2025 IEEE 31th International Conference on Parallel and Distributed Systems (ICPADS) | 979-8-3315-4901-5/25/$31.00 ©2025 IEEE | DOI: 10.1109/ICPADS67057.2025.11323018

A Secure Anonymous Authentication and Key Agreement Scheme for UAV Swarms in Emergency Rescue Environments

Fengqun Wang, Hang Hai, Jie Cui*, Wuquan Wen, Qingyang Zhang and Hong Zhong
School of Computer Science and Technology, Anhui University, Hefei, China
Anhui Engineering Research Center for IoT Security Technologies, Anhui University, Hefei, China
*Corresponding Author: cuijie@mail.ustc.edu.cn

***Abstract*—To achieve secure communication in unmanned aerial vehicle (UAV) swarms during emergency rescue operations, A wide range of authentication key agreement (AKA) schemes has emerged in recent research. However, these schemes generally face three critical limitations. First, UAVs may struggle to maintain persistent communication with the trusted authority in complex environments, and efficient authentication cannot be guaranteed when the TA is offline. Second, most existing schemes lack traceability, preventing the TA from revealing the true identity of malicious UAVs. Finally, UAVs are constrained by limited computational and communication resources. So we propose an AKA scheme that enables secure communication between the UAV and BS. Specifically, proposed scheme eliminates reliance on a trusted authority during the AKA phase, while still ensuring the establishment of a secure communication channel. In addition, by combining the Chinese remainder theorem with the chameleon hash function, the scheme not only enables mutual authentication between UAVs and base stations but also enhances overall computational efficiency in complex environments. Formal security analysis and rigorous proof indicate this design upholds various protective attributes and effectively withstands diverse known attacks. Finally, experimental evaluations validate efficiency and practicality about proposed scheme in some environments.**

***Index Terms*—Authentication, key agreement, chameleon hash function, Chinese remainder theorem (CRT).**

I. Introduction

Unmanned Aerial Vehicle (UAV) are compact and versatile aerial platforms. Recently, UAVs have been widely adopted in emergency rescue environments [1], [2]. For instance, during fires, traffic accidents, and other sudden disasters, UAVs can be rapidly deployed for environmental monitoring, victim search and localization, and the delivery of essential supplies in affected areas, thereby substantially improving rescue efficiency [3]. However, in such emergency rescue environments, the stability and security of communication networks are critical, as they directly determine the efficiency of task coordination and the reliability of rescue operations. Therefore, constructing a reliable communication network is an essential prerequisite for ensuring the successful execution of emergency rescue missions.

The work was supported in part by the National Natural Science Foundation of China under Grant 62372002, Grant U24A20243, Grant 62272002 and Grant 62202005, in part by the Natural Science Foundation of Anhui Province, China under Grant 2508085QF243 and in part by the China Postdoctoral Science Foundation under Grant 2025M771549.

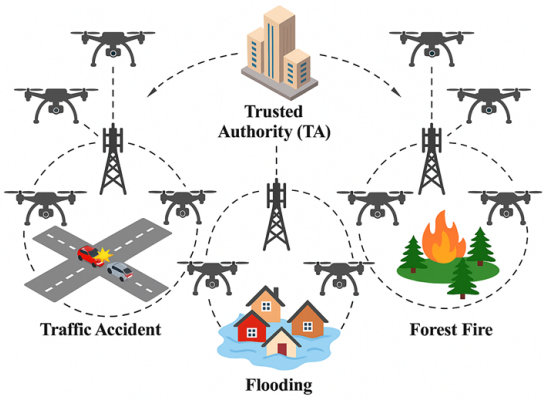

Fig. 1. UAV-Assisted Emergency Rescue Scenario.

Fig. 1 illustrates a representative collaborative rescue framework utilizing a UAV swarm. In this framework, UAV swarm collaborate to transmit rescue data to base stations, where critical rescue information is subsequently extracted. By leveraging this collaborative mechanism, the UAV swarm efficiently acquires and processes data in complex environments, thereby reducing information acquisition costs and enhancing both data collection efficiency and response timeliness [4]. However, in practical applications, the framework continues to face several security and reliability challenges. In emergency rescue environments, data transmitted over public channels is highly susceptible to replay, tampering, and forgery attacks, which pose serious threats to the accuracy and integrity of rescue information [5], [6]. Moreover, UAV swarms involve multi-node collaboration, and without effective identity authentication mechanisms, malicious nodes may impersonate legitimate UAVs to join the swarm, thereby disrupting task execution or even compromising the entire rescue process [7], [8].

Authentication and Key Agreement (AKA) serves as the fundamental mechanism for establishing secure communications within UAV swarms [9]. It allows UAVs to authenticate with base stations and negotiate shared session keys, thereby

forming a trusted bidirectional communication channel [10], [11], [12]. As the operating environment of UAVs is relatively complex, they may be unable to maintain persistent communication with the trusted authority [13]. When the authentication process depends on online participation of a trusted authority, the complex interactions inevitably introduce extra communication rounds, incurring additional overhead and creating potential single-point-of-failure vulnerabilities [14], [15]. More critically, certain schemes risk disclosing true identities of UAVs during authentication, which can result in privacy leakage [16].

Moreover, an effective traceability mechanism within a UAV swarm is essential for enabling the timely identification and tracing of malicious nodes when abnormal behavior occurs [17], [18]. In practice, malicious UAVs may disrupt swarm collaboration through data tampering or other similar attacks. If traceability is lacking, malicious UAVs cannot be detected and revoked in a timely manner, potentially leading to mission failure or severe security incidents [19], [20]. In the event of a security incident, trusted authority should be able to uncover the UAV's identity. Therefore, in emergency rescue environments, it is necessary to design an AKA scheme with the ability to ensure secure authentication when the trusted authority is offline, preserves UAV identity privacy, and enforces effective traceability. The main contributions of this paper are summarized as follows.

1) Focusing on the security problems of UAV swarms in emergency rescue environments, we propose an AKA scheme. The scheme uses a chameleon hash function to enable rapid authentication of UAV by base station and leverages the Chinese remainder theorem (CRT) to facilitate distributing trapdoor key, ensuring that only UAVs possessing the trapdoor key can be successfully authenticated. Moreover, this scheme operates without requiring the trusted authority to participate online, thereby reducing reliance on it.
2) To address the disruption of swarm cooperation caused by malicious UAVs in emergency rescue environments, particularly through data tampering, we propose a traceability mechanism that provides reliable misbehavior detection and accountability.
3) A formal security analysis based on the Real-or-Random (ROR) model demonstrates the strength of the scheme. Performance assessments indicate that it delivers a secure AKA process while keeping both computational and communication overheads low.

The remainder of this paper is organized as follows: Section II reviews several representative studies. Section III introduces the necessary preliminary knowledge. Section IV provides background information. Section V introduces the full workflow of this scheme, Section VI provides a security analysis. Subsequently, Section VII presents the performance analysis. Finally, Section VIII wraps up the scheme and highlights possible avenues for future task.

II. RELATED WORK

Owing to openness of wireless communication and high mobility of UAVs, UAV swarms encounter substantial challenges in AKA. The existing schemes have some limitations, particularly in emergency rescue environments.

Zhang et al. [21] proposed a seamless handover authentication scheme utilizing chameleon hash function. In this scheme, devices possessing trapdoor key are able to generate inputs that satisfy authentication requirements, whereas verifiers conduct identity authentication using predefined chameleon hash values. Pundir et al. [22] proposed a secure authentication scheme using hash functions with symmetric encryption algorithms, allowing UAVs and ground stations to establish session key under the supervision of a trusted authority. Xue et al. [23] proposed an authentication scheme using temporary credentials, where in the verifier authenticates a device's identity by validating the credential's legitimacy. However, renewing temporary credentials results in considerable overhead, and the update process also requires involvement from trusted authority.

