pairing", a "fuzzy extractor", an "elliptic curve point addition", an "elliptic curve point multiplication", a "ECDSA generation", "ECDSA verification", and a "map to point", respectively. It is assumed that $T_{fe} \approx T_{ecm}$ [47].

The computation cost values are calculated on the basis of values given in Tables 6 and 7. The computation cost values for the BAKMM-IoD are calculated $8T_h \approx 2.47$ ms (for drone) and $8T_h \approx 0.44$ ms for (ground satiation server). From Table 8, it is clear that the BAKMM-IoD has less computation costs than the other compared schemes, i.e., the schemes of Cho et al. [23], Rodrigues et al. [25], Ever [27], and Algarni and Jan [36], whereas it is very similar to the scheme of Ali et al. [34] and Mishra et al. [35].

**Table 8**
Comparing different computation costs.

| Scheme | Smart device/Drone | GSS/Server |
|---|---|---|
| Ali et al. [34] | $18T_h + T_{fe} + T_{senc}$ $\approx 7.868$ ms | $7T_h + 3T_{senc}/T_{sdec}$ $\approx 0.394$ ms |
| Cho et al. [23] | $2T_{ecsigv} + T_{sdec}$ $+10001T_h$ $\approx 3100.125$ ms | $2T_{ecsigg} + T_{senc}$ $+10001T_h$ $\approx 551.516$ ms |
| Rodrigues et al. [25] | $9T_h + 6T_{ecm}$ $\approx 16.509$ ms | $9T_h + 2T_{ecm}$ $\approx 1.843$ ms |
| Ever [27] | $9T_h + 2T_{bp}+$ $2T_{mtp} + 3T_{ecm}$ $\approx 74.583$ ms | $6T_h + 3T_{bp}+$ $2T_{mtp} + 3T_{ecm}$ $\approx 16.728$ ms |
| Bera et al. [21] | $9T_h + 2T_{senc}/T_{sdec}$ $+2T_{ecm} + T_{eca}$ $\approx 7.405$ ms | $9T_h + 2T_{senc}/T_{sdec}$ $2T_{ecm} + T_{eca}$ $\approx 1.851$ ms |
| Mishra et al. [35] | $9T_h$ $\approx 2.78$ ms | $7T_h$ $\approx 0.39$ ms |
| Algarni and Jan [36] | $T_{fe} + 14T_h$ $\approx 6.614$ ms | $6T_h$ $\approx 0.33$ ms |
| BAKMM-IoD | $8T_h$ $\approx 2.47$ ms | $8T_h$ $\approx 0.44$ ms |

### 7.2. Comparison of communication costs

To compute the communication expenses, we have presumed the terms "identity", "random number", and "elliptic curve point $P = (P_x, P_y) \in E_q(a, b)$", where the coordinates of $P$ are denoted as $P_x$ and $P_y$, hash output, generated using the SHA-256 hashing algorithm, and the timestamp are 160 bits, 160 bits, (160 + 160) = 320 bits, 256 bits, and 32 bits, respectively. We subsequently calculate communication costs in terms of the bit count necessary for transmitting messages $MSG_1$, $MSG_2$, and $MSG_3$.

In the authentication and key establishment process of drone $DE_i$ and the $ES_j$ three messages are exchanged, which are $MSG_1 = \{TID_{DE_i}, M_1, M_2, T_1\}$, $MSG_2 = \{M_3, M_4, M_5, T_2\}$, $MSG_3 = \{M_6, T_3\}$. If we calculate the sizes of these messages, this is estimated as $|MSG_1| = 160+256+256+32 = 704$ bits, $|MSG_2| = 256+256+256+32 = 800$ bits, and $|MSG_3| = 256+32 = 2880$ bits, as a whole the communication of the BAKMM-IoD becomes 704+ 800+ 288 = 1782 bits. The communication expenses of different schemes are presented in Table 9. The data in Table 9 indicates that the communication cost of the BAKMM-IoD is lower than that of the other examined schemes.

