Article

Bio-2FA-IoD: A Biometric-Enhanced Two-Factor Authentication Protocol for Secure Internet of Drones Operations

Hyunseok Kim *

Posted Date: 5 June 2025

doi: 10.20944/preprints202506.0422.v1

Keywords: Internet of Drones (IoD); UAV Security; Biometric Authentication; Two-Factor Authentication (2FA); Fuzzy Extractor; BAN Logic; BPR Model; Lightweight Authentication; Secure Communication

 doi:10.20944/preprints202506.0422.v1

Article

Bio-2FA-IoD: A Biometric-Enhanced Two-Factor Authentication Protocol for Secure Internet of Drones Operations

Hyunseok Kim †

Dept. of Information and Security at ICT Polytech Institute of Korea; hskim@ict.ac.kr; Tel.: +82-031-760-3391 (F.L.)
† 16-26 Sunamro, Gwangjusi, Gyunggido: ICT Polytehch Institute of Korea

Abstract: The Internet of Drones (IoD) is increasingly utilized in sensitive applications, demanding robust authentication mechanisms. Traditional authentication methods face challenges from various attacks, and the unique operational context of IoD, including potential drone capture, necessitates advanced security measures. This paper proposes a Biometric-Enhanced Two-Factor Authentication Protocol for IoD (Bio-2FA-IoD), drawing inspiration from established principles in two-factor authentication and leveraging recent advancements in biometric security. The protocol aims to provide strong mutual authentication between a drone operator (via an operator device), the drone (acting as a relay), and a Ground Control Station (GCS), facilitated by a Trusted Authority (TA). We detail the registration and authentication phases, integrating fuzzy extractors for reliable biometric key generation, a technique proven effective in various secure systems. The security of Bio-2FA-IoD is then analyzed using BAN (Burrows-Abadi-Needham) logic to demonstrate the establishment of shared beliefs and authenticated key agreement, and through the Bellare-Pointcheval-Rogaway (BPR) model to formally prove its security against active adversaries in the Authenticated Key Exchange (AKE) context. A comparative performance evaluation highlights the protocol's efficiency in terms of computational and communication costs, positioning it as a viable solution for resource-constrained IoD environments.

Keywords: Internet of Drones (IoD); UAV security; biometric authentication; two-factor authentication (2FA); fuzzy extractor; BAN logic; BPR model; lightweight authentication; secure communication

1. Introduction

The proliferation of Unmanned Aerial Vehicles (UAVs), or drones, within the Internet of Drones (IoD) paradigm has significantly expanded their application scope, from civilian tasks like surveillance and logistics to critical military operations [7,8,12,13]. This expansion, however, brings forth substantial security challenges. IoD systems predominantly rely on open wireless channels for communication, making them susceptible to a multitude of threats such as eavesdropping, impersonation, Man-in-the-Middle (MitM) attacks, and replay attacks [6,9]. Furthermore, the physical nature of drones means they can be deployed in hostile or unattended areas, increasing the risk of physical capture and subsequent compromise of any stored sensitive information. Traditional password-based authentication, while common, often falls short in such environments due to vulnerabilities like keylogging and shoulder-surfing, especially when operators interact with ground control systems through potentially insecure terminals.

To address these evolving security needs, particularly in resource-constrained IoD environments, there is a compelling need for authentication protocols that are both robust and efficient. Two-factor authentication (2FA) principles, combining different types of credentials (e.g., something you know, something you have, something you are), offer a significant security enhancement over single-factor methods. Biometric authentication, which utilizes unique physiological or behavioral characteristics of an individual, stands out as a strong candidate for the "something you are" factor, providing inherent user verification and resistance to observational attacks. Recent research in IoD and related fields has

 doi:10.20944/preprints202506.0422.v1

explored various methods to integrate biometrics, often with fuzzy extractors to handle the inherent variability of biometric data, and lightweight cryptographic primitives to ensure efficiency [6,11,15].

This paper introduces **Bio-2FA-IoD**, a novel Biometric-Enhanced Two-Factor Authentication Protocol designed specifically for the IoD ecosystem. Instead of relying on visual mechanisms like QR codes, which can be impractical for direct drone or operator-to-drone interactions in dynamic field conditions, Bio-2FA-IoD leverages secure biometric verification managed through a dedicated Operator's Device (OD). This approach aims to provide strong user authentication while mitigating risks associated with traditional password entry and physical observation. The protocol architecture involves an Operator (U_O), their personal OD, a Drone (D_i) primarily acting as a communication relay, a Ground Control Station (GCS), and a Trusted Authority (TA) for initial registration and trust establishment. The design prioritizes the security of operator credentials and the efficiency of the authentication process, making it suitable for the often resource-limited nature of IoD components [7,11].

Contributions of this paper include:

1. The design of a novel two-factor authentication protocol for IoD environments (Bio-2FA-IoD) that integrates biometrics as a primary operator verification factor, managed through a trusted Operator's Device. This approach is inspired by the need for strong, user-bound authentication that is resilient to observational attacks.
2. Detailed registration and authentication phases that incorporate fuzzy extractors [2] for robust and reliable biometric key generation, a critical component for practical biometric systems [6,10].
3. A formal security analysis of the proposed protocol using Burrows-Abadi-Needham (BAN) logic [3] to rigorously verify the achievement of mutual authentication and the secure establishment of shared beliefs regarding key parameters between communicating entities.
4. A formal proof of security for the derived session key within the Bellare-Pointcheval-Rogaway (BPR) model [4] for Authenticated Key Exchange (AKE), demonstrating the protocol's resilience against a wide range of active adversarial attacks.
5. A comprehensive performance evaluation, comparing Bio-2FA-IoD with several contemporary IoD authentication protocols in terms of computational costs, communication overhead, and qualitative energy efficiency, thereby highlighting its suitability for resource-constrained IoD deployments.

The remainder of this paper is organized as follows: Section 2 reviews related work in IoD authentication, focusing on biometric and lightweight approaches. Section 3 details the system and threat models, crucial security requirements for IoD, the cryptographic primitives employed (including fuzzy extractors), and the formal security models (BAN logic and BPR model) used for analysis. Section 4 presents the Bio-2FA-IoD protocol in detail, outlining the registration and authentication phases. Section 5 provides a comprehensive security assessment, encompassing an informal analysis against various attacks, the BAN logic verification, and the BPR model proof sketch. Section 6 evaluates and discusses the performance of the proposed protocol. Finally, Section 7 concludes the paper and outlines potential directions for future research, followed by the list of references in Section 7.

2. Related Work

The quest for secure and efficient authentication in Internet of Drones (IoD) has spurred significant research, moving beyond traditional password-based systems. While Khedr's work on visual two-factor authentication [1] effectively tackled keylogging and shoulder-surfing using QR codes and smartphones in general computing environments, its direct applicability to IoD is limited due to the impracticality of QR code scanning in typical drone operational scenarios.

Recent IoD authentication research has increasingly focused on integrating stronger, user-bound factors and lightweight cryptographic primitives suitable for resource-constrained UAVs. Biometrics, as a "something you are" factor, has emerged as a key technology. Nyangaresi et al. [6] proposed

 doi:10.20944/preprints202506.0422.v1

an IoD protocol combining biometrics with Physically Unclonable Functions (PUFs), emphasizing resistance to drone capture and utilizing the RoR model for formal security. This work underscores the value of hardware-based security and biometrics in mitigating physical threats. Khan et al. [15] designed an ECC-based mutual authentication scheme for smart grids that incorporates biometrics with fuzzy extractors, demonstrating the feasibility of such an approach in related distributed systems. The use of fuzzy extractors to manage the inherent noise in biometric data is a common theme in robust biometric authentication [2,10,21].

Lightweight design is paramount in IoD. Jan et al. [7] developed a protocol based on HMACSHA1, focusing on minimal computational and communication overhead, and formally verified its security using ROM and ProVerif. Their approach highlights the preference for symmetric key cryptography and hash functions in resource-sensitive environments. Similarly, Najafi et al. [11] explored DRAM PUFs with entropy-derived features for lightweight authentication in general IoT, which shares similar constraints with IoD.

Several other protocols address specific IoD challenges. Khalid et al. [9] presented HOOPOE, an anonymous handover authentication protocol using AES-RSA, formally verified with the RoR model and ProVerif, addressing security during drone mobility between zones. Berini et al. [8] introduced HCALA, which employs Hyperelliptic Curve Cryptography (HECC) and blockchain for anonymous authentication in IoD, also verified using ROM and AVISPA. Blockchain has also been explored by Akram et al. for privacy-preserving authentication [10] and for general IoDT authentication frameworks [16]. These blockchain-based approaches aim to provide decentralization and tamper-resistance, though often at the cost of increased overhead compared to non-blockchain solutions.

The security of IoD protocols is continuously scrutinized. For instance, Jafarian [26] provided a cryptanalysis of Nikooghadam et al.'s lightweight IoD authentication scheme [25], identifying vulnerabilities such as user tracking and stolen verifier attacks. This emphasizes the need for rigorous security analysis, including both formal proofs and informal assessments against a broad range of attack vectors. Other relevant works include efficient three-factor authentication for IoD by Zhang et al. [20] and various ECC-based schemes [18,19].

Our proposed Bio-2FA-IoD protocol draws inspiration from the need for strong, two-factor authentication that is resilient to observational attacks, as emphasized by Khedr [1]. However, it diverges by replacing the visual QR code component with a biometric factor managed by the Operator's Device, making it more aligned with IoD practicalities. It focuses on secure operator-to-GCS authentication using lightweight symmetric key operations and fuzzy extractors, differentiating itself from more complex PUF-based, ECC/HECC-based, or blockchain-centric solutions by aiming for a balance of strong security against specific threats (keylogging, shoulder-surfing, basic impersonation) with high efficiency. The dual formal verification using BAN logic and the BPR model aims to provide a high degree of confidence in its security claims within its defined scope.