Zhang et al. [24] proposed a dynamic AKA scheme that ensures sender non-repudiation and privacy protection. This scheme allows members to establish a key and distribute individual decryption keys with only one communication in open networks. However, the scheme requires numerous bilinear pairings, which renders it impractical for UAV swarm communications. Xiong et al. [25] proposed a authentication scheme utilizing Chinese remainder theorem and elliptic curve cryptography, which achieves authentication while simultaneously ensuring conditional privacy preservation. Liu et al. [26] proposed a collaborative key generation scheme. All nodes jointly generate and maintain the key. However, changes in node states render the key generation process complex and time-consuming, which may negatively impact task performance and overall efficiency.

Wang et al. [27] proposed UAV-assisted AKA scheme using three-factor authentication with physical unclonable function (PUF). It ensures the physical security of devices, thereby enhancing communication security in high-risk environments [28]. In [29], UAV-assisted authentication scheme was proposed that tailored for edge computing environments, where UAVs function as intermediate nodes to generate session keys between vehicles and roadside units. Zhang et al. [30] proposed chaotic mapping to generate symmetric session keys among UAVs and incorporated secret sharing techniques to derive group keys for UAV swarms. However, the scheme lacks traceability, which enables malicious UAVs to disseminate false information and consequently undermine group collaboration.

In [31], blockchain-based AKA scheme was proposed that exploits decentralization and fault-tolerance features of blockchain to mitigate single-point failures. Yu et al. [32] proposed an AKA agreement scheme. However, the authentication process between UAVs and base stations depends on blockchain access, which significantly decreases system

efficiency. In [33], blockchain-assisted AKA scheme was proposed. Although blockchain offers strong security guarantees, its heavy reliance on blockchain infrastructure substantially increases the complexity of system deployment. In complex environments, the efficient deployment and stable operation of such schemes remain challenging [34].

According to the above representative works, it can be observed that most existing schemes suffer from heavy reliance on trusted authority, high system overhead, and inadequate an effective traceability. The proposed scheme ensures system security while providing traceability.

III. PRELIMINARIES

A. Chameleon Hash Function

The chameleon hash function is defined as $CH_{pk}(m_0, r_0) = m_0 \cdot P + r_0 \cdot pk$, with the trapdoor $tp = (x, k^*)$, Let $k^* = m_0 + r_0 \cdot x$. The properties of the chameleon hash function are as follows:

1) One-wayness: Given the input (m_0, r_0, pk), computing CH_{pk} is straightforward. However, it is infeasible to compute m_0 and r_0 from the output value $CH_{pk} = m_0 \cdot P + r_0 \cdot pk$.
2) Trapdoor collision: Given the trapdoor $tp = (x, k^*)$, for an initial input (m_0, r_0), and a given input r_1, one can compute $CH_{pk}(m_1, r_1) = CH_{pk}(m_0, r_0)$, where $m_1 = k^* - r_1 \cdot x$.
3) Collision resistance: Without the trapdoor tp, it is infeasible to find $(m_1, r_1) \neq (m_0, r_0)$ such that $CH_{pk}(m_1, r_1) = CH_{pk}(m_0, r_0)$ for a given initial input (m_0, r_0).

B. Chinese Remainder Theorem

The CRT can be applied to address systems of linear congruences. If an integer n is divided by multiple integers and the remainders of each division are known, it is possible to compute the original integer n using these remainders. Based on this property, a group key can be embedded into corresponding congruences for secure key distribution. For k congruences:

$$\begin{cases} x \equiv a_1 \pmod{n_1} \\ x \equiv a_2 \pmod{n_2} \\ \vdots \\ x \equiv a_k \pmod{n_k} \end{cases}$$

The solution process of the Chinese remainder theorem mainly involves the following steps:

Compute the product of all moduli $n_1, n_2, \ldots, n_k$ as $N = \prod_{i=1}^{k} n_i$.

For the ith congruence equation:

1) Compute $N_i = \frac{N}{n_i}$.
2) Compute the modular inverse M_i of N_i modulo n_i, such that $N_i \cdot M_i \equiv 1 \pmod{n_i}$.
3) Finally, compute $x = \sum_{i=1}^{k} a_i \cdot N_i \cdot M_i \mod N$.

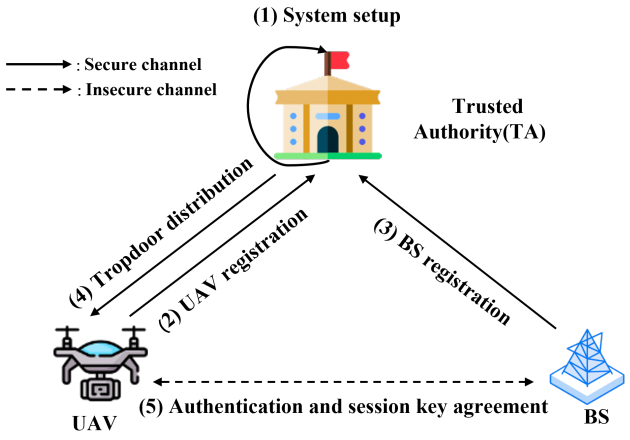

Fig. 2. System model for UAV-Assisted Emergency Rescue.

C. Complexity Assumptions

- DL Problem: Let G be a cyclic group of prime order q defined over an elliptic curve $E(F_t)$, and let P be its generator. Given $(P, Q = kP)$, it is computationally infeasible to derive k from $Q = kP$.
- CDH Problem: Consider a cyclic group G of prime order q with generator P defined on an elliptic curve $E(F_t)$. For $a, b \in \mathbb{Z}_q^*$ and the corresponding points aP and bP in G, the CDH problem is described as determining abP from the tuple (P, aP, bP).

IV. BACKGROUND

A. System Model

The model involves three entities: Trusted Authority (TA), Base Station (BS), and Unmanned Aerial Vehicle (UAV), as illustrated in Fig. 2.

- Trusted Authority (TA): TA acts as a fully trusted authority within the system. It providing registration support for UAVs and the BS, ensuring the security of registration information.
- Base Station (BS): It broadcasts messages to UAVs within the group and is responsible for authenticating external UAVs and establishing secure communication channels with them.
- UAV: The UAV is considered a semi-trusted entity with constrained computational resources. It is tasked with collecting and transmitting environmental data.

B. Threat Model

Similar to the schemes presented in [35] and [36], our approach adopts the recognized Dolev-Yao threat model. All entities interact via an insecure, physically open channel. The adversary $\mathcal{A}$ possesses both active and passive attack capabilities, including impersonation, eavesdropping, message tampering, and other attacks. $\mathcal{A}$ is a legitimate network user, capable of accessing all transmitted messages within the network, interacting with other users, receiving messages from the subject, and impersonating the subject to send messages

TABLE I
NOTATIONS

Notation	Description
BS_j	The j-th BS
UAV_i	The i-th UAV
PID_i	The pseudonyms of UAV_i
ID_i	The real identity of UAV_i
ID_j	The real identity of BS_j
T_i	Timestamp
s_i, s_j	Secret key of UAV_i and BS_j, respectively
R_i, R_j	Public key of UAV_i and BS_j, respectively
(k_i, x_g)	The trapdoor of UAV_i
MAC_i	The message authentication code
$SK_{i,j}$	Secret session key between UAV_i and BS_j

to other entities. However, adversary $\mathcal{A}$ is computationally bounded and cannot accurately guess random numbers in sufficiently large finite fields, nor can it solve certain existing hard mathematical problems.