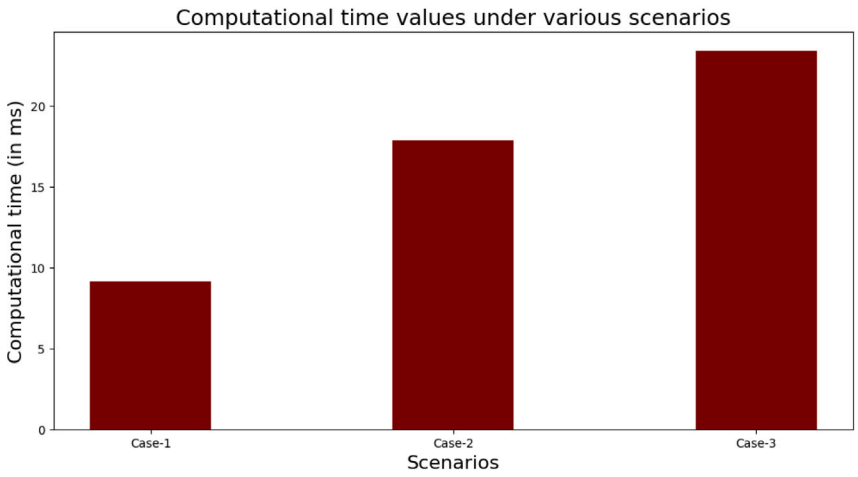


**Fig. 6.** Results of implementation of blockchain for the proposed BAKMM-IoD: effect on computational time.

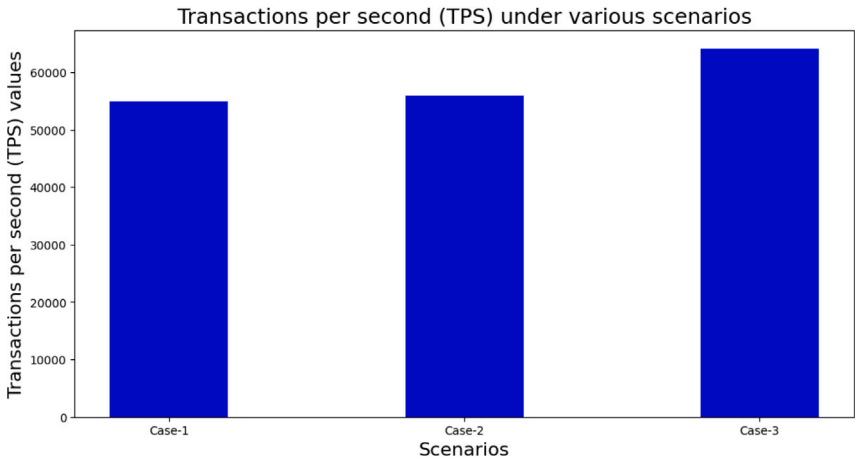


**Fig. 7.** Results of implementation of blockchain for the proposed BAKMM-IoD: effect on transactions per second (TPS).

**Table 9**
Comparative study on communication costs.

| Scheme | No. of messages | Total cost (in bits) |
|---|---|---|
| Ali et al. [34] | 3 | 3424 |
| Cho et al. [23] | 3 | 3968 |
| Rodrigues et al. [25] | 4 | 3456 |
| Ever [27] | 6 | 5344 |
| Bera et al. [21] | 3 | 2368 |
| Mishra et al. [35] | 3 | 1792 |
| Algarni and Jan [36] | 4 | 2784 |
| BAKMM-IoD | 3 | 1792 |

### *7.3. Comparison of security and functionality attributes*

The juxtaposition of security and functionality attributes is presented in Table 10. Based on the comparison, it is evident that the BAKMM-IoD offers superior security and additional functional features compared to the other schemes given by Ali et al. [34], Cho et al. [23], Rodrigues et al. [25], Ever [27], Bera et al. [21], Mishra et al. [35], and Algarni and Jan [36].