3. Preliminaries

This section details the foundational concepts, models, and cryptographic primitives underpinning the proposed Bio-2FA-IoD protocol. A clear understanding of these preliminaries is essential for comprehending the design and security analysis of our protocol.

3.1. System and Network Model for IoD

The Internet of Drones (IoD) paradigm typically involves a network of Unmanned Aerial Vehicles (UAVs) interacting with various ground entities to perform tasks such as surveillance, data collection, and delivery [9,12,19,22]. The specific architecture for our Bio-2FA-IoD protocol includes the following key entities (visualized in Figure 1):

- **Operator (U_O)**: The legitimate human user who initiates control commands or data requests for a drone. The operator's identity is primarily verified through biometric means.

 doi:10.20944/preprints202506.0422.v1

- **Operator's Device (OD)**: A trusted personal device (e.g., smartphone, dedicated handheld controller, tablet) belonging to the operator. It is equipped with, or can interface with, a biometric sensor. The OD is responsible for capturing biometric data, performing initial processing (e.g., feature extraction, fuzzy key generation), securely storing operator-specific credentials derived during registration, and executing the client-side authentication logic. This device is analogous to the smartphone in Khedr's visual authentication protocol [1].
- **Drone (D_i)**: The Unmanned Aerial Vehicle. In the context of operator authentication to the GCS, the drone (D_i) primarily functions as a mobile platform that facilitates communication between the OD and the GCS. While drones possess their own identities and security mechanisms for flight control and drone-to-drone communication (which are outside the scope of this specific operator authentication protocol), their role here is to securely forward authentication messages between the OD and the GCS. Drones typically operate within specific flight zones and are managed by a GCS [8]. Drone technology integrates various components like communication modules (e.g., for FANETs [7,24]), sensors, and actuators, making them versatile but also complex systems to secure.
- **Ground Control Station (GCS)**: A server entity, often part of a larger ground infrastructure or cloud backend. The GCS is responsible for receiving authentication requests from operators (via OD/Drone), verifying operator credentials, managing drone operations, and logging relevant activities. It is analogous to the "Server" in Khedr's protocol [1] and the "Control Server (CS)" or "Ground Station Server (GSS)" in other IoD literature [8,10,19]. The GCS is assumed to have significantly more computational and storage resources than drones or ODs. It often connects to a data processing center.
- **Trusted Authority (TA)**: A fully trusted, offline or highly secured entity responsible for the initial system setup and registration of all legitimate participants: operators (and their biometric data), ODs (by issuing initial secrets during operator registration), and GCSs. The TA manages master keys, identities, and ensures the integrity of the registration process [8,22]. It does not participate in every authentication session but establishes the root of trust.

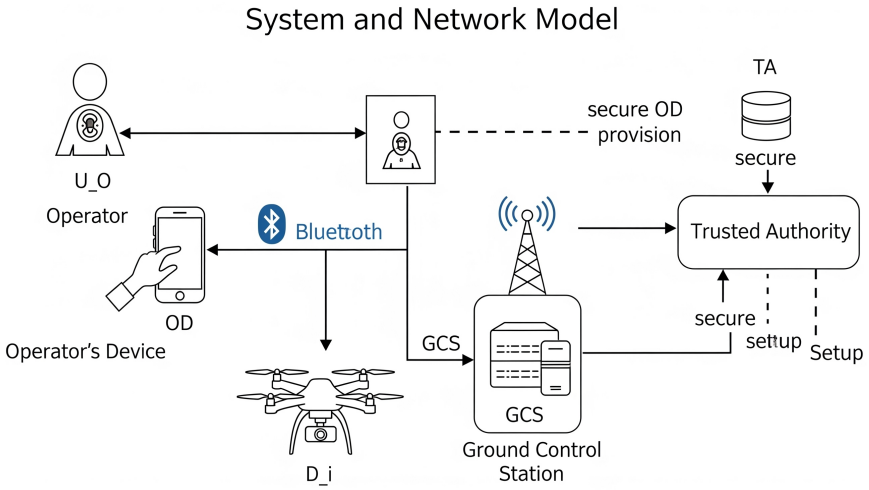

Figure 1. System and Network Model for Bio-2FA-IoD. This figure illustrates the key entities: Operator (U_O), Operator's Device (OD), Drone (D_i), Ground Control Station (GCS), and Trusted Authority (TA). It depicts the primary communication links: (1) Operator's physical interaction with OD for biometric input. (2) Local secure link between OD and Drone. (3) Public wireless channel between Drone and GCS. (4) Registration/setup phase interactions with the TA by OD (via Operator) and GCS.

Communication pathways are as follows: The U_O interacts directly with the OD for biometric input. The OD communicates with D_i, typically over a short-range wireless link (e.g., Bluetooth Low Energy, local Wi-Fi). This link must be secured, or its exposure minimized, as it's a crucial step in

 doi:10.20944/preprints202506.0422.v1

relaying authentication data. The D_i then communicates with the GCS over potentially long-range and insecure public wireless channels (e.g., cellular networks like 5G [19], dedicated IoD frequencies). The TA interacts with entities primarily during the offline registration phase through secure channels. Figure 1 provides a visual representation of these entities and their interactions.

3.2. Threat Model

We adopt a comprehensive adversary model based on the Dolev-Yao (DY) model [5,7], extended with considerations relevant to IoD and biometric systems, similar to threat models discussed in [6,8,17,22]. The adversary ($\mathcal{A}$) is assumed to have the following capabilities:

- **Full Control over Public Communication Channels**: $\mathcal{A}$ can eavesdrop on, intercept, delete, modify, inject, and replay any messages transmitted over insecure wireless links (e.g., D_i-GCS, and potentially OD-D_i if the local link is not perfectly secured).
- **Impersonation Attempts**: $\mathcal{A}$ can try to impersonate any legitimate entity (U_O, OD, D_i, GCS) by replaying old messages or attempting to construct valid-looking new messages using any information gathered.
- **Malicious Node Participation**: $\mathcal{A}$ can be a registered but compromised entity (e.g., a captured drone whose limited credentials might be extracted, or a compromised OD). The TA is assumed to be incorruptible.
- **Drone Capture and Physical Attacks**: Drones are vulnerable to physical capture, especially if operating in unattended or hostile environments. Upon capture, $\mathcal{A}$ may perform physical attacks, side-channel attacks, or power analysis to extract any sensitive data stored on the drone's hardware [6,9,19]. The protocol design aims to minimize the impact of such a compromise by limiting the sensitive data stored directly on the drone for operator authentication.
- **Operator Device Attacks**: While the OD is trusted by the operator, it can be subject to attacks if it relies on traditional password/PIN entry as a fallback or secondary factor (e.g., keylogging, shoulder-surfing) [1]. The use of biometrics as the primary factor mitigates this, and mechanisms like Khedr's SVOSK [1] can further protect any manual input on the OD. Stolen ODs are a threat if biometric authentication can be bypassed or stored encrypted secrets are weak.
- **Server-Side Database Attacks (GCS/TA Verification Data)**: $\mathcal{A}$ might attempt to compromise the GCS or TA databases to obtain stored verification data (e.g., $RPMD_{ID_O}$, $Salt_1$, LLT_O). The protocol should ensure that such a breach does not directly reveal primary secrets like P_{ID_O} or the operator's biometric template [26].
- **Biometric System Attacks**: $\mathcal{A}$ may attempt to spoof the biometric sensor on the OD with a fake biometric sample (presentation attack) or attack the fuzzy extractor mechanism (e.g., by analyzing helper data HD_O). The inherent security of the biometric sensor (e.g., liveness detection) and the properties of the fuzzy extractor are critical assumptions.

The TA is assumed secure. The GCS, while a critical component, is designed such that a compromise of its stored verification data does not lead to an immediate compromise of the operator's primary secrets (P_{ID_O}, $K_{\beta O}$). The cryptographic primitives (hash functions, symmetric encryption) are assumed to be computationally secure, meaning an adversary cannot break their fundamental security properties (e.g., find hash collisions, decrypt ciphertexts without the key) within a polynomial time frame.

3.3. Security Requirements for Bio-2FA-IoD

The proposed Bio-2FA-IoD protocol is designed to meet the following critical security and functional requirements essential for a robust IoD authentication system. These requirements are drawn from a consensus in existing IoD security literature [6–9,11–13,19]:

- **Mutual Authentication**: All communicating IoD entities (e.g., operator via OD, and GCS) must verify each other's identity before any sensitive payload exchanges or command execution can

 doi:10.20944/preprints202506.0422.v1

occur. This is a fundamental requirement to prevent unauthorized access and ensure that all parties are legitimate.