V. THE PROPOSED SCHEME

This section presents the scheme and outlines its phases: system initialization, UAV and BS registration, tropdoor key distribution, mutual AKA and device traceability. Table I lists the symbols used in the scheme.

A. System Initialization

TA initializes several system parameters. Over a finite field $\mathbb{F}_p$, and $E : y^2 = x^3 + ax + b \mod p$ is selected, where $\mathbb{F}_p$ is a finite field defined by a prime number p, and $a, b \in \mathbb{F}_p$. A point P on E is chosen to generate a cyclic group G of order q. Additionally, TA select $s_t \in \mathbb{Z}_q^*$, and compute $P_{pub} = s_t \cdot P$. Eventually, hash function $h : \{0,1\}^* \rightarrow \mathbb{Z}_q^*$ and publishes common parameters $\{E, q, G, P_{pub}, \mathbb{Z}_q^*, h\}$ to system.

B. Entity Registration

Each legitimate UAV and BS must register with the TA for subsequent identity authentication and key exchange, as illustrated in Fig. 3. The process is executed via a secure channel.

1) UAV Registration

i) UAV_i sends its identity information ID_i to the TA.

ii) Upon receiving the message, the TA checks whether ID_i already exists in its local database to determine whether UAV_i has previously registered. If it exists, the TA rejects the registration request. Otherwise, the TA selects $s_i \in \mathbb{Z}_q^*$ and chooses two initial inputs $m_0, r_0 \in \mathbb{Z}_q^*$. The $R_i = s_i \cdot P$. The TA then sends s_i, R_i, m_0 and r_0 to UAV_i and securely stores $\{ID_i, R_i\}$.

iii) After receiving the message, UAV_i securely stores s_i, m_0, r_0 and R_i in secret.

2) BS Registration

i) BS_j sends its identity information ID_j to the TA.

ii) Upon receiving the message, the TA checks whether ID_j already exists locally to determine if BS_j has previously registered. If it does, the TA rejects the registration request. Otherwise, TA selects $c_j \in \mathbb{Z}_q^*$, computing the public key as $R_j = c_j P$, , private key as $s_j = c_j + h(ID_j \,\|\, R_j)s_t$. Then TA secretly store $\{R_j, ID_j\}$.

iii) After receiving the message, BS_j securely stores $\{s_j, R_j\}$ and publishes R_j.

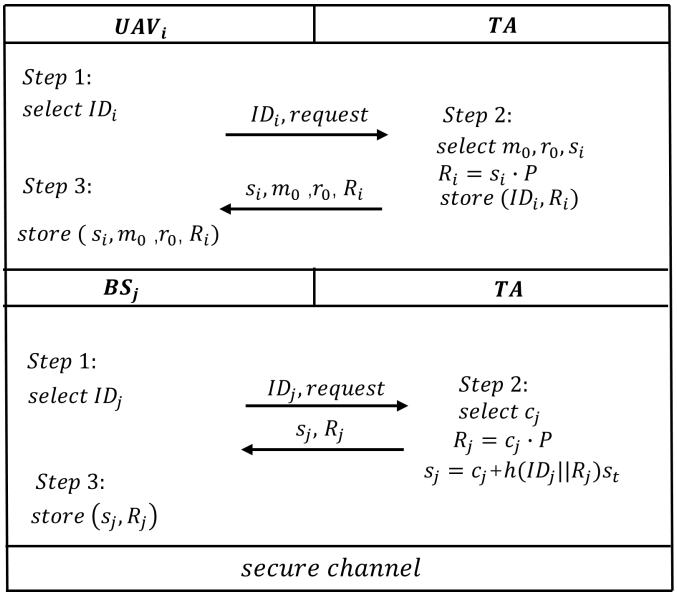

Fig. 3. Entity Registration.

C. Tropdoor Key Distribution

1) After all UAVs within the group complete registration, the TA computes $N = \prod_{i=1}^{l} s_i$, where l is the number of UAVs in the group. Then, it calculates $N_i = N/s_i$ and the multiplicative inverse M_i of N_i. Let $v_i = M_i \cdot N_i$, and compute $u = \sum_{i=1}^{l} v_i$. TA selects $x_g \in \mathbb{Z}_q^*$ as the group key, computing $t_g = u \cdot x_g$ and the group public key $PK_g = x_g \cdot P$. It then computes $ch_g = m_0 \cdot P + r_0 \cdot PK_g$, and selects $y_g \in \mathbb{Z}_q^*$, compute $Y_g = y_g P$. The TA generate the signature $SIG = y_g + h(Y_g \,\|\, t_g \,\|\, PK_g \,\|\, CT_i)s_t$, where CT_i denotes the validity period of x_g. Then TA broadcast the message $\{t_g, Y_g, PK_g, SIG, CT_i\}$ to the UAVs in the group via the BS_j. Meanwhile, ch_g is securely preloaded into the BS_j.

2) UAV_i verify the signature $SIG \cdot P \stackrel{?}{=} Y_g + h(Y_g \,\|\, t_g \,\|\, PK_g \,\|\, CT_i)P_{pub}$. Each UAV in the group only needs to perform a single modular division operation $x_g = t_g \bmod s_i$ to obtain the group key x_g. It then computes the trapdoor $k_i = m_0 + r_0 \cdot x_g$, and securely stores (k_i, x_g).

D. Mutual Authentication and Key Agreement

If UAV_i intends to communicate securely with BS_j, it must establish a secure session key SK through this step. The authentication procedure is shown in Fig. 4. The details of the authentication are as follows:

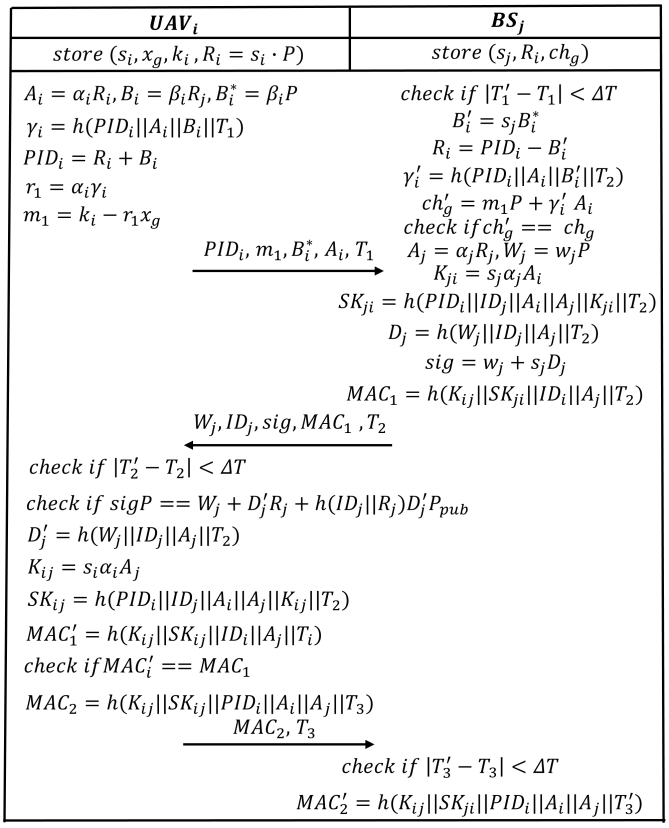

Fig. 4. Mutual authentication and key agreement.