## 8. Practical implementation of BAKMM-IoD: blockchain simulation

The implementation of presented BAKMM-IoD is given here [48]. The details of the parameters that were used in the experimentation are described in Table 11. During the experimentation and validation, three distinct scenarios or cases (case-1, case-2 and case-3) were tested and compared. This experiment was conducted on a Windows 64-bit 11 OS with an Intel(R) Core i5-8250U processor, running at up to 1800 MHz and 8 GB RAM. Open source Visual Studio Code of version 1.93 with Java was used for programming environment. For case-1, the drone deployment was 50, for case-2, drone deployment was 100 and for case-3, it was 150. The five blocks in case-1, ten blocks in case-2 and fifteen blocks in case-3 were computed as well as committed. Four miner nodes (i.e., cloud servers over P2PCS network) were used concurrently. It was deployed, for 10 ground station servers in case-1, 20 in case-2, and 30 in case-3. The voting-based method is followed for making consensus in association with the practical byzantine fault tolerance (pBFT) in blockchain mining work. Such details of the current flow of the transactions are covered under the blanket of the encrypted transaction. For example, the entity (communicating party) by which the information is transmitted, or the underlying logic. The cipher-text of each such transaction depends on elliptic curve cryptography (ECC) algorithm. It could be said that the amount of additional bits necessary to encode the data in the way described is equal to 640 bits which is $(320 + 320)$ bits. Encryption is done in every block to assess transactions worth 100. The results following the simulations were determined as such.

There are other critical applications, where the data is strictly confidential and private. Consider the healthcare applications using the drones. Unmanned aerial vehicle (UAV) technology has greatly enriched the healthcare sector, making substantial contributions [49]. As a result, drones are emerging as one of the fastest-growing technologies in the healthcare industry, offering a diverse array of applications.

**Table 10**
Comparison of security and functionality features.

| Feature ($F$) | Ali et al. [34] | Cho et al. [23] | Rodrigues et al. [25] | Ever [27] | Bera et al. [21] | Mishra et al. [35] | Algarni and Jan [36] | BAKMM-IoD |
|---|---|---|---|---|---|---|---|---|
| $ASFF_1$ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| $ASFF_2$ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| $ASFF_3$ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| $ASFF_4$ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| $ASFF_5$ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| $ASFF_6$ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| $ASFF_7$ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| $ASFF_8$ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| $ASFF_9$ | × | × | × | × | ✓ | ✓ | ✓ | × |
| $ASFF_{10}$ | × | × | × | × | ✓ | ✓ | × | ✓ |
| $ASFF_{11}$ | ✓ | × | × | × | ✓ | ✓ | × | ✓ |
| $ASFF_{12}$ | × | × | × | × | ✓ | × | × | ✓ |
| $ASFF_{13}$ | × | × | ✓ | × | ✓ | ✓ | × | ✓ |
| $ASFF_{14}$ | × | × | × | × | × | × | × | ✓ |

$ASFF_1$: "protection for replay attack"; $ASFF_2$: "protection for man-in-the-middle attack"; $ASFF_3$: "availability of mutual authentication"; $ASFF_4$: "availability of key agreement"; $ASFF_5$: "protection for device/drone impersonation attack"; $ASFF_6$: "protection for GSS/server impersonation attack"; $ASFF_7$: "protection for malicious device deployment attack"; $ASFF_8$: "protection for drone/device physical capture attack"; $ASFF_9$: "formal security verification using AVISPA/Scyhter tool"; $ASFF_{10}$: "protection for ESL attack under the CK-adversary model"; $ASFF_{11}$: "availability of dynamic drone/device addition phase"; $ASFF_{12}$: "implementation of blockchain"; $ASFF_{13}$: "availability of anonymity and untraceability properties"; $ASFF_{14}$: "availability of mechanism for secure communication of ground station server and cloud server".
✓: "a scheme is secure or it supports an attribute"; ×: "a scheme is insecure or it does not support an attribute".

**Table 11**
Simulation parameters and their values used in BAKMM-IoD.

| Parameter | Value |
|---|---|
| Platform used | Windows 11 64 bit OS |
| Processor | Intel (R) core (TM), i5-8250U, 1600 MHz–1800 MHz |
| RAM size | 8 GB |
| Programming platform | Visual studio code v1.93 with Java |
| Quantity of deployed drones | 50 (case-1), 100 (case-2), 150 (case-3) |
| Quantity of ground station server | 10 (case-1), 20 (case-2), 30 (case-3) |
| Quantity of miner nodes over P2P CS network | 4 in all cases |

These applications include real-time data collection, patient monitoring, improved quality of care, and drug transportation. Hospitals are increasingly using drones to deliver medical supplies to remote and rural areas. Additionally, medical professionals are finding that drones can enhance the accuracy of disease diagnoses. This technology has the potential to tackle some of the most pressing healthcare challenges, such as providing medical assistance during disasters and transporting organs for transplantation.