- **Keylogging and Shoulder-Surfing Resistance**: The protocol should protect operator credentials (like passwords or PINs, if used as a secondary factor to biometrics) from being captured by keyloggers on potentially compromised GCS terminals or observed by nearby attackers during input on the Operator's Device (OD) [1].
- **Replay Attack Resistance**: Old authentication messages intercepted by an adversary must not be successfully replayed to gain unauthorized access or disrupt operations. This is often achieved using fresh nonces or timestamps [7].
- **Impersonation Attack Resistance**: An adversary should not be able to successfully impersonate a legitimate operator, OD, drone, or GCS. This includes user impersonation, drone impersonation, and GSS/server impersonation [8].
- **Drone Capture Resilience**: Given that drones may be deployed in unattended or hostile locations, they are susceptible to physical capture. The compromise of a single drone (and any secrets stored on it) should not compromise the operator's long-term credentials, the security of other drones, or past/future session keys for other entities. Protocols should be designed to minimize sensitive data stored on the drone or ensure such data is heavily protected [6,9].
- **Session Key Agreement and Security**: Upon successful mutual authentication, the communicating parties (OD and GCS) should agree upon a secure session key (KS in our context, primarily used for OTAC) to encrypt critical parts of their authentication exchange. This key must be protected from disclosure and should be unique to each session.
- **Confidentiality**: Sensitive data exchanged during the authentication process (e.g., PIN, RP_{ID_O}) and subsequent communications (if applicable) must be kept confidential from eavesdroppers.
- **Integrity**: The protocol must provide means to ensure that messages exchanged between entities are not tampered with or altered by an adversary during transit.
- **Anonymity and Untraceability**: Ideally, the real identities and locations of IoD entities (especially users and drones) should remain unknown to adversaries even if they can eavesdrop on exchanged messages. Attackers should not be able to trace or link communication sessions to specific users or drones over time [6,8,9].
- **Forward and Backward Secrecy**:
  - **Perfect Forward Secrecy (PFS)**: Compromise of long-term secret keys (e.g., P_{ID_O} or $K_{\beta O}$) should not compromise the confidentiality of past session keys (KS).
  - **Perfect Backward Secrecy (PBS)**: Compromise of long-term secret keys should not compromise the confidentiality of future session keys derived after the compromise. The LAPEC protocol [18], for example, specifically aims for backward secrecy.
- **Resistance to Specific IoD Attacks**: The protocol should withstand attacks common to wireless and drone environments, such as: Man-in-the-Middle (MitM) attacks, Denial of Service (DoS) attacks at the protocol level, privileged insider attacks [1,7], stolen verifier attacks [26], Ephemeral Secret Leakage (ESL) attacks, and physical/side-channel attacks.
- **Efficiency (Lightweight Design)**: Given the resource constraints (CPU, memory, battery) of drones and potentially ODs, the authentication protocol must be lightweight. This implies minimizing complex cryptographic operations and reducing communication overhead [7,11,18].
- **Scalability**: The authentication mechanism should be able to support a potentially large number of drones and users without significant performance degradation or unmanageable key distribution overhead.
- **Flexibility and Usability**: The protocol should be adaptable to various IoD operational scenarios and be user-friendly for the operator during the authentication process. This includes considerations for password/PIN changes and device revocation.

 doi:10.20944/preprints202506.0422.v1

- **Dynamic Operations Support**: The protocol should ideally support dynamic drone addition to the network and handle drone revocation or temporary disconnections and re-authentication seamlessly. Handover authentication [9] is also critical.

3.4. Cryptographic Primitives

The Bio-2FA-IoD protocol relies on standard and well-vetted cryptographic primitives.

3.4.1. Hash Functions

A cryptographic hash function $H : \{0,1\}^* \rightarrow \{0,1\}^l$ maps an input of arbitrary length to a fixed-length output string, known as a hash value or message digest [1,6]. For a hash function to be cryptographically secure, as detailed in [6–8], it must satisfy:

- **Pre-image Resistance (One-wayness)**: Given a hash value y, it is computationally infeasible to find any input x such that $H(x) = y$.
- **Second Pre-image Resistance (Weak Collision Resistance)**: Given an input x, it is computationally infeasible to find a different input $x' \neq x$ such that $H(x') = H(x)$.
- **Collision Resistance (Strong Collision Resistance)**: It is computationally infeasible to find any two distinct inputs x and x' such that $H(x) = H(x')$.

Our protocol assumes the use of a standard secure hash algorithm such as SHA-256 (Secure Hash Algorithm 256-bit), as recommended in NIST FIPS PUB 180-4 [1]. In formal security models like the BPR model, hash functions are often modeled as random oracles.

3.4.2. Symmetric Key Cryptography

Symmetric key cryptography involves the use of a single secret key K for both encryption $E_K(\cdot)$ and decryption $D_K(\cdot)$ operations [1].

- Encryption: $C = E_K(M)$, where M is the plaintext and C is the ciphertext.
- Decryption: $M = D_K(C)$.

The security of symmetric encryption relies entirely on the secrecy of the key K. The chosen algorithm (e.g., AES-128 - Advanced Encryption Standard with 128-bit key, specified in FIPS PUB 197 [1]) should be resistant to known cryptanalytic attacks and provide semantic security (IND-CPA: Indistinguishability under Chosen-Plaintext Attack). Our protocol uses AES-128 for OTAC encryption.

3.4.3. Key Derivation Functions (KDFs)

A Key Derivation Function (KDF) is used to derive one or more cryptographically secure secret keys from a master secret value, such as a pre-shared key, a password, or a Diffie-Hellman exchanged value, often in conjunction with a salt [1].

- $K_{derived} = \text{KDF}(MasterKey, Salt, ContextInfo, OutputLength)$

KDFs are designed to be computationally intensive to slow down brute-force attacks on the input master secret (especially if it's a low-entropy password). They should produce outputs that are pseudorandom and indistinguishable from truly random keys. Our protocol uses a KDF based on a secure hash function (e.g., PBKDF2 as defined in RFC 2898 [1], using HMAC-SHA-256 as the underlying pseudorandom function) for deriving keys like K_O and KS.

3.5. Biometrics and Fuzzy Extractor

Biometric authentication leverages unique physiological (e.g., fingerprint [14], iris) or behavioral (e.g., voice, signature) characteristics of an individual for identity verification [6,15]. A significant challenge with biometric data is its inherent variability; readings of the same biometric trait are rarely identical due to sensor noise, environmental conditions, or physiological changes. Key performance metrics for biometric systems include False Acceptance Rate (FAR) and False Rejection Rate (FRR). A **Fuzzy Extractor**, introduced by Dodis et al. [2], is a cryptographic tool designed to address this issue by reliably extracting a stable, uniformly random cryptographic key from noisy biometric inputs.

 doi:10.20944/preprints202506.0422.v1

Many IoD and related security protocols acknowledge its utility [6,10,15,21]. It generally consists of two main algorithms:

- **Generation (*Gen*)**: During the enrollment phase, this probabilistic algorithm takes an initial high-quality biometric sample β as input. It outputs a secret cryptographic key K_β (which should be close to uniformly random if β has sufficient min-entropy) and public helper data HD. Schematically: $Gen(\beta) \rightarrow (K_\beta, HD)$. The helper data HD is stored publicly and is used during the key reproduction phase. It is crucial that HD does not reveal significant information about either β or K_β.
- **Reproduction (*Rec*)**: During an authentication attempt, this deterministic algorithm takes a fresh (potentially noisy) biometric sample β' from the user and the stored public helper data HD as input. It attempts to reproduce the original cryptographic key K_β. Schematically: $Rec(\beta', HD) \rightarrow K_\beta$. The reproduction is successful if the Hamming distance (or another suitable metric) between the enrollment template β and the current sample β' is within a predefined error tolerance threshold τ, i.e., $d(\beta, \beta') \leq \tau$.

The security of a fuzzy extractor relies on the assumptions that (a) the biometric data β has sufficient entropy, and (b) the helper data HD does not leak information about K_β beyond what is inherently necessary to correct minor variations in β'.

3.6. Formal Security Models

3.6.1. BAN Logic

Burrows-Abadi-Needham (BAN) logic [3] is a widely used modal logic of belief specifically designed for the formal analysis of authentication protocols. Its primary purpose is to determine whether the principals involved in a protocol can logically deduce each other's identities, establish shared secrets, and believe in the freshness of the exchanged messages and keys. BAN logic operates on idealized versions of protocol messages and relies on a set of initial assumptions and inference rules. Key constructs and some relevant rules, as often cited in similar security papers [14,22], include:

- **Beliefs**: $P \mid\equiv X$ (Principal P believes statement X).
- **Sees**: $P \triangleleft X$ (P sees/receives message X).
- **Once Said**: $P \mid\sim X$ (P once said X).
- **Freshness**: $\#(X)$ (Formula X is fresh, e.g., contains a fresh nonce or timestamp).
- **Shared Secret Key**: $P \overset{K}{\leftrightarrow} Q$ (P and Q share a secret key K).
- **Jurisdiction**: $P \mid\Rightarrow X$ (P has jurisdiction over statement X).
- **Message Meaning Rule (for shared keys)**: If $P \mid\equiv P \overset{K}{\leftrightarrow} Q$ and $P \triangleleft \{X\}_K$, then $P \mid\equiv Q \mid\sim X$.
- **Nonce Verification Rule**: If $P \mid\equiv \#(X)$ and $P \mid\equiv Q \mid\sim X$, then $P \mid\equiv Q \mid\equiv X$.
- **Jurisdiction Rule**: If $P \mid\equiv Q \mid\Rightarrow X$ and $P \mid\equiv Q \mid\equiv X$, then $P \mid\equiv X$.

BAN logic helps identify logical flaws in protocols but does not prove security against all types of attacks (e.g., computational attacks). It is often used as a first step in formal analysis.

3.6.2. BPR Model (Bellare-Pointcheval-Rogaway)

The Bellare-Pointcheval-Rogaway (BPR) model [4], and its variants like the Real-or-Random (RoR) model employed in several contemporary IoD security papers [6–10,26], provide a standard and rigorous framework for proving the semantic security of Authenticated Key Exchange (AKE) protocols against active adversaries, often in the random oracle model. The model typically defines:

- **Participants (Oracles)**: Legitimate protocol participants (U_O via OD, GCS in our case) are modeled as oracles. An oracle Π_U^s represents instance s of principal U engaging in a protocol session.
- **Adversary ($\mathcal{A}$)**: A probabilistic polynomial-time (PPT) adversary interacts with these oracles. $\mathcal{A}$ has full control over the communication network (as per Dolev-Yao [5]). Its capabilities are modeled through a set of allowed oracle queries. Based on similar models in [6] and the original BPR model, common queries include:

 doi:10.20944/preprints202506.0422.v1

- 'Send(Π_U^s, M)': $\mathcal{A}$ sends message M to instance Π_U^s and receives the protocol-defined response.
- 'Execute($\Pi_U^{s_1}, \Pi_V^{s_2}$)': $\mathcal{A}$ eavesdrops on an honest execution of the protocol.
- 'Reveal(Π_U^s)': $\mathcal{A}$ obtains the session key computed by instance Π_U^s.
- 'Corrupt(U)': $\mathcal{A}$ obtains all long-term secret keys of principal U.
- 'Hash(M)': $\mathcal{A}$ queries the random oracle for the hash function $H(M)$ or KDF.
- 'FuzzyExtractor(Gen/Rec, data)': $\mathcal{A}$ can query fuzzy extractor oracles.
- 'Test(Π_U^s)': Queried once on a "fresh" instance. The oracle flips a secret random bit b. If $b = 1$, it returns the actual session key; if $b = 0$, it returns a random string.