1) When the UAV_i moves into the service range of BS BS_j, mutual AKA between UAV_i and BS_j must be initiated to establish a session key. UAV_i selects two random numbers α_i, $\beta_i \in \mathbb{Z}_q^*$, the current timestamp T_1. The UAV_i computes $A_i = \alpha_i \cdot R_i$, $B_i = \beta_i \cdot R_j$, $B_i^* = \beta_i \cdot P$ and the pseudonym identity $PID_i = R_i + B_i$. It uses the locally stored (k_i, x_g) to compute the initial input $\gamma_i = h(PID_i \| A_i \| B_i \| T_1)$, $r_1 = \alpha_i \gamma_i$ then $m_1 = k_i - r_1 \cdot x_g$. And UAV_i sends the message $MSG_1 = \{PID_i, m_1, B_i^*, A_i, T_1\}$ to BS_j.
2) Upon receiving MSG_1, BS_j selects the current timestamp T_1' to verify the freshness of T_1, ensuring that $T_1' - T_1 \leq \Delta T$. ΔT is a predefined time threshold. If the verification sucesses, BS_j computes $B_i' = s_j \cdot B_i^*$ and recovers $R_i = PID_i - B_i'$, then checks the local database for the existence of R_i. If found, it computes $\gamma_i' = h(PID_i \| A_i \| B_i' \| T_1)$ and verifies $ch_g' = m_1 \cdot P + \gamma_i' \cdot A_i$. If $ch_g' = ch_g$, the authentication of UAV_i is successful. The BS then selects the current timestamp T_2, random numbers $\alpha_j, w_j \in \mathbb{Z}_q^*$, and computes $A_j = \alpha_j \cdot R_j$, $K_{ji} = s_j \alpha_j A_i$, and $W_j = w_j \cdot P$. The session key is derived as $SK_{ji} = h(PID_i \| ID_j \| A_i \| A_j \| K_{ji} \| T_2)$. The BS_j computes $D_j = h(W_i \,\|\, ID_j \,\|\, A_j \,\|\, T_2)$ and generates the signature $sig = w_i + D_j\, s_j$. The message authentication code is computed as $MAC_1 = h(K_{ji} \| SK_{ji} \| ID_j \| A_j \| T_2)$. Then it send the message $MSG_2 = \{W_j, ID_j, A_j, sig, MAC_1, T_2\}$.
3) Upon receiving message MSG_3, it selects the current timestamp T_2' to verify the freshness of T_2, ensuring $T_2' - T_2 \leq \Delta T$. It then verifies $sig \cdot P \stackrel{?}{=} W_j + D_j' R_j + h(ID_j \,\|\, R_j) D_j' P_{pub}$, where $D_j' = h(W_j \,\|\, ID_j \,\|\, A_j \,\|\, T_2)$. If the verifications are successful, it computes $K_{ji} = s_i \alpha_i A_j$, and derives the session key $SK_{ij} = h(PID_i \| ID_j \| A_i \| A_j \| K_{ji} \| T_2)$. Then, it computes $MAC_1' = h(K_{ij} \| SK_{ij} \| ID_j \| A_j \| T_2)$ and checks whether $MAC_1' = MAC_1$. If this verification passes, SK_{ij} is accepted as the session key. Finally, it obtains T_3, computing $MAC_2 = h(K_{ij} \| SK_{ij} \| PID_i \| A_i \| A_j \| T_3)$. Then send the message $MSG_3 = \{MAC_2, T_3\}$ to BS_j.
4) Upon receiving message MSG_3, BS_j first checks whether $T_3' - T_3 \leq \Delta T$, where T_3' is current timestamp. It then computes $MAC_2' = h(K_{ji} \| SK_{ji} \| PID_i \| A_i \| A_j \| T_3)$ and verifies whether $MAC_2' = MAC_2$. If successful, the BS_j accepts SK_{ji} as the key shared with UAV_i.

E. Device Traceability

When UAV_i uses the pseudonym $PID_i = R_i + B_i$ to perform malicious activities, the TA can compute $R_i = PID_i - B_i$. Then, the TA queries the locally stored $\{ID_i, R_i\}$ to reveal and revoke ID_i of UAV_i. Afterward, TA computes $u' = u - v_i$, selects $x_g' \in \mathbb{Z}_q^*$, computing $t_g' = ux_g'$ as well as $PK_g' = x_g' P$. Then, the TA computes $ch_g^* = m_0 P + r_0 PK_g'$, selects $y_g' \in \mathbb{Z}_q^*$, computing $Y_g' = y_g' P$ to generate the signature $SIG' = y_g' + h(Y_g' || t_g' || PK_g' || CT_i) s_t$. Subsequently, the TA broadcasts the message $\{t_g', Y_g', PK_g', SIG', CT_i\}$ to the UAVs in the group via the BS_j, and preloads ch_g^* securely into the BS_j. Upon receiving the message from the BS_j, UAVs in the group verify the signature by checking $SIG' P \stackrel{?}{=} Y_g' + h(Y_g' || t_g' || PK_g' || CT_i) P_{pub}$. If the verification passes, each UAV in the group only needs to perform one modular division operation to obtain x_g', and then computes $k_i' = m_0 + r_0 x_g'$, which is secretly stored as (k_i', x_g').

VI. SECURITY ANALYSIS AND PROOF

We use the ROR model [37], [38] in conjunction with heuristic security analysis. These methods provide complementary perspectives that collectively offer strong support for the scheme's security, ensuring its resilience against various real-world security threats.

A. Security Analysis

The ROR model functions as an established framework for analyzing various cryptographic schemes and security schemes, such as key agreement schemes, authentication mechanisms, and encryption systems. It has been extensively applied across both traditional network communication security and emerging domains such as UAVs and vehicular networks [39]. As a foundation for formal security analysis, the core idea is to evaluate whether an adversary differentiate between a session key and randomly created ones. If adversary

fails to make this distinction effectively, the scheme is deemed secure under the model.

Participants: Let Π_{UAV}^{e} and Π_{BS}^{f} denote the eth and fth instances of the scheme executed by the UAV and the BS, respectively.

Acceptance State: If all messages exchanged between the UAV and the BS are completed in the correct order and the final message is successfully received, then Π_{UAV}^{e} and Π_{BS}^{f} are said to be in an acceptance state.

Matching Sessions: If Π_{UAV}^{e} and Π_{BS}^{f} are part of the same session and both are in the acceptance state, they are considered matching instances.

Freshness: If SK established between Π_{UAV}^{e} and Π_{BS}^{f} has not been obtained by the adversary $\mathcal{A}$, the session is considered fresh.

Adversary Model: The adversary $\mathcal{A}$ can interact with Π_{UAV}^{e} and Π_{BS}^{f} through the following predefined queries and is capable of modifying and replaying communication messages.

1) Execute($\Pi_{UAV}^{e}, \Pi_{BS}^{f}$): This query indicates that the adversary $\mathcal{A}$ can eavesdrop on all messages exchanged between Π_{UAV}^{e} and Π_{BS}^{f}.

2) Send($\Pi_{UAV}^{e}, \Pi_{BS}^{f}, m$): This query represents a direct offensive attempt. It can generate, intercept, modify, and replay m to Π_{UAV}^{e} and Π_{BS}^{f}. Upon receiving message m, Π_{UAV}^{e} and Π_{BS}^{f} will return a response message to $\mathcal{A}$.

3) CorruptUAV(Π_{UAV}^{e}): Through it, adversary gains access to key information of Π_{UAV}^{e}.

4) CorruptBS(Π_{BS}^{f}): Through it, adversary gains access to key information of Π_{BS}^{f}.