Consider another sensitive application using the drones for battlefield or military [50], where the data is also private and confidential. The increasing adoption of UAVs in the defense and security sectors for various purposes – including surveying, mapping, transportation, combat operations, and monitoring – is anticipated to drive demand for military UAVs in the coming years. Additionally, the rise in defense budgets across multiple countries aimed at acquiring modern and technologically advanced military drones is expected to contribute to the growth of the global market.

For the simplicity of the implementation, the information which is used in the creation of the blocks, i.e., for the transactions field are like, "current temperature value for a particular location of region", "current humidity level for a particular location of a region". Likewise, we have used various information in the transactions fields of a block. All these information are sent by the drones to the connected ground station servers in a secure way with the help of the deployed "authentication and key establishment phase". After that the ground station server creates partial block from this information by putting this information in the transaction field of the partial block. The transactions are encrypted (i.e., via Elliptic Curve Cryptography (ECC)-based encryption algorithm) since we need to provide the secrecy to the data. Please refer to the information given in Section 4.5.

### 8.1. Effect on computational time

The computation time values (in ms) were assessed to evaluate the effect of a rising number of drones and ground station servers in each scenario examined. The estimated computational times for case-1, case-2, and case-3 are 9.12 ms, 17.88 ms, and 23.43 ms, respectively. The outcomes are also depicted in Fig. 6. The computational time escalates with the growth in the number of drones and ground station servers from case-1 to case-2 and from case-2 to case-3 due to the rise in the number of drones and ground station servers result in the generation and incorporation of additional blocks (creation and mining) in the blockchain.

### 8.2. Effect on transactions per second (TPS)

The effect of BAKMM-IoD on transactions per second (TPS) in the examined situations is measured. The transactions per second (TPS) values are 54825, 55928 and 64103 for case-1, case-2 and case-3, respectively. The supplementary findings are depicted in Fig. 7. The transactions value per second (TPS) on the blockchain escalates with the augmentation of drones and ground station servers. This is the result of the production and incorporation (mining) of further blocks entries to the blockchain.

## 9. Conclusions

Security solutions are essential for safeguarding the data and devices, such as drones and servers, within IoD networks. A reliable blockchain-enabled authentication and key management mechanism for various IoD applications (BAKMM-IoD) was introduced. BAKMM-IoD has been demonstrated to be secure against numerous potential threats through comprehensive security study and formal verification with the widely recognized Scyther tool. BAKMM-IoD outperforms other comparable current mechanisms regarding communication cost, calculation cost, and attributes of security and functionality. At the end, a practical implementation of BAKMM-IoD is subsequently shown to illustrate its applicability in real-world scenarios and highlight its effect on key performance metrics.

In the future, we intend to provide machine learning/deep learning-based big data analytics phase in the presented scheme for the real-time data analysis of the received data. We have plan to provide a testbed implementation for the presented scheme. The post-quantum cryptography (PQC)-based security primitives can also be incorporated in the design of the presented scheme to make it more secure especially for the era of quantum cryptography.

## CRediT authorship contribution statement

**Mohammad Wazid:** Writing – original draft, Resources, Methodology, Formal analysis, Data curation, Conceptualization. **Saksham Mittal:** Visualization, Software, Resources, Data curation. **Ashok Kumar Das:** Writing – review & editing, Validation, Investigation, Conceptualization. **SK Hafizul Islam:** Validation, Methodology, Investigation, Formal analysis. **Mohammed J.F. Alenazi:** Resources, Project administration, Investigation, Funding acquisition. **Athanasios V. Vasilakos:** Visualization, Project administration, Investigation, Funding acquisition.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Acknowledgments

The authors extend their appreciation to Researcher Supporting Project number (RSPD2025R582), King Saud University, Riyadh, Saudi Arabia. The authors would also like to thank the anonymous reviewers and associate editor for their valuable feedback on the paper.

## Data availability

No data was used for the research described in the article.