- **Freshness**: An instance Π_U^s is "fresh" if its session key has not been trivially revealed.
- **AKE Security Definition**: An AKE protocol is secure if $Adv_{\text{Prot}}^{AKE}(\mathcal{A}) = |2 \cdot Pr[\mathcal{A} \text{ correctly guesses } b] - 1|$ is negligible. Proofs usually proceed by game-hopping.

3.7. Notations

Table 1 summarizes the key notations used throughout this paper.

Table 1. Notations Used in Bio-2FA-IoD Protocol

Symbol	Meaning
U_O	Drone Operator
OD	Operator's Device
D_i	i-th Drone
GCS	Ground Control Station
TA	Trusted Authority
ID_X	Identity of entity X
β_O	Operator's biometric data (enrollment)
β'_O	Operator's fresh biometric data (verification)
$K_{\beta O}$	Biometric key derived from β_O
HD_O	Helper data for biometric key reconstruction
P_{ID_O}	Operator's password/PIN (local to OD)
K_O	Symmetric key derived from P_{ID_O} and $Salt_2$ on OD
RP_{ID_O}	Random Password for the operator, generated by TA
C_{RP}	RP_{ID_O} encrypted with K_O on OD
$RPMD_{ID_O}$	Hash digest of RP_{ID_O} and $Salt_1$, stored at TA/GCS
$Salt_1, Salt_2$	Random salt values
$H(\cdot)$	Secure one-way hash function (e.g., SHA-256)
$KDF(\cdot,\cdot)$	Key Derivation Function (e.g., PBKDF2)
$E_K(\cdot)$	Symmetric encryption with key K (e.g., AES-128)
$D_K(\cdot)$	Symmetric decryption with key K (e.g., AES-128)
T_{ID}	Timestamp generated by OD
KS	Ephemeral session key for OTAC ($KDF(P_{ID_O}, T_{ID})$)
KS_{GCS}	GCS's version of KS
OTAC	One-Time Authentication Code
PIN	Random number generated by OD, part of OTAC
LLT_O	Operator's last login time, stored at TA/GCS
Δ_t	Maximum allowable time difference for freshness
$\|$	Concatenation operation
$\oplus$	Bitwise XOR operation

3.8. Assumptions of the Protocol

The security and correct functioning of the Bio-2FA-IoD protocol depend on the following underlying assumptions:

1. **Secure TA**: The Trusted Authority (TA) is honest, secure, and will not be compromised. It correctly executes its role during the registration phase, and its long-term secrets remain confidential [1].
2. **Secure Initial Provisioning of OD**: The transfer of initial secrets (RP_{ID_O}, HD_O) from the TA to the Operator's Device (OD) during registration is conducted over a secure channel or within a physically secure environment.

 doi:10.20944/preprints202506.0422.v1

3. **Operator's Device (OD) Trustworthiness and Security**: The OD is considered a trusted computing base for the operator. It is assumed to securely store its provisioned secrets (C_{RP}, HD_O, $Salt_2$) and protect them from malware or unauthorized local access, potentially using hardware-backed security features (e.g., secure element, TEE). The operator's local password/PIN P_{ID_O} (if used) is entered onto the OD through a secure interface (e.g., SVOSK-like [1]).
4. **Biometric System Reliability and Security**:
   - The biometric sensor on the OD is assumed to be resistant to common spoofing attacks (e.g., presentation attacks with fake biometric samples).
   - The fuzzy extractor (Gen, Rec algorithms) is assumed to be correctly implemented and secure: it reliably reproduces $K_{\beta O}$ from a legitimate (though possibly noisy) sample β'_O using HD_O, and the helper data HD_O does not leak computationally useful information about β_O or $K_{\beta O}$ beyond what is necessary for error correction [2,6].
   - The operator's biometric trait β_O possesses sufficient entropy for $K_{\beta O}$ to be a strong cryptographic key.
5. **Cryptographic Primitive Idealness**: The chosen cryptographic hash function $H(\cdot)$ (e.g., SHA-256) behaves as a random oracle (or at least meets standard security properties like collision resistance). The symmetric encryption scheme $E_K(\cdot)/D_K(\cdot)$ (e.g., AES-128) is IND-CPA secure. The Key Derivation Function (KDF) (e.g., PBKDF2) is a secure pseudorandom function.
6. **Secure Local OD-Drone Link (for relay)**: The communication link between the OD and the drone D_i for relaying authentication messages M_1 and M_2 is assumed to be reasonably secure (e.g., encrypted Bluetooth, local Wi-Fi with WPA2/3) or its exposure is limited due to short range.
7. **Loosely Synchronized Clocks**: The OD and the GCS maintain loosely synchronized clocks, allowing for effective timestamp-based replay attack detection within a predefined tolerance window Δ_t [1].
8. **GCS Database Protection**: While the protocol aims to mitigate the impact of a GCS database compromise (e.g., by not storing P_{ID_O} directly), standard security practices are assumed to be in place to protect the GCS and its database of ($RPMD_{ID_O}$, $Salt_1$, LLT_O).

4. Proposed Bio-2FA-IoD Protocol

The proposed protocol enhances Khedr's two-factor scheme [1] by integrating biometrics, thereby extending its applicability to drone operations while aiming to preserve its established resistance against observational attacks.

4.1. Registration Phase

This phase (illustrated in Figure 2) is performed in a secure environment and involves the Operator (U_O), their Operator's Device (OD), and the Trusted Authority (TA).

1. **Operator Enrollment ($U_O \rightarrow TA$)**: The operator U_O initiates the registration process by submitting their chosen identity ID_O to the TA and enrolls their biometric trait β_O using a trusted sensor interfaced with the TA's system.
2. **TA Operations (Credential Generation and Storage)**:
   - The TA uses a fuzzy extractor to generate a stable biometric key $K_{\beta O}$ and public helper data HD_O from the enrolled biometric β_O: $(K_{\beta O}, HD_O) = Gen(\beta_O)$ [2].
   - TA generates a unique, high-entropy random password RP_{ID_O} specifically for this operator, and a random salt $Salt_1$.
   - TA computes a hashed version of this random password: $RPMD_{ID_O} = H(RP_{ID_O} \| Salt_1)$.
   - TA securely stores the tuple $(ID_O, RPMD_{ID_O}, Salt_1, LLT_O = 0)$ in its database, where LLT_O is the operator's last login time, initialized to zero [1].

 doi:10.20944/preprints202506.0422.v1

3. **OD Provisioning ($TA \rightarrow OD$)**: The TA securely transmits the generated random password RP_{ID_O} and the helper data HD_O to the operator's designated OD. This transfer must occur over a secure channel or through a secure physical provisioning process.
4. **OD Final Setup (with U_O assistance)**:
   - The operator U_O is prompted to create a local password or PIN, denoted as P_{ID_O}, on their OD.
   - The OD derives a symmetric key $K_O = KDF(P_{ID_O}, Salt_2)$, where $Salt_2$ is another random salt.
   - The OD encrypts the received random password RP_{ID_O} using K_O: $C_{RP} = E_{K_O}(RP_{ID_O})$.
   - The OD securely stores C_{RP}, HD_O, and $Salt_2$.

Drone D_i is also registered with TA/GCS and provisioned with its own credentials.