5) Test($\Pi_{UAV}^{e}, \Pi_{BS}^{f}$): When Π_{UAV}^{e} and Π_{BS}^{f} share the same SK and $\mathcal{A}$ issues this query, if the hidden bit $b = 1$, $\mathcal{A}$ receives the real SK. If $b = 0$, $\mathcal{A}$ is given an unrelated bitstring matching the size of SK, while b remains unknown to $\mathcal{A}$. If Π_{UAV}^{e} and Π_{BS}^{f} do not share a session key, $\mathcal{A}$ receives an invalid symbol $\perp$.

Semantic Security of the Session Key: We can infer that the generated SK is sufficiently random and avoids information leakage. The adversary $\mathcal{A}$ can perform the itTest($\Pi_{UAV}^{e}, \Pi_{BS}^{f}$) query and guess the bit $b' \in \{0, 1\}$. If $b' = b$, then $\mathcal{A}$ wins. Let scheme be denoted by P. $\mathcal{A}$'s capability against scheme P within time t under the ROR model is quantified as $Adv_{\mathcal{A}}^{P} = |2 \cdot \Pr[b' = b] - 1|$, where $\Pr[EV]$ represents the likelihood of event EV occurring. If $Adv_{\mathcal{A}}^{P}$ is negligible, P is considered secure under the ROR model.

Definition 1: Given $H : \{0,1\}^{*} \rightarrow \{0,1\}^{w}$, We quantify how often $\mathcal{A}$ finds a collision within time t as $Adv_{\mathcal{A}}^{HASH} = \Pr[(u_1 = u_2) \leftarrow \mathcal{A} : u_1 \neq u_2 \text{ and } H(u_1) = H(u_2)]$, where u_1 and u_2 are randomly chosen by the adversary $\mathcal{A}$.

Definition 2: Given the set $\{P, \delta_1 P, \delta_2 P\}$ with $\delta_1, \delta_2 \in \mathbb{Z}_q^{*}$, we define the probability that an adversary $\mathcal{A}$ successfully computes $\delta_1 \delta_2 P$, i.e., solves the CDH, as $Adv_{\mathcal{A}}^{CDH}$.

Theorem 1: Let q_H represent the count of hash invocations, $|Hash|$ refer to the dimension of the hash function's output space, and $Adv_{\mathcal{A}}^{CDH}$ capture the likelihood that $\mathcal{A}$ solves the CDH problem. The adversary's ability to compromise security of scheme is bounded by

$$Adv_{\mathcal{A}}^{p} \leq \frac{q_H^2}{|Hash|} + 2 \cdot Adv_{\mathcal{A}}^{CDH}.$$

Proof: We adopt the proof technique proposed in [40]. To compute $Adv_{\mathcal{A}}^{p}$, we define the following sequence of games $Game_i$, and let $succ_i$ represent the case where $\mathcal{A}$ identifies $b' = b$ in $Game_i$.

Game 0: It allows $\mathcal{A}$ to conduct an actual offensive action against scheme P. Before the game starts, $\mathcal{A}$ is given a random bit b. It follows $Adv_{\mathcal{A}}^{p} = |2 \cdot \Pr[succ_0] - 1|$.

Game 1: $\mathcal{A}$ is allowed to passively eavesdrop on the message exchange between Π_{UAV}^{e} and Π_{BS}^{f} by executing the query it Execute($\Pi_{UAV}^{e}, \Pi_{BS}^{f}$). Eventually, $\mathcal{A}$ performs it Test($\Pi_{UAV}^{e}, \Pi_{BS}^{f}$). Recall the key generation formula: $SK = h(PID_i \| ID_j \| A_i \| A_j \| K_{ij} \| T_2)$, where $K_{ji} = s_i \alpha_i A_j = s_j \alpha_j A_i$. However, cannot derive s_i, α_i or s_j, α_j through passive eavesdropping, and thus cannot compute real SK. Therefore, Game$_0$ is equivalent to Game$_1$, It follows $\Pr[succ_0] = \Pr[succ_1]$.

Game 2: It allows $\mathcal{A}$ to perform itSend($\Pi_{UAV}^{e}, \Pi_{BS}^{f}, m$) query and make hash queries as in Game$_1$. At this stage, $\mathcal{A}$ launches impersonation attacks by generating, modifying, and sending message requests to Π_{UAV}^{e} and Π_{BS}^{f}. In our scheme, the messages that $\mathcal{A}$ can modify include PID_i, B_i^{*}, and $\{MAC_1, sig, A_i, A_j\}$, all of which incorporate random factors and independent timestamps. Therefore, $\mathcal{A}$ can perform it Send without causing hash collisions. Based on birthday paradox, we obtain:

$$|\Pr[succ_1] - \Pr[succ_2]| \leq \frac{q_H^2}{2|Hash|}.$$

Game 3: It allows $\mathcal{A}$ to obtain SK by executing itCorruptUAV(Π_{UAV}^{e}) or itCorruptBS(Π_{BS}^{f}). The adversary can access information such as PID_i, B_i^{*}, and $\{MAC_1, A_i, A_j\}$. The key is generated as: $SK = h(PID_i \| ID_j \| A_i \| A_j \| K_{ij} \| T_2)$, where $K_{ji} = s_i \alpha_i A_j = s_j \alpha_j A_i$, which results from the unpredictable values α_i, α_j. These values arise only during the mutual AKA between Π_{UAV}^{e} and Π_{BS}^{f}, and not stored after use. Therefore, in order to derive SK, the adversary must compute K_{ij} from A_i and A_j, which reduces to solving the CDH problem. Thus, we obtain:

$$|\Pr[succ_2] - \Pr[succ_3]| \leq Adv_{\mathcal{A}}^{CDH}.$$

In the final game, $\mathcal{A}$ has invoked all available query interfaces in an attempt to compromise the protection framework of P, must resort to guessing bit b to win. Therefore, we have:

$\Pr[succ_3] = \frac{1}{2}$. Based on the inequalities above, we obtain:

$$\begin{aligned}\frac{1}{2}Adv_{\mathcal{A}}^{p} &= \left|\Pr[succ_0] - \frac{1}{2}\right| \\ &= |\Pr[succ_0] - \Pr[succ_3]| \\ &\leq |\Pr[succ_0] - \Pr[succ_1]| + |\Pr[succ_1] - \Pr[succ_2]| \\ &\quad + |\Pr[succ_2] - \Pr[succ_3]| \\ &\leq \frac{q_H^2}{2|Hash|} + Adv_{\mathcal{A}}^{CDH}.\end{aligned}$$

Therefore, we obtain:

$$Adv_{\mathcal{A}}^{p} \leq \frac{q_H^2}{|Hash|} + 2 \cdot Adv_{\mathcal{A}}^{CDH}.$$

B. Security Analysis

1) Mutual Authentication: The authentication between UAV_i and BS_j is based on the construction of PID_i and ch_i, where each entity verifies the other by checking the received messages. BS_j authenticates UAV_i by verifying PID_i and ch_i, and then UAV_i authenticates BS_j by verifying sig, thereby achieving mutual authentication.

2) Session Key Establishment: The scheme we propose enables both UAV_i and BS_j to construct SK, computed as $SK = h(PID_i \| ID_j \| A_i \| A_j \| K_{ji} \| T_2)$, where T_2 is a timestamp. The ability to decrypt the ciphertext and derive the SK is restricted to adversaries in possession of the private key.

3) Device Anonymity: UAV_i does not expose its real identity when the authentication process. Instead, it uses a dynamically updated pseudonym $PID_i = R_i + B_i$, where $B_i = \beta_i R_j$. Since the adversary cannot obtain s_j, it is infeasible to recover the UAV_i's identity. Therefore, the proposed scheme achieves device anonymity.