## References

[1] C. Lin, D. He, N. Kumar, K.-K.R. Choo, A. Vinel, X. Huang, Security and privacy for the internet of drones: Challenges and solutions, IEEE Commun. Mag. 56 (1) (2018) 64–69.
[2] C. Singh, R. Mishra, H.P. Gupta, P. Kumari, The internet of drones in precision agriculture: Challenges, solutions, and research opportunities, IEEE Internet Things Mag. 5 (1) (2022) 180–184.
[3] M.P. Singh, G.S. Aujla, R.S. Bali, Blockchain for the internet of drones: Applications, challenges, and future directions, IEEE Internet Things Mag. 4 (4) (2021) 47–53.
[4] Z. Lv, Y. Li, J. Wu, H. Lv, Securing the internet of drones against cyber-physical attacks, IEEE Internet Things Mag. 4 (4) (2021) 74–78.
[5] A. Derhab, O. Cheikhrouhou, A. Allouch, A. Koubaa, B. Qureshi, M.A. Ferrag, L. Maglaras, F.A. Khan, Internet of drones security: Taxonomies, open issues, and future directions, Veh. Commun. 39 (2023) 100552.
[6] W. Yang, S. Wang, X. Yin, X. Wang, J. Hu, A review on security issues and solutions of the internet of drones, IEEE Open J. Comput. Soc. 3 (2022) 96–110.
[7] C. Badii, P. Bellini, A. Difino, P. Nesi, Smart city IoT platform respecting GDPR privacy and security aspects, IEEE Access 8 (2020) 23601–23623.
[8] N. Azam, L. Michala, S. Ansari, N.B. Truong, Data privacy threat modelling for autonomous systems: A survey from the GDPR's perspective, IEEE Trans. Big Data 9 (2) (2023) 388–414.
[9] C. Li, B. Palanisamy, Privacy in Internet of Things: From principles to technologies, IEEE Internet Things J. 6 (1) (2019) 488–505.
[10] P.-Y. Kong, A survey of cyberattack countermeasures for unmanned aerial vehicles, IEEE Access 9 (2021) 148244–148263.
[11] G.N. Nguyen, N.H.L. Viet, M. Elhoseny, K. Shankar, B. Gupta, A.A.A. El-Latif, Secure blockchain enabled cyber–physical systems in healthcare using deep belief network with ResNet model, J. Parallel Distrib. Comput. 153 (2021) 150–160.
[12] A. Raj, S. Prakash, A privacy-preserving authentic healthcare monitoring system using blockchain, Int. J. Softw. Sci. Comput. Intell. 14 (2022) 1–23.
[13] Y. Xu, Z. Peng, C. Zhang, G. Wang, H. Wang, H. Jiang, Y. Zhang, Enhancing privacy in cyber-physical systems: An efficient blockchain-assisted data-sharing scheme with deniability, J. Syst. Archit. 150 (2024) 103132.
[14] Y. Zhang, L. Xiong, F. Li, X. Niu, H. Wu, A blockchain-based privacy-preserving auditable authentication scheme with hierarchical access control for mobile cloud computing, J. Syst. Archit. 142 (2023) 102949.
[15] C.-M. Chen, S. Liu, X. Li, S.H. Islam, A.K. Das, A provably-secure authenticated key agreement protocol for remote patient monitoring IoMT, J. Syst. Archit. 136 (2023) 102831.
[16] A. Shahidinejad, J. Abawajy, S. Huda, Untraceable blockchain-assisted authentication and key exchange in medical consortiums, J. Syst. Archit. 151 (2024) 103143.
[17] Y. Li, An improved lightweight and privacy preserving authentication scheme for smart grid communication, J. Syst. Archit. 152 (2024) 103176.
[18] G. Thakur, S. Prajapat, P. Kumar, C.-M. Chen, A privacy-preserving three-factor authentication system for IoT-enabled wireless sensor networks, J. Syst. Archit. 154 (2024) 103245.
[19] Y. Yao, H. Chen, K. Wang, H. Yu, Y. Wang, Q. Wang, Efficient iNTRU-based public key authentication keyword searchable encryption in cloud computing, J. Syst. Archit. 154 (2024) 103231.
[20] A. Yazdinejad, R.M. Parizi, A. Dehghantanha, H. Karimipour, G. Srivastava, M. Aledhari, Enabling drones in the Internet of Things with decentralized blockchain-based security, IEEE Internet Things J. 8 (8) (2021) 6406–6415.
[21] B. Bera, A.K. Das, A.K. Sutrala, Private blockchain-based access control mechanism for unauthorized UAV detection and mitigation in internet of drones environment, Comput. Commun. 166 (2021) 91–109.
[22] C. Feng, B. Liu, Z. Guo, K. Yu, Z. Qin, K.-K.R. Choo, Blockchain-based cross-domain authentication for intelligent 5G-enabled internet of drones, IEEE Internet Things J. 9 (8) (2022) 6224–6238.
[23] G. Cho, J. Cho, S. Hyun, H. Kim, SENTINEL: A secure and efficient authentication framework for unmanned aerial vehicles, Appl. Sci. 10 (9) (2020).