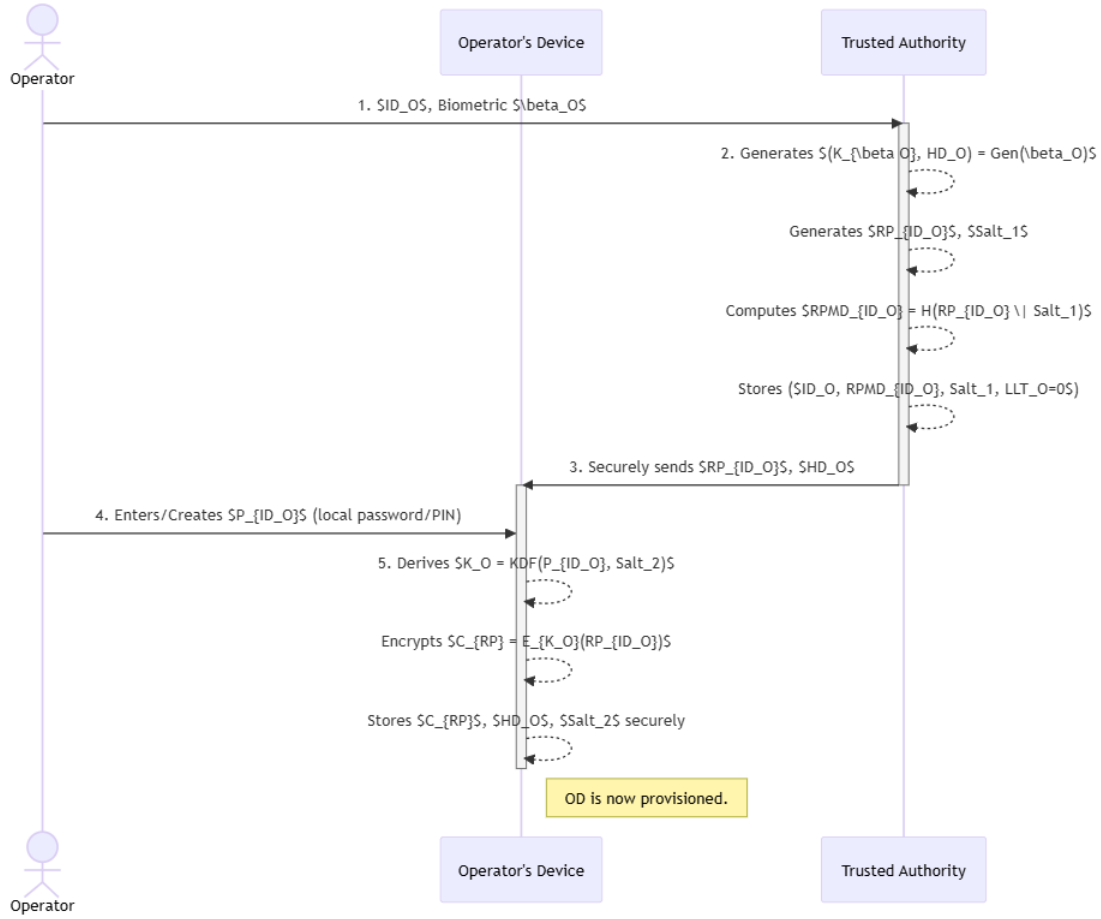

Figure 2. Bio-2FA-IoD - Registration Phase. Interactions between Operator (U_O), Operator's Device (OD), and Trusted Authority (TA) for initial setup and credential provisioning.

4.2. Authentication and Key Exchange Phase

This phase (illustrated in Figure 3) describes how the operator U_O (via their OD) authenticates to the GCS, potentially using a drone D_i as an intermediary communication link.

 doi:10.20944/preprints202506.0422.v1

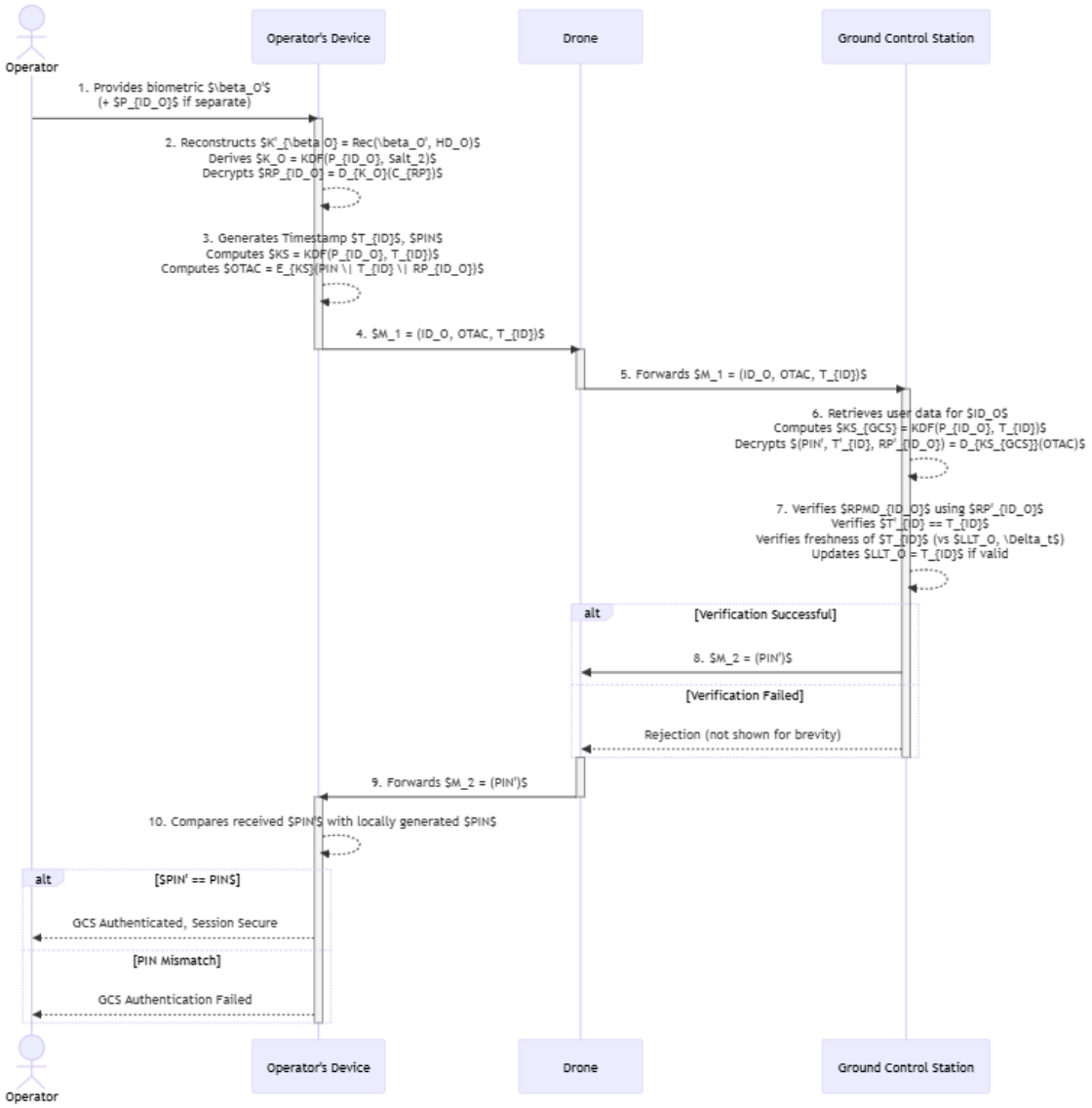

Figure 3. Bio-2FA-IoD - Authentication & Key Exchange Phase. Interactions between Operator (U_O), OD, Drone (D_i), and Ground Control Station (GCS) for mutual authentication.

1. **Operator Biometric Input and Credential Activation ($U_O \rightarrow OD$)**:
   - The operator U_O provides a fresh biometric sample β'_O to the biometric sensor on their OD.
   - The OD uses the stored helper data HD_O to reconstruct the biometric key: $K'_{\beta O} = Rec(\beta'_O, HD_O)$ [2]. If reconstruction fails, the process aborts.
   - Operator U_O enters P_{ID_O}. The OD regenerates $K_O = KDF(P_{ID_O}, Salt_2)$.
   - The OD decrypts $RP_{ID_O} = D_{K_O}(C_{RP})$. If decryption fails, the process aborts.
2. **OD: OTAC Generation**:
   - The OD generates current timestamp T_{ID} and a fresh random nonce PIN.
   - OD generates ephemeral session key $KS = KDF(P_{ID_O}, T_{ID})$.
   - OD computes $OTAC = E_{KS}(PIN \| T_{ID} \| RP_{ID_O})$.
3. **Authentication Request ($OD \rightarrow D_i \rightarrow GCS$)**:
   - OD sends $M_1 = (ID_O, OTAC, T_{ID})$ to Drone D_i.
   - Drone D_i forwards M_1 to the GCS.
4. **GCS: Verification Process**:

 doi:10.20944/preprints202506.0422.v1

- Upon receiving M_1, GCS retrieves $(RPMD_{ID_O}, Salt_1, LLT_O)$ for ID_O.
- GCS computes $KS_{GCS} = KDF(P_{ID_O}, T_{ID})$.
- Decrypts $(PIN', T'_{ID}, RP'_{ID_O}) = D_{KS_{GCS}}(OTAC)$.
- Verifies $H(RP'_{ID_O} \| Salt_1) == RPMD_{ID_O}$.
- Verifies $T'_{ID} == T_{ID}$.
- Verifies $T_{ID} > LLT_O$ and $T_{current\_GCS} - T_{ID} \leq \Delta_t$.

5. **Mutual Authentication Response ($GCS \rightarrow D_i \rightarrow OD$)**:
   - If all checks pass, GCS updates $LLT_O = T_{ID}$, sends $M_2 = (PIN')$ to D_i.
   - Drone D_i forwards M_2 to the OD.
6. **OD: Final Verification (Mutual Authentication)**:
   - OD receives M_2 containing PIN'.
   - OD compares PIN' with its original PIN. If match, GCS is authenticated.

5. Security Analysis

This section provides a multi-faceted security analysis of the Bio-2FA-IoD protocol, including an informal assessment against key security requirements, a formal verification using BAN logic, and a proof sketch under the BPR model.

5.1. Informal Security Analysis

We evaluate the protocol against the security requirements outlined in Section 3.3.

- **Mutual Authentication**: Achieved. The OD (for U_O) authenticates to GCS by proving knowledge of P_{ID_O} and RP_{ID_O} (via OTAC). GCS authenticates to OD by returning the correct PIN from OTAC. This meets requirement 1. This is similar to the mechanism in Khedr [1] but adapted for biometrics.
- **Keylogging and Shoulder-Surfing Resistance**: Primary authentication is biometric. Any PIN P_{ID_O} entry on the OD can use an SVOSK-like interface . This meets requirement 2.
- **Replay Attack Resistance**: Timestamps T_{ID} and LLT_O checks at GCS, along with T_{ID} being part of the encrypted OTAC, prevent replay of M_1 messages . This meets requirement 3.
- **Impersonation Attack Resistance**:
  - *Operator/OD Impersonation*: Requires β_O (or $K_{\beta O}$), P_{ID_O}, and C_{RP} to form a valid OTAC. Infeasible without these secrets.
  - *GCS Impersonation*: Requires deriving PIN from OTAC, which needs KS. Impossible for an adversary without P_{ID_O} and current T_{ID}.