4) Device Traceability: If UAV_i uses the pseudonym $PID_i = R_i + B_i$ to perform malicious activities and is detected, TA can compute $R_i = PID_i - B_i$. Then, the TA queries the locally stored $\{ID_i, R_i\}$ to reveal the UAV's true identity ID_i.

5) Resistance to Message Tampering Attacks: Adversary $\mathcal{A}$ cannot compute r_1 without knowing x_g and s_i. Since the adversary does not possess trapdoor of chameleon hash function, they are unable to forge r_1. If $\mathcal{A}$ forges γ_i', the derived r_1 will not pass verification by BS_j, making tampering with MSG_1 infeasible. The UAV_i authenticates BS_j through sig, and ensures data integrity by verifying MAC_1. Since the adversary lacks access to s_j, α_j, and w_j, forging MAC_1 is not possible. If MSG_2 is tampered with, the MAC_1' computed by UAV_i will fail verification. The analysis for MSG_3 is similar to that of MSG_1 and MSG_2, and is therefore omitted.

6) Resistance to Replay Attacks: We uses a widely accepted timestamp verification mechanism. When an adversary attempts to replay previously transmitted messages, the receiving party verifies whether $T_1' - T_1 \leq \Delta T$. If holds, the verification proceeds to the next. Otherwise, the process is terminated.

7) Resistance to Impersonation Attacks: The adversary cannot compute x_g and s_i, thus making it infeasible to impersonate UAV_i. For impersonating BS_j, we assume that BS_j is secure and that the adversary cannot obtain local data from the BS. As a result, the adversary cannot forge a valid MAC_1 that would pass verification by UAV_i. This ensures that the adversary cannot impersonate BS_j.

8) Resistance to Man-in-the-Middle Attacks: The attack occurs that adversary interacts separately with UAV_i and BS_j, attempting to authenticate with both. However, our scheme enables mutual authentication between entities. If the adversary were capable of impersonating a legitimate UAV_i or BS_j, it would contradict the resistance to impersonation attacks.

9) Resistance to Known Session Key Attacks: After authentication, UAV_i and BS_j establish a session key $SK = h(PID_i \| ID_j \| A_i \| A_j \| K_{ji} \| T_2)$, where $K_{ji} = s_j \alpha_j A_i$. Constructing K_{ji} relies on s_j and the random number α_j. Thus, although an attacker may gain access to SK, Other keys security is not compromised as a result.If an adversary intends to compute K_{ji}, it must first solve the CDH problem.

TABLE II
COMPARISON OF SECURITY PROPERTIES

Security Property	Xie et al.	Sutrala et al.	Saleem et al.	Ours
Replay Attack	✓	✓	×	✓
Anonymity	×	×	✓	✓
TA offline authentication	✓	✓	×	✓
No Password Exposure	✓	×	✓	✓
Data Tampering Attack	×	✓	✓	✓
Traceability	×	×	×	✓
Impersonation Attacks	✓	✓	×	✓
Man-in-the-middle Attack	✓	✓	×	✓
Forward Secret	✓	×	✓	✓
Known Session Key Attack	✓	✓	×	✓
Eavesdropping Attacks	✓	×	✓	✓

VII. PERFORMANCE ANALYSIS

The scheme is evaluated with respect to computation and communication cost, compared with three related schemes Xie et al. [35], Sutrala et al. [41], and Saleem et al. [42]. The evaluation utilised a system featuring an Intel i7-10700 2.90 GHz CPU and 32 GB of RAM. Cryptographic operations are implemented using the Miracl Core library, with the ED25519 elliptic curve selected and achieve 128-bit security level. PUF operations were simulated using the puf_py library in Python 3.13.3. Lightweight operations such as XOR are considered negligible in terms of computational overhead. The SHA-256 algorithm was useed as the secure hash function.

A. System Security Property Analysis

We have analyzed the security properties of the scheme put forward, and further contrasted them with those of the reference schemes. From Table II, Our scheme has more security attributes and exhibits similar security features to [35]. In conclusion, our proposed scheme achieves the highest level of performance in the aspect of system security.

TABLE III
BASIC OPERATION TIMES

Operation	Description	Time Cost (ms)
T_h	Hash operation	0.018
T_{puf}	PUF generation	0.756
T_{pa}	Point addition on ECC	0.004
T_{sm}	Scalar multiplication on ECC	0.586
T_{enc}	Encryption operation	0.356
T_{dec}	Decryption operation	0.378

B. Comparison of Simulation Calculation Costs

This section provides a statistical analysis about computational cost associated with proposed scheme and offers a comparative evaluation against several related schemes. The overall computational cost is determined by tallying cryptographic operations required at each stage. Table III lists the execution time. The computational overheads of the registration phase and the mutual AKA phase are computed similarly, only the cost of the mutual AKA phase is presented for conciseness. As illustrated in Table IV, both the theoretical computational cost and the simulated cost derived from a single authentication process are compared. In the table, a dash ("–") denotes that the respective entity takes no engage in the verification procedure. During the mutual AKA between the UAV and BS, the UAV incurs a computational cost of $7T_{sm} + 3T_{pa} + 6T_h = 4.222$ ms, whereas the BS incurs $7T_{sm} + 2T_{pa} + 5T_h = 4.20$ ms. Therefore, the total computational cost amounts to 8.422 ms.

As the computational costs for schemes [35] and [42] can be obtained through similar methodologies, this discussion primarily focuses on [41]. In [41], the computational cost on the device side is $9T_{sm} + 3T_{pa} + 24T_h = 5.718$ ms, while the cost on the RSU side is $3T_{sm} + 2T_{pa} + 8T_h = 1.91$ ms. Although our computational cost is higher than scheme [41], the significantly lower device-side cost is essential for resource-constrained platforms such as UAVs. Furthermore, the proposed scheme exhibits superior performance compared to that of scheme [35]. Although our computational cost is higher than scheme [42], the latter exhibits weaker anonymity and traceability, thereby resulting in reduced overall security. In addition, scheme [42] necessitates the involvement of a TA during the authentication process, which incurs additional communication overhead and increases system burden. Overall, the proposed scheme provides enhanced security features, including self-generated pseudonyms and traceability, which are not supported by the previously discussed schemes. Consequently, although the proposed scheme introduces slightly higher computational overhead, this represents a reasonable trade-off for attaining stronger security guarantees.

Fig. 5 presents a comparative visualization of the computational costs associated with various schemes, clearly illustrating the cost distribution across different entities. As illustrated, the computational costs incurred by both the UAV and the BS remain within a reasonable range.

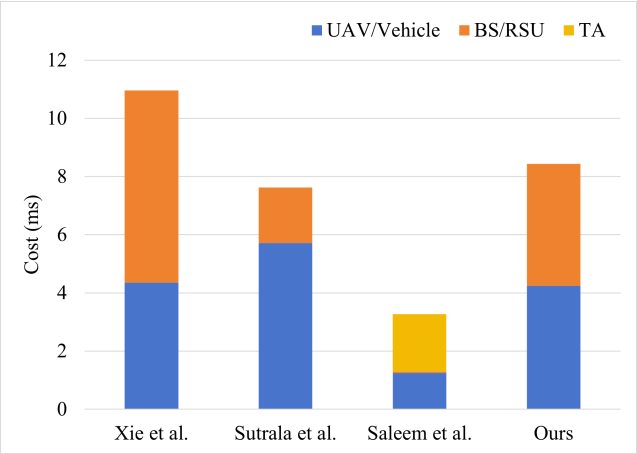

Fig. 5. Computation costs comparison.

C. Communication Overhead

We analysis contrasts the communication overhead involved in the authentication process for different schemes. In our scheme, the lengths of the ID, random number, PUF challenge, response are all set to 128 bits, while the lengths of the ECC points, hash results, AES-256 ciphertext, private key, and timestamp are set to 512 bits, 256 bits, 256 bits, 256 bits and 32 bits, respectively. The difference of communication overhead is shown in Fig. 6. To calculate communication overhead, we first define the bit sizes for different messages. Specifically, the first message, MSG_1, sent by UAV$_i$ to the BS$_j$ is given by $MSG_1 = \{PID_i, m_1, B_i^*, A_i, T_1\}$, thus the size of MSG_1 is $|MSG_1| = 512 + 128 + 512 + 512 + 32 = 1696$ bits. The message sent by BS$_j$ to UAV$_i$, $MSG_2 = \{W_i, ID_j, A_j, sig, MAC_1, T_2\}$, has a size of approximately $|MSG_2| = 512 + 512 + 256 + 128 + 128 + 32 = 1568$ bits. Finally, the message sent by UAV$_i$ to BS$_j$, $MSG_3 = \{MAC_3, T_3\}$, which contains only the hash value and timestamp, is approximately $|MSG_3| = 256 + 32 = 288$ bits. Therefore, the total communication overhead is $|MSG_1| + |MSG_2| + |MSG_3| = 1696 + 1568 + 288 = 3552$ bits.

Since the communication overheads of schemes [41] and [42] can be calculated in a similar way, we focus on Xie et al's scheme, and analyze detail. In scheme [35], the data sent from vehicle V_1 to RSU is $\{PID_i, A_1, D_i, PK_{V_i}, c_i, T_1\}$, which has a size of approximately $128 + 512 + 512 + 512 + 256 + 32 = 1952$ bits. The data sent from RSU to vehicle V_1 is $\{f_t, PK_{R_t}, E_t, Z_t, N_1, T_2, PID_i, RID_t\}$, which has a size of approximately $256 + 512 + 512 + 512 + 32 + 128 + 128 = 2080$ bits. The total overhead is 4032 bits. By comparing this with the two schemes, we has a lower overall communication overhead. We has significantly stronger security properties compared with the other schemes, especially in terms of defense capabilities. Compared with two of the schemes, our communication overhead is slightly lower, indicating that our scheme optimizes the use of communication resources while ensuring security. Overall, although the scheme has a higher communication overhead in some specific environments, the

TABLE IV
COMPARISON OF COMPUTATION COST (MS)

Scheme	Device	BS/RSU	TA	Total
Xie et al. [35]	$7T_{sm} + 4T_{pa} + 13T_h \approx 4.352$	$2T_{puf} + 8T_{sm} + 4T_{pa} + 22T_h \approx 6.612$	–	10.964
Sutrala et al. [41]	$9T_{sm} + 3T_{pa} + 24T_h \approx 5.718$	$3T_{sm} + 2T_{pa} + 8T_h \approx 1.91$	–	7.628
Saleem et al. [42]	$T_{puf} + T_{dec} + 6T_h \approx 1.242$	$3T_h \approx 0.054$	$T_{puf} + T_{dec} + 2T_{enc} + 6T_h \approx 1.972$	3.268
Ours	$7T_{sm} + 3T_{pa} + 6T_h \approx 4.222$	$7T_{sm} + 2T_{pa} + 5T_h \approx 4.20$	–	8.422

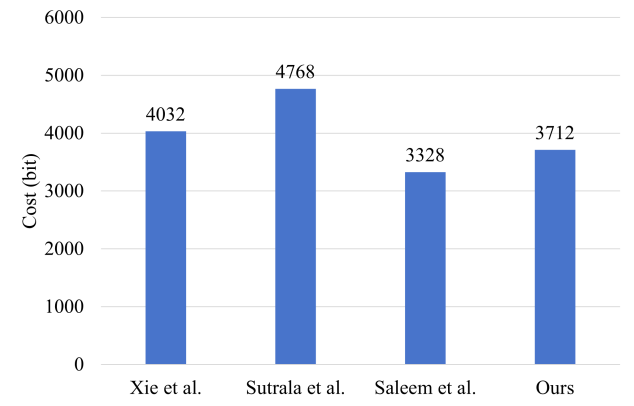

Fig. 6. Communication cost comparison.

security and robustness it provides make this overhead worthwhile. With further optimization, the communication overhead is expected to decrease further, thus improving the practical applicability and deployment effectiveness of our scheme.

VIII. CONCLUSION

This paper centers on UAV application in emergency rescue environments and highlights the security challenges they face. The AKA scheme using chameleon hash function and CRT. The scheme demonstrates strong performance while reducing reliance on the TA. Compared with existing methods, our scheme provides stronger security, making it particularly suitable for UAV swarm rescue tasks requiring rapid response. To thoroughly verify its security, we conduct a formal proof through the ROR model, along with heuristic security analysis. Experimental results indicate we can effectively resist most common attacks, thereby ensuring both security and practical applicability. In the future, We will design an AKA scheme with cross-domain support for UAV swarms.

REFERENCES

[1] N. H. Motlagh, T. Taleb, and O. Arouk, "Low-altitude unmanned aerial vehicles-based internet of things services: Comprehensive survey and future perspectives," *IEEE internet of things journal*, vol. 3, no. 6, pp. 899–922, 2016.

[2] H. Shen, T. Wang, J. Chen, Y. Tao, and F. Chen, "Blockchain-based batch authentication scheme for internet of vehicles," *IEEE Transactions on Vehicular Technology*, vol. 73, no. 6, pp. 7866–7879, 2024.

[3] P. Gope and B. Sikdar, "An efficient privacy-preserving authenticated key agreement scheme for edge-assisted internet of drones," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 11, pp. 13 621–13 630, 2020.

[4] Y. Liu, L. Ni, and M. Peng, "A secure and efficient authentication protocol for satellite-terrestrial networks," *IEEE Internet of Things Journal*, vol. 10, no. 7, pp. 5810–5822, 2022.

[5] J. Cui, Y. Zhu, H. Zhong, Q. Zhang, C. Gu, and D. He, "Efficient blockchain-based mutual authentication and session key agreement for cross-domain iiot," *IEEE Internet of Things Journal*, vol. 11, no. 9, pp. 16 325–16 338, 2024.

[6] X. Wang, S. Garg, H. Lin, G. Kaddoum, J. Hu, and M. F. Alhamid, "An intelligent uav based data aggregation algorithm for 5g-enabled internet of things," *Computer Networks*, vol. 185, p. 107628, 2021.

[7] C. Zhong, J. Yao, and J. Xu, "Secure uav communication with cooperative jamming and trajectory control," *IEEE Communications Letters*, vol. 23, no. 2, pp. 286–289, 2018.

[8] Y. Zeng and R. Zhang, "Energy-efficient uav communication with trajectory optimization," *IEEE Transactions on wireless communications*, vol. 16, no. 6, pp. 3747–3760, 2017.

[9] Q. Zhang, X. Zhou, H. Zhong, J. Cui, J. Li, and D. He, "Device-side lightweight mutual authentication and key agreement scheme based on chameleon hashing for industrial internet of things," *IEEE Transactions on Information Forensics and Security*, 2024.

[10] J. Srinivas, A. K. Das, N. Kumar, and J. J. Rodrigues, "Tcalas: Temporal credential-based anonymous lightweight authentication scheme for internet of drones environment," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 7, pp. 6903–6916, 2019.

[11] D. He, S. Chan, and M. Guizani, "Communication security of unmanned aerial vehicles," *IEEE wireless communications*, vol. 24, no. 4, pp. 134–139, 2016.

[12] T. Alladi, G. Bansal, V. Chamola, M. Guizani *et al.*, "Secauthuav: A novel authentication scheme for uav-ground station and uav-uav communication," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 12, pp. 15 068–15 077, 2020.

[13] C. Feng, B. Liu, Z. Guo, K. Yu, Z. Qin, and K.-K. R. Choo, "Blockchain-based cross-domain authentication for intelligent 5g-enabled internet of drones," *IEEE Internet of Things Journal*, vol. 9, no. 8, pp. 6224–6238, 2021.

[14] F. Wang, J. Cui, Q. Zhang, D. He, C. Gu, and H. Zhong, "Blockchain-based lightweight message authentication for edge-assisted cross-domain industrial internet of things," *IEEE transactions on dependable and secure computing*, vol. 21, no. 4, pp. 1587–1604, 2023.

[15] Y. Tan, J. Wang, J. Liu, and N. Kato, "Blockchain-assisted distributed and lightweight authentication service for industrial unmanned aerial vehicles," *IEEE Internet of Things Journal*, vol. 9, no. 18, pp. 16 928–16 940, 2022.

[16] L. Gupta, R. Jain, and G. Vaszkun, "Survey of important issues in uav communication networks," *IEEE communications surveys & tutorials*, vol. 18, no. 2, pp. 1123–1152, 2015.

[17] G. Bansal and B. Sikdar, "S-maps: Scalable mutual authentication protocol for dynamic uav swarms," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 11, pp. 12 088–12 100, 2021.

[18] B. B. Ehuil, C. Chen, S. Wang, H. Guo, and J. Liu, "A secure mutual authentication protocol based on visual cryptography technique for iot-cloud," *Chinese Journal of Electronics*, vol. 33, no. 1, pp. 43–57, 2024.

[19] C. Lai, J. Ma, X. Wang, H. Zhou, and D. Zheng, "A novel authentication and key agreement scheme for in-vehicle networks," *IEEE Transactions on Vehicular Technology*, 2025.

[20] B. Palaniswamy, S. Camtepe, E. Foo, and J. Pieprzyk, "An efficient authentication scheme for intra-vehicular controller area network," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 3107–3122, 2020.

[21] Y. Zhang, R. H. Deng, E. Bertino, and D. Zheng, "Robust and universal seamless handover authentication in 5g hetnets," *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 2, pp. 858–874, 2019.

[22] M. Pundir and A. Kumar, "An efficient conference key agreement protocol suited for resource constrained devices," *Journal of Parallel and Distributed Computing*, vol. 196, p. 105011, 2025.
[23] K. Xue, C. Ma, P. Hong, and R. Ding, "A temporal-credential-based mutual authentication and key agreement scheme for wireless sensor networks," *Journal of Network and Computer Applications*, vol. 36, no. 1, pp. 316–323, 2013.
[24] R. Zhang, L. Zhang, K.-K. R. Choo, and T. Chen, "Dynamic authenticated asymmetric group key agreement with sender non-repudiation and privacy for group-oriented applications," *IEEE Transactions on Dependable and Secure Computing*, vol. 20, no. 1, pp. 492–505, 2021.
[25] H. Xiong, J. Chen, Q. Mei, and Y. Zhao, "Conditional privacy-preserving authentication protocol with dynamic membership updating for vanets," *IEEE Transactions on Dependable and Secure Computing*, vol. 19, no. 3, pp. 2089–2104, 2020.
[26] G. Liu, H. Li, N. Wang, T. Xiang, and Y. Liu, "Degkm: Decentralized group key management for content push in integrated networks," *IEEE Transactions on Dependable and Secure Computing*, vol. 21, no. 5, pp. 4784–4800, 2024.
[27] D. Wang, Y. Cao, K.-Y. Lam, Y. Hu, and O. Kaiwartya, "Authentication and key agreement based on three factors and puf for uavs-assisted post-disaster emergency communication," *IEEE Internet of Things Journal*, 2024.
[28] Q. Zhang, J. Wu, H. Zhong, D. He, and J. Cui, "Efficient anonymous authentication based on physically unclonable function in industrial internet of things," *IEEE Transactions on Information Forensics and Security*, vol. 18, pp. 233–247, 2022.
[29] Z. Guo, J. Cao, X. Wang, Y. Zhang, B. Niu, and H. Li, "Uava: Unmanned aerial vehicle assisted vehicular authentication scheme in edge computing networks," *IEEE Internet of Things Journal*, 2024.
[30] Z. Zhang, X. Li, Y. Wang, Y. Miao, X. Liu, J. Weng, and R. H. Deng, "Tagka: threshold authenticated group key agreement protocol against member disconnect for uanet," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 11, pp. 14 987–15 001, 2023.
[31] A. HE and H. Xiao, "A group key agreement protocol for vanet based on chinese remainder theorem and blockchain," *Authorea Preprints*, 2023.
[32] S. Yu, A. K. Das, and Y. Park, "Rlba-uav: A robust and lightweight blockchain-based authentication and key agreement scheme for puf-enabled uavs," *IEEE Transactions on Intelligent Transportation Systems*, 2024.
[33] K. Huang, H. Hu, and C. Lin, "Bakas-uav: A secure blockchain-assisted authentication and key agreement scheme for unmanned aerial vehicles networks," *IEEE Internet of Things Journal*, 2024.
[34] F. Wang, J. Cui, Q. Zhang, D. He, and H. Zhong, "Blockchain-based secure cross-domain data sharing for edge-assisted industrial internet of things," *IEEE Transactions on Information Forensics and Security*, vol. 19, pp. 3892–3905, 2024.
[35] Q. Xie, Z. Ding, W. Tang, D. He, and X. Tan, "Provable secure and lightweight blockchain-based v2i handover authentication and v2v broadcast protocol for vanets," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 12, pp. 15 200–15 212, 2023.
[36] P. Kumar, A. Braeken, A. Gurtov, J. Iinatti, and P. H. Ha, "Anonymous secure framework in connected smart home environments," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 4, pp. 968–979, 2017.
[37] M. Wazid, A. K. Das, V. Odelu, N. Kumar, and W. Susilo, "Secure remote user authenticated key establishment protocol for smart home environment," *IEEE Transactions on Dependable and Secure Computing*, vol. 17, no. 2, pp. 391–406, 2017.
[38] H. Zhang, X. Li, S.-Y. Tan, M. J. Lee, and Z. Jin, "Privacy-preserving biometric authentication: Cryptanalysis and countermeasures," *IEEE Transactions on Dependable and Secure Computing*, vol. 20, no. 6, pp. 5056–5069, 2023.
[39] M. Abdalla, P.-A. Fouque, and D. Pointcheval, "Password-based authenticated key exchange in the three-party setting," in *International workshop on public key cryptography*. Springer, 2005, pp. 65–84.
[40] M. Qaisi, S. Althunibat, and M. Qaraqe, "Phase-assisted dynamic tag-embedding message authentication for iot networks," *IEEE Internet of Things Journal*, vol. 9, no. 20, pp. 20 620–20 629, 2022.
[41] A. K. Sutrala, M. S. Obaidat, S. Saha, A. K. Das, M. Alazab, and Y. Park, "Authenticated key agreement scheme with user anonymity and untraceability for 5g-enabled softwarized industrial cyber-physical systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 3, pp. 2316–2330, 2021.
[42] M. A. Saleem, X. Li, M. F. Ayub, S. Shamshad, F. Wu, and H. Abbas, "An efficient and physically secure privacy-preserving key-agreement protocol for vehicular ad-hoc network," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 9, pp. 9940–9951, 2023.