[24] R. Gupta, P. Bhattacharya, S. Tanwar, N. Kumar, S. Zeadally, GaRuDa: A blockchain-based delivery scheme using drones for healthcare 5.0 applications, IEEE Internet Things Mag. 4 (4) (2021) 60–66.
[25] M. Rodrigues, J. Amaro, F.S. Osorio, B. Kalinka. R. L. J. C., Authentication methods for UAV communication, in: 2019 IEEE Symposium on Computers and Communications, ISCC, 2019, pp. 1210–1215, http://dx.doi.org/10.1109/ISCC47284.2019.8969732.
[26] M. Farash, M. Turkanovic, S. Kumari, M. Holbl, An efficient user authentication and key agreement scheme for heterogeneous wireless sensor network tailored for the Internet of Things environment, Ad Hoc Netw. 36 (2016) 152–176.
[27] Y. Kirsal Ever, A secure authentication scheme framework for mobile-sinks used in the internet of drones applications, Comput. Commun. 155 (2020) 143–149.
[28] M.P. Singh, G.S. Aujla, R.S. Bali, Blockchain for the internet of drones: Applications, challenges, and future directions, IEEE Internet Things Mag. 4 (4) (2021) 47–53.
[29] R. Xiong, Q. Xiao, Z. Wang, Z. Xu, F. Shan, Leveraging lightweight blockchain for secure collaborative computing in UAV Ad-Hoc Networks, Comput. Netw. 251 (2024) 110612.
[30] W. Wang, Z. Han, T.R. Gadekallu, S. Raza, J. Tanveer, C. Su, Lightweight blockchain-enhanced mutual authentication protocol for UAVs, IEEE Internet Things J. 11 (6) (2024) 9547–9557.
[31] W. Wang, Y. Yang, Z. Yin, K. Dev, X. Zhou, X. Li, N.M.F. Qureshi, C. Su, BSIF: Blockchain-based secure, interactive, and fair mobile crowdsensing, IEEE J. Sel. Areas Commun. 40 (12) (2022) 3452–3469.
[32] X. Yu, Y. Xie, Q. Xu, Z. Xu, R. Xiong, Secure data sharing for cross-domain industrial IoT based on consortium blockchain, in: 26$^{th}$ IEEE International Conference on Computer Supported Cooperative Work in Design, CSCWD, Rio de Janeiro, Brazil, 2023, pp. 1508–1513, http://dx.doi.org/10.1109/CSCWD57460.2023.10152584.
[33] J. Srinivas, A.K. Das, N. Kumar, J.J.P.C. Rodrigues, TCALAS: Temporal credential-based anonymous lightweight authentication scheme for internet of drones environment, IEEE Trans. Veh. Technol. 68 (7) (2019) 6903–6916.
[34] Z. Ali, S.A. Chaudhry, M.S. Ramzan, F. Al-Turjman, Securing smart city surveillance: A lightweight authentication mechanism for unmanned vehicles, IEEE Access 8 (2020) 43711–43724.
[35] A.K. Mishra, M. Wazid, D.P. Singh, A.K. Das, J. Singh, A.V. Vasilakos, Secure blockchain-enabled authentication key management framework with big data analytics for drones in networks beyond 5G applications, Drones 7 (8) (2023).
[36] F. Algarni, S.U. Jan, PSLAPS-IoD: A provable secure and lightweight authentication protocol for securing internet-of-drones (IoD) environment, IEEE Access 12 (2024) 45948–45960, http://dx.doi.org/10.1109/ACCESS.2024.3382579.
[37] K.A. Tychola, K. Voulgaridis, T. Lagkas, Beyond flight: Enhancing the internet of drones with blockchain technologies, Drones 8 (6) (2024) URL https://www.mdpi.com/2504-446X/8/6/219.
[38] A. Mitra, B. Bera, A.K. Das, S.S. Jamal, I. You, Impact on blockchain-based AI/ML-enabled big data analytics for cognitive Internet of Things environment, Comput. Commun. 197 (2023) 173–185.
[39] D. Dolev, A. Yao, On the security of public key protocols, IEEE Trans. Inform. Theory 29 (2) (1983) 198–208.
[40] R. Canetti, H. Krawczyk, Universally composable notions of key exchange and secure channels, in: International Conference on the Theory and Applications of Cryptographic Techniques– Advances in Cryptology, EUROCRYPT 2002, Amsterdam, The Netherlands, 2002, pp. 337–351.
[41] T.S. Messerges, E.A. Dabbish, R.H. Sloan, Examining smart-card security under the threat of power analysis attacks, IEEE Trans. Comput. 51 (5) (2002) 541–552.
[42] M. Wazid, A.K. Das, N. Kumar, M. Alazab, Designing authenticated key management scheme in 6G-enabled network in a box deployed for industrial applications, IEEE Trans. Ind. Inf. 17 (10) (2021) 7174–7184.
[43] M. Wazid, B. Bera, A.K. Das, S.P. Mohanty, M. Jo, Fortifying smart transportation security through public blockchain, IEEE Internet Things J. 9 (17) (2022) 16532–16545.