  This fulfills requirement 4. Many IoD protocols like HCALA [8] and Jan et al. [7] also focus on strong impersonation resistance.
- **Drone Capture Resilience**: Secrets $P_{ID_O}, RP_{ID_O}, HD_O$ are on the OD, not the drone D_i. Drone capture does not compromise operator credentials for this protocol. This meets requirement 5, a critical aspect discussed in [6].
- **Session Key (KS) Security**: $KS = KDF(P_{ID_O}, T_{ID})$. Security depends on P_{ID_O} (biometrically protected) and fresh T_{ID}. Meets requirement 6.
- **Confidentiality of OTAC Payload**: PIN, T_{ID}, RP_{ID_O} are encrypted within OTAC using KS. Meets requirement 7.
- **Integrity of Authentication Messages**: Implicit integrity via successful decryption and verification of OTAC contents (RP_{ID_O} vs $RPMD_{ID_O}$, T_{ID} consistency) and PIN return. Modification leads to verification failure. Meets requirement 8.
- **User Anonymity and Untraceability**: ID_O is sent in M_1. This is a limitation regarding requirement 9. Future work could use pseudonyms as in [8,9].
- **Perfect Forward Secrecy (PFS)**: KS depends on long-term P_{ID_O}. PFS is not achieved for KS. This is a limitation regarding requirement 10.

 doi:10.20944/preprints202506.0422.v1

- **Resistance to Stolen Verifier Attack**: GCS stores $RPMD_{ID_O}$. Stealing this does not reveal RP_{ID_O} or P_{ID_O} due to hash properties and salting [1]. Jafarian's cryptanalysis [26] highlights the danger of weak verifiers; our $RPMD_{ID_O}$ is based on TA-generated RP_{ID_O}. This addresses part of requirement 11.
- **Efficiency**: Uses symmetric crypto and hashes, suitable for IoD [7,11]. Meets requirement 12.

5.2. BAN Logic Analysis

(Follows standard BAN logic proof structure [3,14,22]). The goal is to show that $OD \mid\equiv OD \overset{KS}{\leftrightarrow} GCS$ and $GCS \mid\equiv GCS \overset{KS}{\leftrightarrow} OD$.

Initial Assumptions (Axioms):

A1. $TA \mid\equiv (ID_O, P_{ID_O}, RP_{ID_O}, Salt_1)$.
A2. $OD \mid\equiv TA \Rightarrow (RP_{ID_O}, HD_O)$.
A3. $GCS \mid\equiv TA \Rightarrow (ID_O, \text{key material for } P_{ID_O}, RPMD_{ID_O}, Salt_1)$.
A4. $U_O \overset{K_{\beta O}}{\leftrightarrow} OD$.
A5. $OD \mid\equiv \mathsf{fresh}(()T_{ID}), OD \mid\equiv \mathsf{fresh}(()PIN)$.
A6. $OD \mid\equiv P_{ID_O}$.
A7. $GCS \mid\equiv P_{ID_O}$.
A8. $OD, GCS \mid\equiv (KDF(S,T) \implies K_{S,T})$.

Idealized Protocol Messages:

M1. $OD \rightarrow GCS : ID_O, T_{ID}, \{PIN, T_{ID}, RP_{ID_O}\}_{KS}$
M2. $GCS \rightarrow OD : \{PIN\}_{KS'}$

Analysis Sketch:

1. OD sends $M1$. OD believes it shares P_{ID_O} with GCS (via TA) to compute KS.
2. GCS receives $M1$. Computes KS_{GCS}. Using Message Meaning Rule on $\{X\}_{KS_{GCS}}$ (where $X = (PIN, T_{ID}, RP_{ID_O})$), $GCS \mid\equiv OD \mid\sim X$.
3. GCS believes T_{ID} is fresh (after checks), hence $\#(X)$. By Nonce Verification, $GCS \mid\equiv OD \mid\equiv X$. GCS verifies RP_{ID_O} from X against stored $RPMD_{ID_O}$.
4. OD receives $M2 = \{PIN'\}_{KS'}$. OD uses its KS. By Message Meaning, $OD \mid\equiv GCS \mid\sim PIN'$.
5. Since $OD \mid\equiv \#(PIN)$ and PIN' matches original PIN, by Nonce Verification, $OD \mid\equiv GCS \mid\equiv PIN$. This confirms GCS processed X correctly.
6. Both OD and GCS can now believe KS is a good shared key: $OD \mid\equiv OD \overset{KS}{\leftrightarrow} GCS$ and $GCS \mid\equiv GCS \overset{KS}{\leftrightarrow} OD$.

The BAN logic analysis suggests successful mutual authentication and shared key belief establishment under its assumptions.

5.3. BPR Model Analysis

We sketch the security proof for the session key $KS = KDF(P_{ID_O}, T_{ID})$ in the BPR model [4], similar to analyses in [6,9,26].

Oracle Setup and Adversarial Capabilities:

Standard oracles: 'Send', 'Execute', 'RevealSessionKey' (for KS), 'Corrupt(OD_O)'(reveals P_{ID_O}, C_{RP}, HD_O), 'Corrupt(GCS)' (reveals $RPMD_{ID_O}$ table), 'Test' (for KS), 'Hash', 'KDF', 'SymEnc/Dec', 'FuzzyExtractor'.

Proof Sketch by Game Hopping:

The proof aims to show $Adv^{AKE}_{\text{Bio-2FA-IoD}}(\mathcal{A})$ is negligible.

 doi:10.20944/preprints202506.0422.v1

- **Game** G_0: Real AKE security game.
- **Game** G_1 **(Random Oracles for Hash/KDF)**: H and KDF are simulated as random oracles. $|Adv_{G_0}(\mathcal{A}) - Adv_{G_1}(\mathcal{A})| \leq q_H^2/2|H| + q_{KDF}^2/2|KDF_{range}|$, which is negligible for secure functions.
- **Game** G_2 **(Fuzzy Extractor Simulation)**: The output $K_{\beta O}$ of $Rec(\beta'_O, HD_O)$ is indistinguishable from random if β_O has sufficient min-entropy and HD_O leaks nothing, or if $\mathcal{A}$ queries with an invalid β'_O. The advantage change is bounded by ϵ_{FE}, the security of the fuzzy extractor against guessing $K_{\beta O}$ from HD_O or predicting output without valid input [2,6]. $|Adv_{G_1}(\mathcal{A}) - Adv_{G_2}(\mathcal{A})| \leq \epsilon_{FE}$.
- **Game** G_3 **(Symmetric Cipher Indistinguishability)**: Ciphertexts produced by E_{KS} are replaced by random strings if KS is unknown to $\mathcal{A}$. If KS is known, the simulation is perfect. The difference is bounded by IND-CPA security of E, ϵ_{SymEnc}. $|Adv_{G_2}(\mathcal{A}) - Adv_{G_3}(\mathcal{A})| \leq \epsilon_{SymEnc}$.
- **Game** G_4 **(Attacking** KS **via 'Test' query)**: Consider a 'Test(Π^s_{OD})' query for $KS = KDF(P_{ID_O}, T_{ID})$. If $\mathcal{A}$ has not successfully executed 'Corrupt(OD$_O$)' for U_O prior to this session becoming complete (i.e., the session is "fresh"), then P_{ID_O} is unknown to $\mathcal{A}$. Since T_{ID} is a fresh, unique timestamp for each session, and KDF is a random oracle, KS is computationally indistinguishable from a random string to $\mathcal{A}$. The adversary might try to forge M_1 to elicit a response or try to guess P_{ID_O} from $RPMD_{ID_O}$ (if GCS is corrupted). $RPMD_{ID_O} = H(RP_{ID_O} \| Salt_1)$ and $RP_{ID_O} = D_{KDF(P_{ID_O}, Salt_2)}(C_{RP})$. These layers make guessing P_{ID_O} hard. If $\mathcal{A}$ queries "Test" on a fresh session where it does not know P_{ID_O}, the game simulator directly provides a truly random string. The only way $\mathcal{A}$ distinguishes this game from G_3 is by successfully guessing P_{ID_O} or breaking the KDF (covered in G_1). The probability of guessing P_{ID_O} from available information (without corruption) is P_{guess}. $|Adv_{G_3}(\mathcal{A}) - Adv_{G_4}(\mathcal{A})| \leq P_{guess}$.
- **Game** G_5 **(Final Game)**: The output of the 'Test' query is always a truly random string for fresh instances. Thus, $Pr[\mathcal{A} \text{ wins in } G_5] = 1/2$, so $Adv_{G_5}(\mathcal{A}) = 0$.

Combining these, $Adv^{AKE}_{\text{Bio-2FA-IoD}}(\mathcal{A}) \leq 2\sum(\text{negligible terms}) + 2 \cdot P_{guess}$. If P_{ID_O} is chosen from a sufficiently large space or effectively protected by $K_{\beta O}$ (which itself has high entropy), P_{guess} is negligible. Thus, KS is AKE-secure.

6. Performance Evaluation

This section evaluates the performance of the proposed Bio-2FA-IoD protocol. We analyze its computational cost, communication overhead, and qualitatively discuss energy consumption, comparing these aspects with several contemporary authentication schemes for IoD environments. Data for comparison is drawn from the cited literature, and estimations for Bio-2FA-IoD are based on common cryptographic operation timings.

6.1. Computational Cost Analysis

The computational cost is primarily determined by the number and type of cryptographic operations performed by each entity during the online authentication phase. Registration phase costs are amortized as they are one-time operations. We use the following estimated execution times for common cryptographic operations, which are representative values found in recent IoD security literature (e.g., Nyangaresi et al. [6], Jan et al. [7], and values from Khedr [1] where applicable):

- T_H (SHA-256 hash): ≈ 0.0025 ms (Nyangaresi et al. [6])
- T_{FE} (Fuzzy Extractor - Rec): ≈ 0.0098 ms (Nyangaresi et al. [6])
- T_S (AES-128 Enc/Dec): ≈ 0.0046 ms (Nyangaresi et al. [6])
- T_{KDF} (PBKDF2-SHA256, simplified for derivation): $\approx 2 \times T_H = 0.0050$ ms (estimation, actual PBKDF2 can be higher due to iterations but here it's used for key derivation not password stretching from low-entropy input).

Bio-2FA-IoD Computational Cost (Authentication Phase):

- **Operator's Device (OD)**:

 doi:10.20944/preprints202506.0422.v1

(a) Biometric Key Reconstruction ($Rec(\beta'_O, HD_O)$): $1 \times T_{FE} = 0.0098$ ms
(b) Local Key K_O Regeneration ($KDF(P_{ID_O}, Salt_2)$): $1 \times T_{KDF} = 0.0050$ ms
(c) RP_{ID_O} Decryption ($D_{K_O}(C_{RP})$): $1 \times T_S = 0.0046$ ms
(d) Session Key KS Generation ($KDF(P_{ID_O}, T_{ID})$): $1 \times T_{KDF} = 0.0050$ ms
(e) OTAC Encryption ($E_{KS}(PIN \| T_{ID} \| RP_{ID_O})$): $1 \times T_S = 0.0046$ ms

Total OD Cost: $0.0098 + 0.0050 + 0.0046 + 0.0050 + 0.0046 =$ **0.029 ms**.

- **Ground Control Station (GCS)**:

(a) Session Key KS_{GCS} Generation ($KDF(P_{ID_O}, T_{ID})$): $1 \times T_{KDF} = 0.0050$ ms
(b) OTAC Decryption ($D_{KS_{GCS}}(OTAC)$): $1 \times T_S = 0.0046$ ms
(c) $RPMD'_{ID_O}$ Calculation ($H(RP'_{ID_O} \| Salt_1)$): $1 \times T_H = 0.0025$ ms

Total GCS Cost: $0.0050 + 0.0046 + 0.0025 =$ **0.0121 ms**.

Total Protocol Computational Cost (Online Phase): 0.029 ms (OD) + 0.0121 ms (GCS) = **0.0411 ms**.

Comparison with other schemes: Table 2 presents a comparison. The proposed Bio-2FA-IoD, with a total cost of 0.0411 ms, demonstrates significantly lower computational overhead compared to schemes that rely heavily on asymmetric cryptography such as ECC (e.g., Akram et al. [10] with costs around 0.448 ms) or HECC (e.g., HCALA by Berini et al. [8] at 3.3873 ms). Its cost is comparable to, and slightly higher than, the PUF and biometric-based scheme by Nyangaresi et al. [6] (0.0388 ms), which involves PUF operations directly in the authentication flow for multiple entities. In contrast, Jan et al.'s scheme [7], while using symmetric primitives, involves more operations leading to a higher cost (17.79 ms). The EPUF-based authentication by Najafi et al. [11] also reports very low computational costs for its core logic (excluding PUF read time), comparable to our proposal. The efficiency of Bio-2FA-IoD stems from its reliance on a minimal set of fast symmetric operations (AES) and hashing (SHA-256) for the core online authentication loop, with the biometric processing localized to the OD.

Table 2. Comparative Computational Costs (ms) of Authentication Phase

Protocol	Total Est. Cost (ms)	Primary Cryptographic Primitives
Bio-2FA-IoD (Proposed)	**0.0411**	FE, KDF, AES, SHA-256
Nyangaresi et al. (2024) [6]	0.0388	PUF, FE, Hash
Jan et al. (2021) [7]	17.7939	HMAC-SHA1, Hash, XOR
Berini et al. (HCALA, 2023) [8]	3.3873	HECC, Hash, AES
Khalid et al. (HOOPOE, 2023) [9]	8.343	AES, RSA, Hash, Signature
Akram et al. (2023) [10]	0.44819	ECC, Hash, Signature
Najafi et al. (EPUF-Auth, 2025) [11]	≈0.016	PUF (DRAM), Hash, XOR

6.2. Communication Cost Analysis

Communication cost is determined by the number of messages exchanged during the authentication phase and their total size in bits.

- Message $M_1 = (ID_O, OTAC, T_{ID})$
- Message $M_2 = (PIN')$

Assuming standard sizes for parameters (output of SHA-256 is 256 bits, AES-128 block size is 128 bits):

- ID_O: e.g., 128 bits (16 bytes)
- T_{ID} (Timestamp): e.g., 32 bits (4 bytes)
- PIN (Nonce): e.g., 128 bits (16 bytes)
- RP_{ID_O} (Random Password, assumed to be key-like): e.g., 160 bits (20 bytes)
- Plaintext for OTAC: $PIN(128) + T_{ID}(32) + RP_{ID_O}(160) = 320$ bits (40 bytes).
- OTAC (Ciphertext using AES-128, e.g., CBC mode with IV and padding): Size will be multiple of 128 bits. For 40 bytes plaintext, it might be 3 blocks = $3 \times 128 = 384$ bits (48 bytes) including IV or padding.
- Size of M_1: $128(\text{for } ID_O) + 384(\text{for } OTAC) + 32(\text{for } T_{ID}) =$ **544 bits**.

 doi:10.20944/preprints202506.0422.v1

- Size of M_2: $128 (\text{for } PIN') = \mathbf{128\ bits}$.

Total Communication Cost: $544 + 128 = \mathbf{672\ bits}$ (or 84 bytes).

Comparison with other schemes: Table 3 compares the communication overhead. The Bio-2FA-IoD protocol's total communication cost of 672 bits exchanged in two messages is exceptionally low. It is significantly more efficient than many other IoD protocols, such as Nyangaresi et al. [6] (2464 bits, 6 messages), Jan et al. [7] (3720 bits, 4 messages), and HCALA by Berini et al. [8] (1536 bits, 3 messages). The EPUF-based protocol by Najafi et al. [11] also boasts low communication costs (around 192 Bytes or 1536 bits, but their message structure is different). This efficiency is critical for IoD networks where bandwidth may be constrained, communication links may be unreliable, or energy consumed by radio transmission is a significant concern.

Table 3. Comparative Communication Costs of Authentication Phase

Protocol	Messages	Total Est. Cost (bits)	Source for Comparison
Bio-2FA-IoD (Proposed)	**2**	**672**	Current Estimation
Nyangaresi et al. (2024) [6]	6	2464	Nyangaresi et al.
Jan et al. (2021) [7]	4	3720	Jan et al.
Berini et al. (HCALA, 2023) [8]	3	1536	Berini et al.
Khalid et al. (HOOPOE, 2023) [9]	2 (D-GSS)	1280	Khalid et al.
Akram et al. (2023) [10]	2	1152	Akram et al.
Najafi et al. (EPUF-Auth, 2025) [11]	2	1536	Najafi et al. (192 Bytes)

6.3. Energy Consumption Analysis

Energy consumption in IoD devices (drones and ODs) is a critical performance metric, directly influenced by computational complexity and communication load (data transmission and reception). Lightweight cryptographic operations and fewer/smaller messages inherently lead to lower energy usage. While precise energy figures require hardware-specific measurements, we can make qualitative comparisons. Given its extremely low total computational cost (0.0411 ms) and minimal communication overhead (672 bits), Bio-2FA-IoD is expected to be very energy-efficient. The computational energy on the OD (0.029 ms) would be in the microjoule range, even with conservative power estimates for mobile CPUs during short cryptographic bursts. Transmission energy for a mere 84 bytes is significantly less than for protocols transferring several kilobytes. For instance, schemes like HCALA [8], using HECC, report an energy consumption for communication around 3.38×10^{-4} J (338 μJ) for 3 messages, indicating a higher baseline due to larger cryptographic payloads typical of public-key schemes. The proposed protocol's reliance on fast symmetric cryptography and hashing for the online phase, with biometric processing localized to the OD, significantly reduces the energy drain, making it well-suited for battery-dependent IoD components. The HOOPOE protocol [9] also provides detailed energy analysis, showing that RSA/AES operations are more energy-intensive than the symmetric primitives used in Bio-2FA-IoD.

6.4. Security and Functionality Features Comparison

Table 4 provides a comparative overview of the security and functionality features of the proposed Bio-2FA-IoD protocol against selected schemes, based on the analysis in Section 5 and common features discussed in IoD literature [6–9,27].

 doi:10.20944/preprints202506.0422.v1

Table 4. Comparative Security and Functionality Features

Feature	Khedr [1] (Adapted)	Nyangaresi [6]	Jan [7]	HCALA [8]	HOOPOE [9]	Bio-2FA-IoD (Proposed)
Mutual Authentication	Yes	Yes	Yes	Yes	Yes	**Yes**
Keylogging Resist. (GCS)	Yes	Yes (Implied)	N/A	N/A	N/A	**Yes**
Shoulder-Surf. Resist.	Yes	Yes (Biometric)	N/A	Yes (PIN on User)	N/A	**Yes**
User Anonymity	No	Yes	Partial	Yes	Yes	**Partial**
Replay Attack Resist.	Yes	Yes	Yes	Yes	Yes	**Yes**
Impersonation Resist.	Yes	Yes	Yes	Yes	Yes	**Yes**
Drone Capture Resilience	N/A	Yes	Partial	Partial	Yes	**Good**
Forward Secrecy (PFS)	Limited	Yes	No	Yes	Yes	**Limited**
Formal Verification	AVISPA (orig.)	RoR, AVISPA	ROM, ProVerif	ROM, AVISPA	RoR, ProVerif	**BAN, BPR**
Lightweight Nature	Yes	Yes	Yes	Moderate	Moderate	**Yes**
Biometric Integration	No (Visual 2FA)	Yes	No	No	No	**Yes**
Three-Factor Auth. Potential	No (2FA)	Yes (Implicit)	No (2FA)	Yes (Implicit)	No	**Yes (Bio, OD, PIN)**
Handles Handover	No	No	No	No	Yes	**No**
Blockchain Integration	No	No	No	Yes	No	**No**
PUF Usage	No	Yes	No	No	No	**No**

The proposed Bio-2FA-IoD effectively provides strong mutual authentication and resists keylogging and shoulder-surfing attacks by adapting Khedr's logic [1] with biometrics. Its lightweight nature is a key advantage for IoD. While user anonymity and perfect forward secrecy are areas for potential future enhancement (currently marked "Partial" and "Limited"), its resilience to drone capture is good due to secrets being primarily on the OD. The provision for dual formal verification (BAN logic and BPR model) adds confidence in its security design. Compared to protocols like HCALA [8] or HOOPOE [9] that offer broader features like blockchain integration or handover at higher complexity, Bio-2FA-IoD focuses on a streamlined, secure, and efficient initial operator-to-GCS authentication suitable for many IoD scenarios.

7. Conclusion

This paper proposed Bio-2FA-IoD, a biometric-enhanced two-factor authentication protocol specifically designed for the Internet of Drones environment. By adapting robust security principles from visual authentication methods and replacing impractical QR-code mechanisms with operator-initiated biometric verification on a personal device, the protocol effectively addresses threats like keylogging and shoulder-surfing while being practical for IoD deployment. The integration of fuzzy extractors ensures reliable biometric key generation, which is crucial for the usability of biometric systems.

A comprehensive security analysis was conducted. The informal analysis demonstrated the protocol's resilience against a range of pertinent attacks, including replay attacks, impersonation attempts, and offered good resilience against drone capture by localizing critical operator secrets to the Operator's Device. Formal verification using BAN logic provided evidence for the correct establishment of mutual authentication and shared beliefs regarding the session key and exchanged parameters, under its idealized assumptions. Furthermore, a security proof sketch within the rigorous BPR model suggests that the derived session key for OTAC protection achieves Authenticated Key Exchange (AKE) security against active adversaries, contingent on the strength of the underlying cryptographic primitives (SHA-256, AES-128, PBKDF2) and the security of the biometric system incorporating fuzzy extractors.

The performance evaluation underscored the lightweight nature of Bio-2FA-IoD. Its reliance on efficient symmetric key cryptography, hash functions, and KDFs for the online authentication phase results in minimal computational costs for both the OD and GCS, and very low communication overhead. This makes it particularly suitable for resource-constrained IoD components, such as operator handheld devices and potentially the drones themselves if their role were to expand beyond relaying.

 doi:10.20944/preprints202506.0422.v1

References

1. Khedr, W. I. (2018). Improved keylogging and shoulder-surfing resistant visual two-factor authentication protocol. *Journal of Information Security and Applications*, 39, 41-57.
2. Dodis, Y., Reyzin, L., & Smith, A. (2004). Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. In *EUROCRYPT 2004* (LNCS 3027, pp. 523-540). Springer.
3. Burrows, M., Abadi, M., & Needham, R. (1989). A Logic of Authentication. *ACM Transactions on Computer Systems (TOCS)*, 8(1), 18-36.
4. Bellare, M., Pointcheval, D., & Rogaway, P. (2000). Authenticated Key Exchange Secure Against Dictionary Attacks. In *Eurocrypt 2000* (LNCS 1807, pp. 139-155). Springer.
5. Dolev, D., & Yao, A. (1983). On the security of public key protocols. *IEEE Transactions on Information Theory*, 29(2), 198-208.
6. Nyangaresi, V. O., Al-Joboury, I. M., Al-sharhanee, K. A., Najim, A. H., Abbas, A. H., & Hariz, H. M. (2024). A biometric and physically unclonable function-Based authentication protocol for payload exchanges in internet of drones. *e-Prime - Advances in Electrical Engineering, Electronics and Energy*, 7, 100471.
7. Jan, S. U., Qayum, F., & Khan, H. U. (2021). Design and Analysis of Lightweight Authentication Protocol for Securing IoD. *IEEE Access*, 9, 69287-69306.
8. Berini, A. D. E., Ferrag, M. A., Farou, B., & Seridi, H. (2023). HCALA: Hyperelliptic curve-based anonymous lightweight authentication scheme for Internet of Drones. *Pervasive and Mobile Computing*, 92, 101798.
9. Khalid, H., Hashim, S. J., Hashim, F., Ahamed, S. M. S., Chaudhary, M. A., Altarturi, H. H. M., & Saadoon, M. (2023). HOOPOE: High Performance and Efficient Anonymous Handover Authentication Protocol for Flying Out of Zone UAVs. *IEEE Transactions on Vehicular Technology*, 72(8), 10906-10920.
10. Akram, M. A., Ahmad, H., Mian, A. N., Jurcut, A. D., & Kumari, S. (2023). Blockchain-based privacy-preserving authentication protocol for UAV networks. *Computer Networks*, 224, 109638.
11. Najafi, F., Kaveh, M., Mosavi, M. R., Brighente, A., & Conti, M. (2025). EPUF: An Entropy-Derived Latency-Based DRAM Physical Unclonable Function for Lightweight Authentication in Internet of Things. *IEEE Transactions on Mobile Computing*, 24(3), 2422-2436.
12. Jan, S. U., & Khan, H. U. (2021). Identity and Aggregate Signature-Based Authentication Protocol for IoD Deployment Military Drone. *IEEE Access*, 9, 126038-126050.
13. Hussain, S., Farooq, M., Alzahrani, B. A., Albeshri, A., Alsubhi, K., & Chaudhry, S. A. (2023). An Efficient and Reliable User Access Protocol for Internet of Drones. *IEEE Access*, 11, 59689-59700.
14. Hasson, M., Yassin, A. A., Yassin, A. J., Rashid, A. M., Yaseen, A. A., & Alasadi, H. (2021). Password authentication scheme based on smart card and QR code. *Indonesian Journal of Electrical Engineering and Computer Science*, 23(1), 140-149.
15. Khan, A. A., Kumar, V., & Ahmad, M. (2022). An elliptic curve cryptography based mutual authentication scheme for smart grid communications using biometric approach. *Journal of King Saud University - Computer and Information Sciences*, 34(1), 698-705.
16. Akram, J., Akram, A., Jhaveri, R. H., Alazab, M., & Chi, H. (2022). BC-IoDT: Blockchain-based Framework for Authentication in Internet of Drone Things. *Proceedings of the ACM MobiCom Workshop on Drone Assisted Wireless Communications for 5G and Beyond (DroneCom '22)*, ACM.
17. Vangala, A., Agrawal, S., Das, A. K., Pal, S., Kumar, N., Lorenz, P., & Park, Y. (2024). Big Data-Enabled Authentication Framework for Offshore Maritime Communication Using Drones. *IEEE Transactions on Vehicular Technology*, 73(7), 10196-10210.
18. Zhang, S., Liu, Y., Han, Z., & Yang, Z. (2023). A Lightweight Authentication Protocol for UAVs Based on ECC Scheme. *Drones*, 7(5), 315.
19. Wu, T., Guo, X., Chen, Y., Kumari, S., & Chen, C. (2022). Amassing the Security: An Enhanced Authentication Protocol for Drone Communications over 5G Networks. *Drones*, 6(1), 10.
20. Zhang, N., Jiang, Q., Li, L., Ma, X., & Ma, J. (2021). An efficient three-factor remote user authentication protocol based on BPV-FourQ for internet of drones. *Peer-to-Peer Networking and Applications*, 14, 3319-3332.
21. Wu, T. Y., Wu, H., Kumari, S., & Chen, C. M. (2025). An enhanced three-factor based authentication and key agreement protocol using PUF in IoMT. *Peer-to-Peer Networking and Applications*, 18, 83. (Note: Year is 2025, might be early access).
22. El-Zawawy, M. A., Brighente, A., & Conti, M. (2023). Authenticating Drone-Assisted Internet of Vehicles Using Elliptic Curve Cryptography and Blockchain. *IEEE Transactions on Network and Service Management*, 20(2), 1775-1789.

 doi:10.20944/preprints202506.0422.v1

23. Cabuk, U. C., Dalkilic, G., & Dagdeviren, O. (2021). CoMAD: Context-Aware Mutual Authentication Protocol for Drone Networks. *IEEE Access*, 9, 78400-78414.
24. Gupta, A., Barthwal, A., Vardhan, H., Kakria, S., Kumar, S., & Parihar, A. S. (2023). Evolutionary study of distributed authentication protocols and its integration to UAV-assisted FANET. *Multimedia Tools and Applications*, 82, 42311-42330.
25. Nikooghadam, M., Amintoosi, H., Islam, S. H., & Moghadam, M. F. (2021). A provably secure and lightweight authentication scheme for Internet of Drones for smart city surveillance. *Journal of Systems Architecture*, 115, 101955.
26. Jafarian, I. (2023). Cryptanalysis of Nikooghadam et al.'s Lightweight Authentication Protocol for Internet of Drones. *arXiv preprint arXiv:2311.02512*. https://arxiv.org/abs/2311.02512
27. Fatima, S., Akram, M. A., Mian, A. N., Kumari, S., & Chen, C. M. (2024). On the Security of a Blockchain and PUF-Based Lightweight Authentication Protocol for Wireless Medical Sensor Networks. *Wireless Personal Communications*, 136, 1079-1106.

Short Biography of Authors

Hyunseok Kim Hyunseok Kimreceived the B.S. degree in the Department of Business Management from Korea Military Academy, Seoul, Korea in 2000, M.S. and Ph.D in the Department of Computer Science and Engineering from Korea University, Seoul, Korea in 2006 and 2009, respectively. He is currently an associate professor at the ICT Polytech Institute of Korea. His research interests include the areas of Formal Methods(Formal Specification, Formal Verification, Model Checking), IoD Authentication Design, Smart Card Privacy, M-Commerce Secure Transaction.