[44] B. Khadem, A.M. Suteh, M. Ahmad, A. Alkhayyat, M.S. Farash, H.S. Khalifa, An improved WBSN key-agreement protocol based on static parameters and hash functions, IEEE Access 9 (2021) 78463–78473.
[45] C.J.F. Cremers, Scyther : Semantics and verification of security protocols, 2006, https://pure.tue.nl/ws/files/2425555/200612074.pdf (Accessed on August 2024).
[46] M. Tanveer, A.H. Zahid, M. Ahmad, A. Baz, H. Alhakami, LAKE-IoD: Lightweight authenticated key exchange protocol for the internet of drone environment, IEEE Access 8 (2020) 155645–155659.
[47] D. He, S. Zeadally, B. Xu, X. Huang, An efficient identity-based conditional privacy-preserving authentication scheme for vehicular ad hoc networks, IEEE Trans. Inf. Forensics Secur. 10 (12) (2015) 2681–2691.
[48] M. Fan, X. Zhang, Consortium blockchain based data aggregation and regulation mechanism for smart grid, IEEE Access 7 (2019) 35929–35940.
[49] Drones in healthcare: A lifesaving innovation, 2024, Available at: https://www.indowings.com/blog/5-reasons-why-we-need-to-use-drones-in-the-hospital-management.php. (Accessed on October 2024).
[50] Military drone market, 2023, https://www.fortunebusinessinsights.com/military-drone-market-102181. (Accessed on October 2024).

**Mohammad Wazid** received his Master of Technology in Computer Network Engineering from Graphic Era University, Dehradun, India, and received a Ph.D. in Computer Science and Engineering from the International Institute of Information Technology, Hyderabad, India. He is currently working as a Professor in the Department of Computer Science and Engineering, Graphic Era University, Dehradun, India. He is the head of the cybersecurity and IoT research group at Graphic Era University, Dehradun, India. Prior to this, he was an assistant professor in the Department of Computer Science and Engineering at the Manipal Institute of Technology, MAHE, Manipal, India. He was also a post-doctoral researcher in the cyber security and networks lab, Innopolis University, Innopolis, Russia. His current research interests include security, remote user authentication, the Internet of Things (IIoT), and cloud computing. He has published more than 100 papers in international journals and conferences in the above areas. He was a recipient of the University Gold Medal and the Young Scientist Award from UCOST, the Department of Science and Technology, Government of Uttarakhand, India. He is a senior member of IEEE.

**Saksham Mittal** is pursuing Ph.D. CSE in the department of CSE at Graphic Era Deemed to be University, Dehradun, India. He is also associated with Graphic Era Hill University, Dehradun, India as the teaching staff. His research interests include intrusion detection systems, big data analytics, threat analysis, and machine learning.

**Ashok Kumar Das**, received a Ph.D. degree in computer science and engineering, an M.Tech. degree in computer science and data processing, and an M.Sc. degree in mathematics from IIT Kharagpur, India. He is currently a full Professor with the Center for Security, Theory and Algorithmic Research, IIIT, Hyderabad, India. He is an adjunct professor at the Korea University, Seoul, South Korea. He was also a visiting research professor with the Virginia Modeling, Analysis and Simulation Center, Old Dominion University, Suffolk, p=VA 23435, USA. His research interests include cryptography, system and network security, blockchain, security in the Internet of Things (IoT), Internet of Vehicles (IoV), Internet of Drones (IoD), smart grids, smart city, cloud/fog computing, intrusion detection, AI/ML security, and post-quantum cryptography. He has authored over 465 papers in international journals and conferences in the above areas, including over 395 reputed journal papers. He was a recipient of the Institute Silver Medal from IIT Kharagpur. He has been listed in the Web of Science (Clarivate™) Highly Cited Researcher 2022 and 2023 in recognition of his exceptional research performance. He is/was on the editorial board of IEEE Transactions on Information Forensics and Security, IEEE Systems Journal, Journal of Network and Computer Applications (Elsevier), Computer Communications (Elsevier), International Journal of Communication Systems (Wiley), Journal of Cloud Computing (Springer), Cyber Security and Applications (Elsevier), Alexandria Engineering Journal (Elsevier), IET Communications, KSII Transactions on Internet and Information Systems, and International Journal of Communication Systems (Wiley). He also served as one of the Technical Program Committee Chairs of the first International Congress on Blockchain and Applications (BLOCKCHAIN'19), Avila, Spain, June 2019, International Conference on Applied Soft Computing and Communication Networks (ACN'20), October 2020, Chennai, India, second International Congress on Blockchain and Applications (BLOCKCHAIN'20), L'Aquila, Italy, October 2020, and International Conference on Applied Soft Computing and Communication Networks (ACN'23), December 2023, Bangalore, India. His Google Scholar h-index is 92 and i10-index is 302 with over 25,200 citations.

**SK Hafizul Islam** received the M.Sc. degree in applied mathematics from Vidyasagar University, Midnapore, India, in 2006, and the M.Tech. degree in Computer Application and the Ph.D. degree in Computer Science and Engineering in 2009 and 2013, respectively, from Indian Institute of Technology [IIT (ISM)] Dhanbad, Jharkhand, India, under the INSPIRE Fellowship Ph.D. Program (funded by the Department of Science and Technology, Government of India). He is currently an Assistant Professor in the Department of Computer Science and Engineering, Indian Institute of Information Technology Kalyani (IIIT Kalyani), West Bengal, India. He has more than ten years of teaching and thirteen years of research experience. He has authored or co-authored 150 research papers in journals and conference proceedings of international reputes. His research interests include Cryptography, Information Security, Neural Cryptography, Lattice-based Cryptography, IoT & Blockchain Security, and Deep Learning. He has edited four books for the publishers Scrivener-Wiley, Elsevier, and CRC Press. He is an Associate Editor for IEEE Transactions on Intelligent Transportation Systems, IEEE Access, International Journal of Communication Systems (Wiley), Telecommunication Systems (Springer), IET Wireless Sensor Systems, Security and Privacy (Wiley), and Array - Journal (Elsevier). He is a senior member of IEEE, and a member of ACM.

**Mohammed J.F. Alenazi** earned his B.S., M.S., and Ph.D. degrees in computer engineering from the University of Kansas, USA, in 2010, 2012, and 2015, respectively. He is a Professor in computer engineering at King Saud University and a reviewer for several international journals. His research interests span cybersecurity, focusing on network security, encryption, and vulnerability analysis, as well as machine learning, where he applies AI to enhance network security and performance. He also works on the design and analysis of resilient networks, network routing, and mobile ad hoc network (MANET) protocols. A member of ACM, his work contributes to the intersection of cybersecurity and machine learning for developing adaptive, threat-resistant systems.

**Athanasios V. Vasilakos** is with the Center for AI Research (CAIR), University of Agder (UiA), Grimstad, Norway. He is WoS Highly Cited Researcher (HC), from 2016 to 2021. He served or is serving as an Editor for many technical journals, such as the IEEE TRANSACTIONS ON NETWORK AND SERVICE MANAGEMENT, IEEE TRANSACTIONS ON CLOUD COMPUTING, IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, IEEE TRANSACTIONS ON CYBERNETICS, IEEE TRANSACTIONS ON NANOBIOSCIENCE, IEEE TRANSACTIONS ON INFORMATION TECHNOLOGY IN BIOMEDICINE, ACM Transactions on Autonomous and Adaptive Systems, and the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS.