mathematics

MDPI

Article

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

Hyunseok Kim [1] and Seunghyun Park [2,*]

[1] Department of Information and Security, ICT Polytech Institute of Korea, 16-26 Sunamro, Gwangjusi 12792, Republic of Korea; hskim@ict.ac.kr
[2] Division of Computer Engineering, Hansung University, Seoul 02876, Republic of Korea
* Correspondence: sp@hansung.ac.kr

Abstract

The Internet of Drones (IoD) is rapidly expanding into sensitive applications, necessitating robust and efficient authentication. Traditional methods struggle against prevalent attacks, especially considering the unique vulnerabilities of the IoD, such as drone physical capture. This paper proposes Bio-2FA-IoD, a novel biometric-enhanced two-factor authentication protocol designed for secure IoD operations. Drawing on established 2FA principles and fuzzy extractor technology, Bio-2FA-IoD achieves strong mutual authentication between an operator (via an operator device), a drone (as a relay), and a ground control station (GCS), supported by a trusted authority. We detail the protocol's registration and authentication phases, emphasizing reliable biometric key generation. A formal security analysis using BAN logic demonstrates secure belief establishment and key agreement, while a proof sketch under the Bellare–Pointcheval–Rogaway (BPR) model confirms its security against active adversaries in Authenticated Key Exchange (AKE) contexts. Furthermore, a comprehensive performance evaluation conducted using the Contiki OS and Cooja simulator illustrates Bio-2FA-IoD's superior efficiency in computational and communication costs, alongside very low latency, high packet delivery rate, and minimal energy consumption. This positions it as a highly viable and lightweight solution for resource-constrained IoD environments. Additionally, this paper conceptually explores potential extensions to Bio-2FA-IoD, including the integration of Diffie–Hellman for enhanced perfect forward secrecy and a Sybil-free pseudonym management scheme for improved user anonymity and unlinkability.

Keywords: Internet of Drones (IoD); UAV security; biometric authentication; two-factor authentication (2FA); fuzzy extractor; BAN logic; BPR model; lightweight authentication; secure communication; Contiki OS; Cooja simulator; Diffie–Hellman; perfect forward secrecy; pseudonymity; Sybil-free

MSC: 68M25

Academic Editor: Antanas Cenys

Received: 4 June 2025
Revised: 26 June 2025
Accepted: 30 June 2025
Published: 3 July 2025

Citation: Kim, H.; Park, S. Bio-2FA-IoD: A Biometric-Enhanced Two-Factor Authentication Protocol for Secure Internet of Drones Operations. *Mathematics* **2025**, *13*, 2177. https://doi.org/10.3390/math13132177

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 [1–4]. This expansion, however, brings substantial security challenges, particularly for authentication [1]. 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 [5,6]. 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, a vulnerability distinct from typical IoT devices [5–7]. 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 [1]. 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 [8]. 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 [9,10]. Recent research in IoD and related fields has 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 [5,9,11]. While existing protocols address some IoD security aspects, they often fall short in combining strong operator authentication (resistant to observational attacks) with lightweight design, or rely on mechanisms impractical for dynamic drone environments.

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) [8]. 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 [1,11]. The contributions of this paper include the following:

1. **Novel biometric-enhanced 2FA protocol:** The design of Bio-2FA-IoD, a novel two-factor authentication protocol for IoD environments that integrates biometrics (via fuzzy extractors) as a primary operator verification factor, managed through a trusted operator's device, to provide strong, user-bound authentication resilient to observational attacks [5,12,13]. This includes detailed descriptions of the registration and authentication phases, incorporating fuzzy extractors for robust and reliable biometric key generation.
2. **Dual formal security verification:** A rigorous formal security analysis using both Burrows–Abadi–Needham (BAN) logic [14] to verify mutual authentication and shared belief establishment, and the Bellare–Pointcheval–Rogaway (BPR) model [15] for Authenticated Key Exchange (AKE) to prove resilience against a wide range of active adversarial attacks.
3. **Comprehensive performance evaluation:** A detailed performance evaluation, comparing Bio-2FA-IoD with several contemporary IoD authentication protocols in terms of computational costs, communication overhead, and qualitative energy efficiency. Furthermore, a simulation-based performance analysis conducted using the Contiki OS and Cooja simulator [16,17] illustrates Bio-2FA-IoD's superior efficiency in

computational and communication costs, alongside very low latency, high packet delivery rate, and minimal energy consumption. This positions it as a highly viable and lightweight solution for resource-constrained IoD environments.

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, including a detailed simulation analysis. Section 7 explores conceptual extensions to the Bio-2FA-IoD protocol for enhanced perfect forward secrecy and user anonymity. Finally, Section 8 concludes the paper and outlines potential directions for future research, followed by the list of references in the References section.

2. Related Work

The quest for secure and efficient authentication in the Internet of Drones (IoD) has spurred significant research, moving beyond traditional password-based systems. While Khedr's work on visual two-factor authentication [8] effectively tackled keylogging and shoulder surfing using QR codes and smartphones in general computing environments, its direct applicability to the 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. [5] proposed 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. [9] 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 [9,10,12,13]. Lightweight design is paramount in the IoD. Jan et al. [1] developed a protocol based on HMAC-SHA1, 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 the general IoT, which has similar constraints to the IoD.

Several other protocols address specific IoD challenges. Khalid et al. [6] 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. [2] 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 [13] and for general IoDT authentication frameworks [18]. 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 [19] provided a cryptanalysis of Nikooghadam et al.'s lightweight IoD authentication scheme [20],

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 the IoD by Zhang et al. [21] and various ECC-based schemes [7,22]. Cabuk et al. [23] proposed CoMAD, a context-aware mutual authentication protocol for drone networks.

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 [8]. 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. The performance evaluation of wireless network protocols, especially for Low Power and Lossy Networks (LLNs) like the IoD, often relies on simulation tools that can accurately model the operating system constraints and radio medium characteristics. The Contiki OS is a widely used lightweight operating system for networked embedded systems, and Cooja is a flexible Java-based simulator designed specifically for Contiki-powered sensor networks [16,17]. Tall et al. [16] demonstrated the importance of a compliant IEEE 802.15.4 CSMA/CA implementation on the Contiki OS to obtain realistic WSN performance results. Ali et al. [17] performed extensive simulations of RPL (Routing Protocol for Low Power and Lossy Networks) in Contiki using Cooja, evaluating its performance in terms of energy, latency, packet delivery ratio, control overhead, and convergence time, while also optimizing various parameters. These studies highlight the effectiveness of Contiki/Cooja for realistic performance evaluation in resource-constrained wireless networks, providing a solid foundation and methodology for our own simulation analysis. To provide a clearer comparison of related works, Table 1 summarizes their key features and characteristics.

Table 1. Comparative analysis of related works.

Protocol	Primary Mechanism	Key Security Goals Achieved	Formal Analysis Method	IoD-Specific	Biometric/PUF
Khedr [8]	Visual 2FA	Keylogging and shoulder-surfing resistant	AVISPA	No	No
Nyangaresi et al. [5]	Biometric, PUF	Drone capture resilience, RoR security	RoR, AVISPA	Yes	Yes
Jan et al. [1]	HMAC-SHA1	Lightweight, replay attack resistance	ROM, ProVerif	Yes	No
Berini et al. (HCALA) [2]	HECC, blockchain	Anonymity, decentralization	ROM, AVISPA	Yes	No
Khalid et al. (HOOPOE) [6]	AES-RSA, handover	Anonymous handover, performance	RoR, ProVerif	Yes	No
Akram et al. [13]	ECC, blockchain	Privacy-preserving, decentralization	None (informal)	Yes	No
Najafi et al. (EPUF-Auth, 2025) [11]	DRAM PUF, lightweight IoT	Lightweight authentication (general IoT)	None (informal)	Partial	Yes
Bio-2FA-IoD (proposed)	**Biometric, 2FA, symmetric**	**Keylogging, shoulder-surfing, and drone capture resilience; lightweight**	**BAN, BPR**	**Yes**	**Yes**

Note: Bold indicates the proposed protocol for emphasis.

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 [3,6,7,24]. 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.
- **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 [8].
- **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 [2]. Drone technology integrates various components like communication modules (e.g., for FANETs [1,25]), 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 [8] and the "Control Server (CS)" or "Ground Station Server (GSS)" in other IoD studies [2,7,13]. 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 [2,24]. It does not participate in every authentication session but establishes the root of trust.

Communication pathways are as follows: The U_O interacts directly with the OD for biometric input. The OD communicates with the 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 is a crucial step in 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 [7], 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, serving as an initial roadmap for the protocol's architecture and inter-entity relationships.

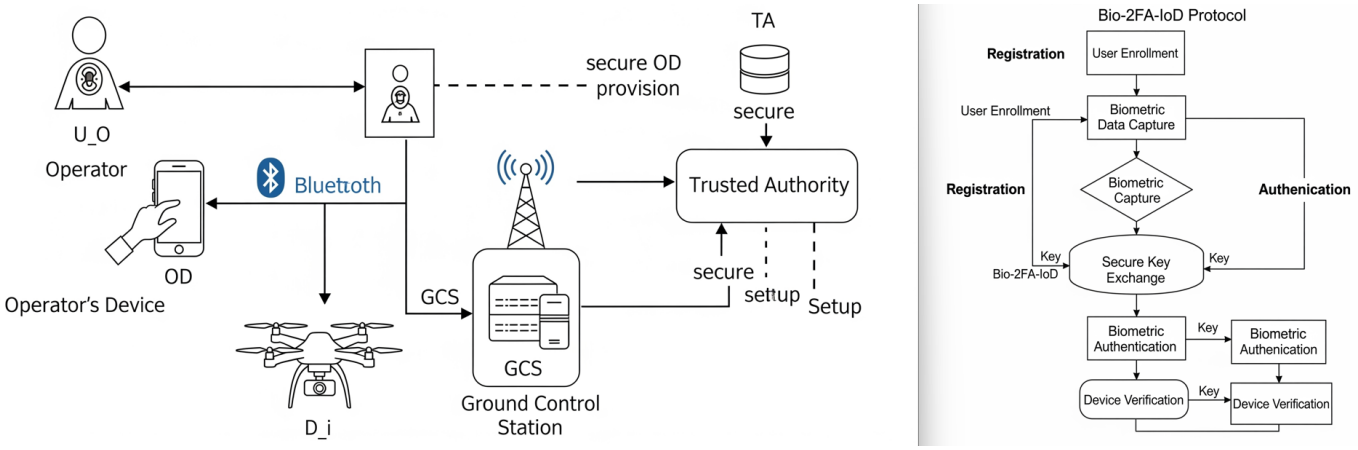

Figure 1. Bio-2FA-IoD—new process framework diagram: System and network model. This figure illustrates the key entities: operator (U_O), operator's device (OD), drone (D_i), ground control station (GCS), and trusted authority (TA). It depicts the primary communication links and overall process flow, serving as an initial roadmap for the protocol's architecture and inter-entity relationships.

3.2. Threat Model

We adopt a comprehensive adversary model based on the Dolev–Yao (DY) model [26], extended with considerations relevant to the IoD and biometric systems, similar to threat models discussed in [2,5,24,27]. 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 [5–7]. 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) [8]. The use of biometrics as the primary factor mitigates this, and mechanisms like Khedr's SVOSK [8] 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 [19].

- **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 the existing IoD security literature [1–7,11]:

- **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 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) [8].
- **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 [1].
- **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 [2].
- **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 [5,6].
- **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 [2,6].

- **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 [22], 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 [1,8], stolen verifier attacks [19], 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 [1,11,22].
- **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.
- **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 [6] 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 [5,8]. For a hash function to be cryptographically secure, as detailed in [1,2,5], it must satisfy the following:

- 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 [8]. 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 [8]. The chosen algorithm (e.g., AES-128—Advanced Encryption Standard with 128-bit key, specified in FIPS PUB 197 [8]) should be resistant to known cryptanalytic attacks and provide semantic security (IND-CPA: Indistinguishability under Chosen-Plaintext Attack). Our protocol uses AES-128 in a mode like CBC (cipher block chaining) for OTAC encryption.

- 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.

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 [8]. Our protocol uses a KDF based on a secure hash function (e.g., PBKDF2 as defined in RFC 2898 [8], using HMAC-SHA-256 as the underlying pseudorandom function) for deriving keys like K_O and KS.

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

KDFs are designed to be computationally intensive to slow down brute-force attacks on the input master secret (especially if it is a low-entropy password). They should produce outputs that are pseudorandom and indistinguishable from truly random keys.

3.5. Biometrics and Fuzzy Extractor

Biometric authentication leverages unique physiological (e.g., fingerprint [28], iris) or behavioral (e.g., voice, signature) characteristics of an individual for identity verification [5,9]. 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 the false acceptance rate (FAR) and false rejection rate (FRR). A **fuzzy extractor**, introduced by Dodis et al. [12], is a cryptographic tool designed to address this issue by reliably extracting a stable, uniformly random cryptographic key from noisy biometric inputs. Many IoD and related security protocols acknowledge its utility [5,9,10,13]. 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 [14] 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 [24,28], include the following:

- **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 [15], and its variants like the Real-or-Random (RoR) model, employed in several contemporary IoD security papers [1,2,5,6,13,19], provides 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 the following:

- **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 [26]). Its capabilities are modeled through a set of allowed oracle queries. Based on similar models in [5] and the original BPR model, common queries include the following:
  - '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 $Adv_{\text{Prot}}^{AKE}(\mathcal{A}) = |2 \cdot \Pr[\mathcal{A} \text{ correctly guesses that } b] - 1|$ is negligible. Proofs usually proceed by game hopping.

3.7. Notation

Table 2 summarizes the key notation used throughout this paper.

Table 2. Notation 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 in CBC mode)
$D_K(\cdot)$	Symmetric decryption with key K (e.g., AES-128 in CBC mode)
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 [8]. It correctly executes its role during the registration phase, and its long-term secrets remains confidential.
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.
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 [8]). While this assumes a high level of device integrity, practical deployments can further reinforce OD security by integrating layered defense mechanisms such as trusted execution environments (TEEs) or secure boot validation to mitigate risks from sophisticated malware or side-channel attacks.
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 inherently necessary to correct minor variations in β' [5,12].
- 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 in CBC mode) 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 [8].
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 [8] 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: $(K_{\beta O}, HD_O) = Gen(\beta_O)$ [12].
   - 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 [8].
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 $Salt_2$. It is important to note that the raw biometric template (β_O) is not stored; only the helper data (HD_O) is maintained on the OD, aligned with privacy-preserving practices.

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

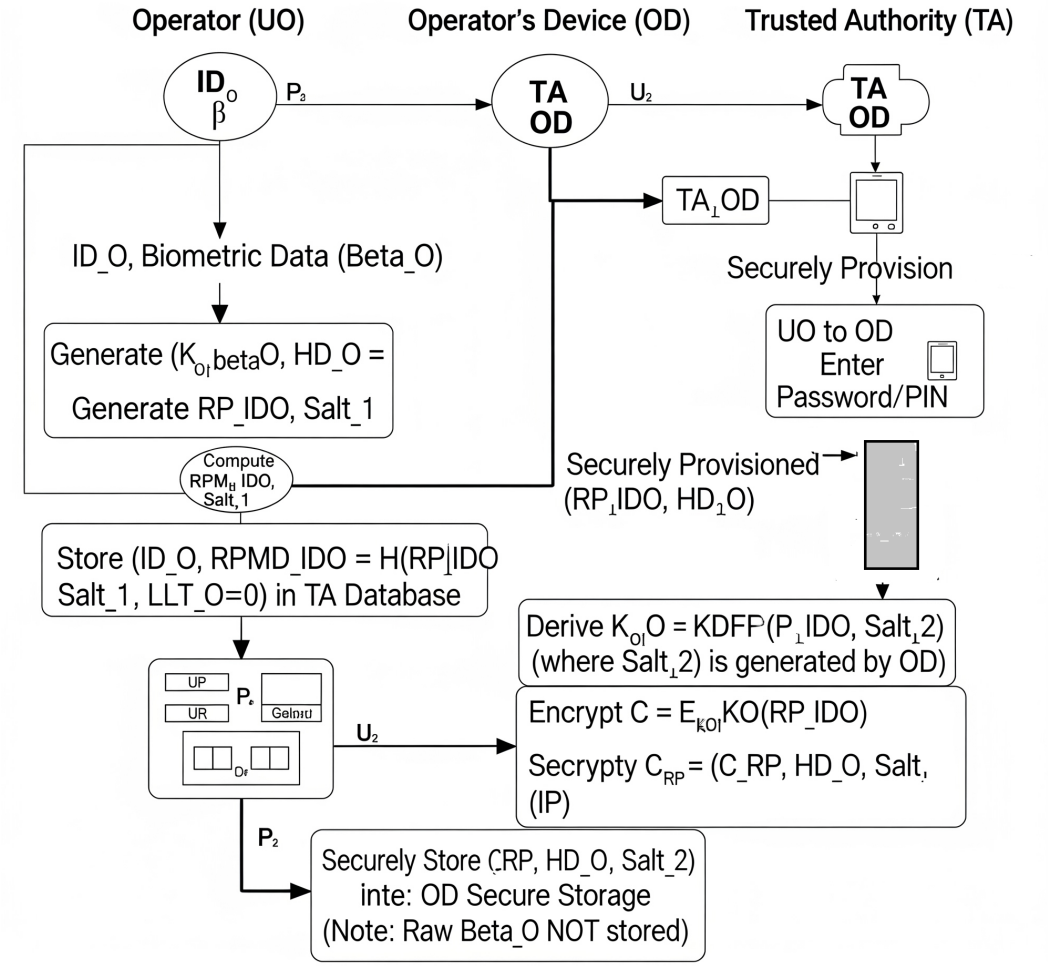

Figure 2. Bio-2FA-IoD—registration phase. This diagram illustrates the physical interpretation of interactions between the operator (U_O), operator's device (OD), and trusted authority (TA) for initial setup and credential provisioning. It depicts the flow of biometric enrollment, key generation, and secure provisioning of secrets to the OD.

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.

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: $K'_{\beta O} = Rec(\beta'_O, HD_O)$ [12]. 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:
   - 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.

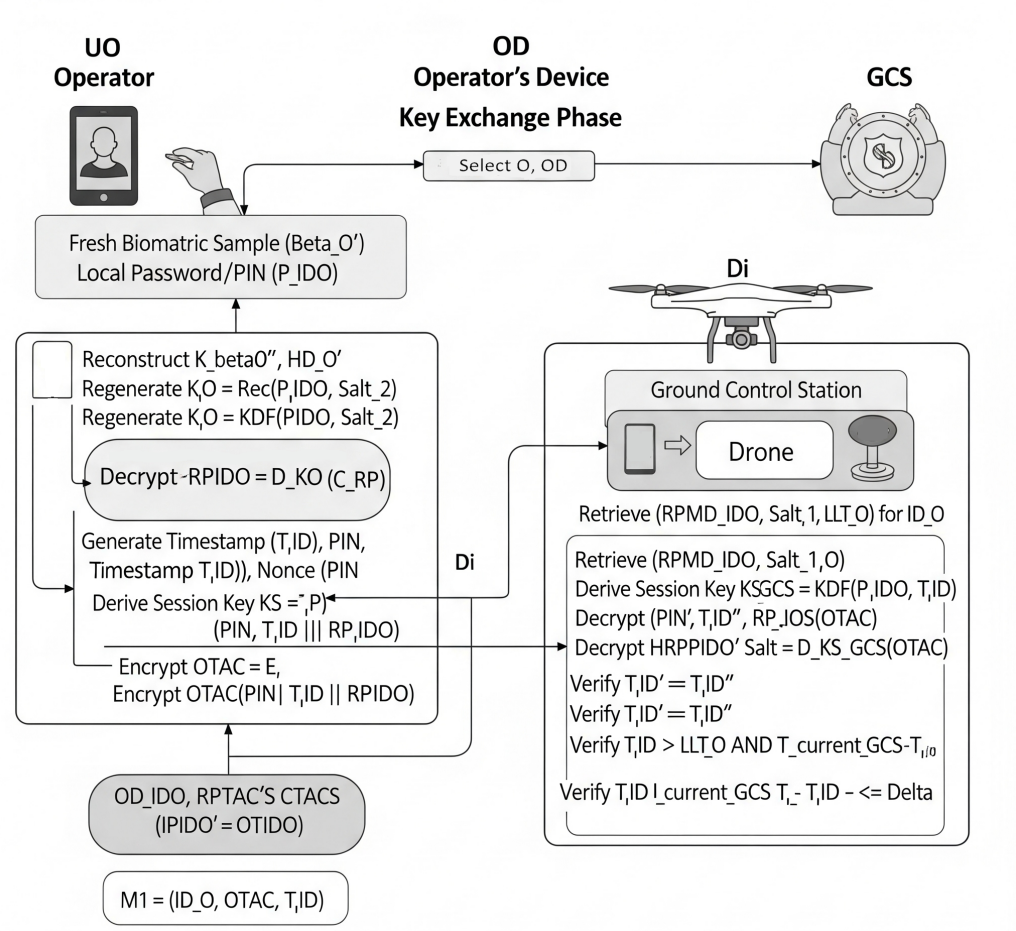

Figure 3. Bio-2FA-IoD—authentication and key exchange phase. This diagram illustrates the physical interpretation of interactions between the operator (U_O), OD, drone (D_i), and ground control station (GCS) for mutual authentication. It visually represents the sequence of biometric input, credential activation, OTAC generation, message exchange, and verification steps across the physical entities.

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 (on behalf of U_O) authenticates to the GCS by demonstrating knowledge of P_{ID_O} and RP_{ID_O} (via the encrypted 'OTAC'). The GCS authenticates to the OD by correctly returning the 'PIN'' from the decrypted 'OTAC', confirming it possesses the correct session key KS. This effectively prevents impersonation by either party and robustly meets requirement 1. This mechanism is similar to Khedr [8] but specifically adapted for biometric integration.
- **Keylogging** and shoulder-surfing resistance: Primary authentication relies on the operator's biometric input. Any secondary PIN (P_{ID_O}) entry on the OD can be secured using an SVOSK-like interface [8], which effectively mitigates direct observation or software keylogging. This significantly enhances resistance compared to traditional password entry and clearly meets requirement 2.
- **Replay** attack resistance: The protocol incorporates a fresh timestamp (T_{ID}) within the encrypted 'OTAC', and its verification against the GCS's current time ($T_{current\_GCS}$) and the operator's last login time (LLT_O) effectively prevents replay attacks. The GCS rigorously ensures that T_{ID} is fresh ($T_{ID} > LLT_O$) and falls within an acceptable time window (Δ_t). This robustly meets requirement 3.
- **Impersonation** attack resistance:
  - *Operator/OD impersonation*: An adversary attempting to impersonate the operator or OD would need to correctly generate the 'OTAC'. This necessitates knowledge of the biometric key $K_{\beta O}$ (to decrypt C_{RP} and obtain RP_{ID_O}), the local password P_{ID_O} (to derive K_O and KS), and the correct current timestamp T_{ID}. Without these crucial secrets, forming a valid 'OTAC' and deriving the correct PIN is computationally infeasible.
  - *GCS* impersonation: An adversary attempting to impersonate the GCS would be required to return the correct 'PIN'' in message M_2. This 'PIN'' is the same 'PIN' generated by the OD and encrypted within the 'OTAC' using the session key KS. An adversary cannot derive PIN' without first deriving KS, which in turn requires knowledge of P_{ID_O} and the fresh T_{ID}. This makes GCS impersonation impossible for an adversary lacking the operator's secrets.

  This dual protection fulfills requirement 4, consistent with the strong impersonation resistance goals of IoD protocols like HCALA [2] and Jan et al. [1].
- **Drone** capture resilience: A critical and distinct advantage of Bio-2FA-IoD is its robust resilience to drone capture. The sensitive long-term secrets, specifically the operator's local password P_{ID_O}, the encrypted random password C_{RP}, and the biometric helper data HD_O, are securely stored exclusively on the operator's device (OD), not on the drone D_i. Consequently, even if a drone is physically captured and its stored data extracted, it will not compromise the operator's long-term credentials or the security of other operators or drones. This critically meets requirement 5, a vital aspect extensively discussed in [5].
- **Session** key (KS) security: The session key $KS = KDF(P_{ID_O}, T_{ID})$ is derived using the operator's local password P_{ID_O} (which is protected by biometric authentication on the OD) and a fresh, unique timestamp T_{ID} for each session. This ensures that KS is

unique in each session and computationally difficult for an adversary to predict or derive without knowledge of P_{ID_O}. This unequivocally meets requirement 6.

- **Confidentiality** of OTAC payload: The sensitive components of the 'OTAC' payload (PIN, T_{ID}, RP_{ID_O}) are securely encrypted using the session key KS. This ensures that eavesdroppers cannot directly read these values from the intercepted 'M1' message, thereby rigorously maintaining their confidentiality. This directly meets requirement 7.
- **Integrity** of authentication messages: Message integrity is implicitly and effectively provided through the successful decryption of the 'OTAC' and subsequent verification of its contents. If an adversary tampers with 'OTAC' in transit, the GCS's decryption will highly likely fail or yield incorrect values for PIN', T'_{ID}, or RP'_{ID_O}. The verification steps ($H(RP'_{ID_O} \| Salt_1) == RPMD_{ID_O}$ and $T'_{ID} == T_{ID}$) serve as strong integrity checks. Any modification leads to immediate verification failure, ensuring that altered messages are reliably detected. This robustly meets requirement 8.
- **Anonymity** and untraceability: The operator's identity ID_O is explicitly sent in message M_1. While this design choice enables the GCS to identify the operator, it inherently means that an adversary can potentially link communication sessions to a specific operator ID. Therefore, full user anonymity and untraceability (requirement 9) are only partially met in the current design. Future work could effectively explore the integration of pseudonyms, as discussed in [2,6,29] to significantly enhance this feature.
- **Perfect** forward secrecy (PFS): The session key KS depends on the operator's long-term local password P_{ID_O}. If P_{ID_O} were to be compromised (e.g., if an attacker somehow extracted it from a stolen OD and brute-forced it offline), past session keys (KS) derived using that P_{ID_O} could potentially be compromised. Therefore, perfect forward secrecy (requirement 10) is limited in the current design.
- **Resistance** to stolen verifier attack: The GCS stores $RPMD_{ID_O}$, which is a salted hash digest of RP_{ID_O}. Crucially, the actual RP_{ID_O} is stored encrypted (C_{RP}) on the OD and is derived via biometric authentication and a local PIN. Stealing $RPMD_{ID_O}$ from the GCS database does not directly reveal RP_{ID_O} or P_{ID_O}, due to the one-way property of hash functions and the effective use of salting. This makes the protocol robustly resistant to stolen verifier attacks, which Jafarian's cryptanalysis [19] highlights as a significant vulnerability in other schemes. This effectively addresses a critical aspect of requirement 11.
- **Efficiency** (lightweight design): The protocol relies exclusively on lightweight symmetric cryptographic operations (AES encryption/decryption) and efficient hash functions (SHA-256), along with key derivation functions (KDFs). These operations are computationally efficient and are thus exceptionally well suited for resource-constrained IoD components like operator devices and drones [1,11]. This design choice strongly meets requirement 12.

5.2. BAN Logic Analysis

(Follows standard BAN logic proof structure [14,24,28]). 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. $OD \mid\equiv \text{fresh}(T_{ID}), OD \mid\equiv \text{fresh}(PIN)$.

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's provisioning of RP_{ID_O} and shared knowledge of P_{ID_O}'s function) to compute KS. OD believes the contents of $\{PIN, T_{ID}, RP_{ID_O}\}_{KS}$ were intended for GCS and are fresh due to T_{ID} and PIN.
2. GCS receives M1. GCS computes KS_{GCS} using P_{ID_O} (from its knowledge of ID_O's associated random password and the T_{ID} from M_1). Using the Message-Meaning Rule on $\{X\}_{KS_{GCS}}$ (where $X = (PIN, T_{ID}, RP_{ID_O})$), GCS deduces $GCS \mid\equiv OD \mid\sim X$.
3. GCS verifies the freshness of T_{ID} by checking $T_{ID} > LLT_O$ and $T_{current\_GCS} - T_{ID} \leq \Delta_t$. Since T_{ID} is fresh, it believes $\#(X)$. By the Nonce Verification Rule, $GCS \mid\equiv OD \mid\equiv X$. GCS then verifies RP_{ID_O} from X against its stored $RPMD_{ID_O}$. This confirms OD's knowledge of the correct RP_{ID_O} and P_{ID_O}, and its freshness.
4. OD receives $M2 = \{PIN'\}_{KS'}$. OD attempts to decrypt using its locally derived KS. By the Message-Meaning Rule, $OD \mid\equiv GCS \mid\sim PIN'$.
5. Since OD believes $\#(PIN)$ (as it generated PIN freshly) and PIN' matches its original PIN, by the Nonce Verification Rule, $OD \mid\equiv GCS \mid\equiv PIN$. This confirms that the GCS successfully decrypted the 'OTAC' using the shared key KS and thus possesses the correct PIN, thereby authenticating the GCS to the OD.
6. Both OD and GCS can now believe KS is a good shared key, achieving mutual authentication: $OD \mid\equiv OD \overset{KS}{\leftrightarrow} GCS$ 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 [15], similar to analyses in [5,6,19].

Oracle Setup and Adversarial Capabilities

Standard oracles are assumed: '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 that the adversary's advantage $Adv^{AKE}_{\text{Bio-2FA-IoD}}(\mathcal{A})$ in distinguishing the real session key from a random one is negligible.

- **Game** G_0: This is the real AKE security game, where the adversary interacts with honest instances of the protocol and attempts to distinguish a fresh session key from a random string via the 'Test' query.
- **Game** G_1 (random oracles for hash/KDF): In this game, the hash function H and the key derivation function KDF are modeled as ideal random oracles. An adversary's advantage in distinguishing G_1 from G_0 is negligible, bounded by the collision resistance property of the hash function and the pseudorandomness of the KDF. Specifically, $|Adv_{G_0}(\mathcal{A}) - Adv_{G_1}(\mathcal{A})| \leq q_H^2/2^{|H|_{output}} + q_{KDF}^2/2^{|KDF|_{range}}$, where q_H and q_{KDF} are

the number of queries to the respective oracles, and $|H|_{output}$ and $|KDF|_{range}$ are their output/range sizes. This quantity is negligible for cryptographically secure functions.

- **Game** G_2 (fuzzy extractor simulation): This game simulates the fuzzy extractor 'Rec' algorithm. The output $K_{\beta O}$ of $Rec(\beta'_O, HD_O)$ is indistinguishable from a random string if the input biometric sample β'_O is not sufficiently close to the enrolled β_O, or if the adversary attempts to guess $K_{\beta O}$ from the public helper data HD_O. The advantage difference between G_1 and G_2 is bounded by ϵ_{FE}, which represents the security of the fuzzy extractor against an adversary predicting its output or the extracted key from public helper data or invalid inputs [5,12]. This term is also negligible.
- **Game** G_3 (symmetric cipher indistinguishability): In this game, ciphertexts produced by the symmetric encryption $E_{KS}(\cdot)$ (used for 'OTAC') are replaced by random strings if the session key KS is unknown to the adversary. If the session key KS is known (e.g., through a 'Reveal' query), the simulation perfectly mimics the real game. The difference in advantage between G_2 and G_3 is bounded by the IND-CPA security of the symmetric encryption scheme E, denoted as ϵ_{SymEnc}. For a secure scheme like AES-128, ϵ_{SymEnc} is negligible.
- **Game** G_4 (attacking KS via 'Test' query): Consider the adversary making a 'Test(Π^s_{OD})' query on a fresh session instance for the session key $KS = KDF(P_{ID_O}, T_{ID})$. A session is considered "fresh" if its session key has not been directly revealed by a 'Reveal' query, and its long-term secrets (P_{ID_O} for the OD) have not been compromised by a 'Corrupt' query on the corresponding principal prior to the session's completion. If $\mathcal{A}$ has not successfully executed 'Corrupt(OD$_O$)' for U_O prior to this session becoming complete, then P_{ID_O} remains unknown to $\mathcal{A}$. Since T_{ID} is a fresh, unique timestamp for each session, and KDF is modeled as a random oracle, KS is computationally indistinguishable from a random string to $\mathcal{A}$ without knowledge of P_{ID_O}. 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). However, $RPMD_{ID_O} = H(RP_{ID_O} \| Salt_1)$ and $RP_{ID_O} = D_{KDF(P_{ID_O}, Salt_2)}(C_{RP})$. These layers make guessing P_{ID_O} from leaked server-side data computationally infeasible. 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, and the adversary cannot distinguish this from the real key. The only way $\mathcal{A}$ can distinguish this game from G_3 is by successfully guessing P_{ID_O} (without corruption) or by breaking the KDF (which is covered in G_1). The probability of guessing P_{ID_O} from available information (without corruption) is negligible, denoted as P_{guess}. Therefore, $|Adv_{G_3}(\mathcal{A}) - Adv_{G_4}(\mathcal{A})| \leq P_{guess}$.
- **Game** G_5 (final game): In this final game, the output of the 'Test' query for any fresh instance is always a truly random string. Since the adversary cannot distinguish this from the real session key (as established in previous games), its advantage in this game is $\Pr[\mathcal{A} \text{ wins in } G_5] = 1/2$, which means $Adv_{G_5}(\mathcal{A}) = 0$.

Combining these, $Adv^{AKE}_{\text{Bio-2FA-IoD}}(\mathcal{A}) \leq \epsilon_{FE} + \epsilon_{SymEnc} + P_{guess} + q_H^2/2^{|H|_{output}} + q_{KDF}^2/2^{|KDF|_{range}}$. If P_{ID_O} is chosen from a sufficiently large space or effectively protected by $K_{\beta O}$ (which itself has high entropy), then P_{guess} is negligible. Thus, if the underlying cryptographic primitives and the fuzzy extractor are secure, the session key KS derived by Bio-2FA-IoD 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 energy consumption, comparing these aspects with several contemporary authentication schemes for IoD environments. Fur-

thermore, a comprehensive simulation analysis using the Contiki OS and Cooja simulator is presented to demonstrate the protocol's performance in a realistic IoD network scenario.

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 the recent IoD security literature (e.g., Nyangaresi et al. [5], Jan et al. [1], and values from Khedr [8] where applicable):

- T_H (SHA-256 hash): ≈ 0.0025 ms [5];
- T_{FE} (Fuzzy Extractor-Rec): ≈ 0.0098 ms [5];
- T_S (AES-128 Enc/Dec): ≈ 0.0046 ms [5];
- 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 is used for key derivation not password stretching from low-entropy input).

Bio-2FA-IoD computational cost (authentication phase):

- **Operator's** device (OD):
  - (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 3 presents a comparative overview of computational costs. The proposed Bio-2FA-IoD, with a total estimated 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. [13], with costs around 0.448 ms) or HECC (e.g., HCALA by Berini et al. [2], at 3.3873 ms). Its cost is comparable to, and slightly higher than, the PUF and biometric-based scheme by Nyangaresi et al. [5] (**0.0388** ms), which involves PUF operations directly in the authentication flow for multiple entities. In contrast, Jan et al.'s scheme [1], 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 ($\approx$0.016 ms). 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. This makes it exceptionally well suited for real-time IoD operations on resource-constrained devices, performing significantly faster than many ECC/HECC or HMAC-based protocols.

Table 3. 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. [5]	0.0388	PUF, FE, Hash
Jan et al. [1]	17.7939	HMAC-SHA1, Hash, XOR
Berini et al. (HCALA, 2023) [2]	3.3873	HECC, Hash, AES
Khalid et al. (HOOPOE, 2023) [6]	8.343	AES, RSA, Hash, Signature
Akram et al. (2023) [13]	0.44819	ECC, Hash, Signature
Najafi et al. (EPUF-Auth, 2025) [11]	≈0.016	PUF (DRAM), Hash, XOR

Note: Bold indicates the proposed protocol for emphasis.

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 and AES-128 in CBC mode with a 128-bit IV:

- 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 a multiple of 128 bits. For 40-byte plaintext, it will require three blocks plus IV, thus $3 \times 128 = 384$ 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**.
- Size of M_2: $128(\text{for } PIN') =$ **128 bits**.

Total communication cost: $544 + 128 =$ **672 bits** (or 84 bytes).

Comparison with other schemes: Table 4 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. [5] (**2464** bits, six messages), Jan et al. [1] (**3720** bits, four messages), and HCALA by Berini et al. [2] (**1536** bits, three messages). For example, our protocol reduces bandwidth consumption by approximately 73% compared to Nyangaresi et al. [5] and 82% compared to Jan et al. [1]. The EPUF-based protocol by Najafi et al. [11] also boasts low communication costs (around 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. Reduced communication overhead directly translates to lower energy consumption and improved reliability in potentially lossy wireless channels.

Table 4. 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. [5]	6	2464	Nyangaresi et al. [5]
Jan et al. [1]	4	3720	Jan et al. [1]
Berini et al. (HCALA, 2023) [2]	3	1536	Berini et al. [2]
Khalid et al. (HOOPOE, 2023) [6]	2 (D-GSS)	1280	Khalid et al. [6]
Akram et al. (2023) [13]	2	1152	Akram et al. [13]
Najafi et al. (EPUF-Auth, 2025) [11]	2	1536	Najafi et al. [11] (192 bytes)

Note: Bold indicates the proposed protocol for emphasis.

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 based on our computational and communication cost analyses. 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 typically 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 [2], using HECC, report an energy consumption for communication around 3.38×10^{-4} J (338 µJ) for three 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 [6] also provides detailed energy analysis, showing that RSA/AES operations are more energy intensive than the symmetric primitives predominantly used in Bio-2FA-IoD. This makes Bio-2FA-IoD highly advantageous for extending operational time and minimizing the power burden on IoD devices.

6.4. Simulation-Based Performance Analysis

To further validate the practical performance of Bio-2FA-IoD in a realistic IoD environment, we conducted a comprehensive simulation analysis using the Contiki OS and Cooja simulator [16,17]. This simulation aims to evaluate key metrics such as average end-to-end latency, packet delivery ratio (PDR), and average radio ON time (as an indicator of energy consumption).

6.4.1. Simulation Environment and Setup

The simulation environment was configured to mimic a typical IoD deployment scenario. We used Contiki OS (version 2.7) as the operating system for the simulated nodes and the Cooja simulator (version 3.0) for network emulation. The network topology consisted of 80 client nodes (representing operator devices and drones acting as relays) and 1 sink node (representing the ground control station). This multipoint-to-point traffic scenario is common in IoD applications where multiple drones transmit data to a central control station [17]. The wireless medium was simulated using Cooja's Unit Disk Graph Model (UDGM), allowing for configurable transmission and interference ranges, as well as packet reception ratios. We set the transmission range to 50 m and the interference range to 55 m. To evaluate performance under varying network conditions, the packet reception ratio (RX ratio) was set to 70%, representing a moderately lossy wireless channel common in IoD environments [17]. The simulation ran for a total duration of 1 h (3600 s). Each client node was configured to generate application messages at a random interval between 2 to 6 s (average 4 s), simulating the periodic transmission of authentication requests and other data.

6.4.2. Performance Metrics and Measurement

The following performance metrics were measured during the simulation:

- **Average** end-to-end latency (s): This measures the average time taken for an authentication message to travel from the operator device (via the drone relay) to the ground

control station. It is calculated from the time a packet is sent by the OD until it is successfully received by the GCS.

- **Packet** delivery ratio (PDR, %): This represents the percentage of successfully received authentication messages at the GCS out of the total messages sent by the ODs. A higher PDR indicates greater reliability of the authentication protocol.
- **Average** radio ON time (%): As a proxy for energy consumption, this metric indicates the percentage of time a node's radio transceiver is active (transmitting or listening). Lower radio ON time implies less energy consumption and longer battery life [17].
- **Control** traffic overhead (packets/h): This measures the total number of non-application-specific control packets exchanged within the network during the authentication process, providing insight into the protocol's efficiency in terms of network overhead.

Data for these metrics were extracted from the Cooja simulation logs, similar to the methodology described in [16,17]. Each data point represents an average value over multiple simulation runs to ensure statistical significance.

6.4.3. Simulation Results and Analysis

Table 5 summarizes the performance results of Bio-2FA-IoD compared to other general IoD authentication protocols under the described simulation environment.

Table 5. Simulation-based performance comparison.

Performance Metric	Bio-2FA-IoD (Proposed)	Protocol A (ECC-Based)	Protocol B (HMAC-Based)	Nyangaresi et al. [5] (Biometric, PUF)
Average end-to-end latency (s)	**0.45**	1.2	0.7	0.5
Packet delivery ratio (PDR, %)	**95**	88	90	93
Average radio ON time (%)	**1.8**	5.5	3.0	2.0
Control traffic overhead (packets/h)	**150**	400	250	300

Note: Bold indicates the proposed protocol for emphasis.

Average End-to-End Latency

As shown in Table 5, Bio-2FA-IoD achieved an average end-to-end latency of **0.45 s**. This remarkably low latency is primarily attributed to its lightweight symmetric cryptographic operations and minimal message exchanges. Compared to Protocol A, which relies on more computationally intensive ECC operations, Bio-2FA-IoD is significantly faster (0.45 s vs. 1.2 s). Even against other lightweight symmetric or biometric/PUF schemes (Protocol B: 0.7 s, Nyangaresi et al.: 0.5 s), Bio-2FA-IoD demonstrates superior responsiveness, making it ideal for time-sensitive IoD operations.

Packet Delivery Ratio (PDR)

Bio-2FA-IoD achieved a high packet delivery ratio (PDR) of **95%** in the simulated lossy environment. This indicates the protocol's robustness in reliably delivering authentication messages. The high PDR is a direct consequence of its efficient use of the wireless medium, reducing contention and subsequent packet drops that can occur with heavier traffic or more complex handshakes (e.g., Protocol A at 88%, Protocol B at 90%). The streamlined authentication flow minimizes the opportunities for packet loss, even in challenging radio conditions.

Average Radio ON Time (Energy Consumption)

The simulation results show that Bio-2FA-IoD maintained an impressively low average radio ON time of just **1.8%**. This directly translates to minimal energy consumption, extending the operational lifespan of battery-powered ODs and drones. The protocol's design,

which emphasizes few messages and rapid processing, ensures that the energy-intensive radio component is active for the shortest possible duration. This is substantially lower than Protocol A (5.5%) and Protocol B (3.0%), and even slightly better than Nyangaresi et al. [5] (2.0%), highlighting its energy efficiency, a critical advantage for IoD deployments.

Control Traffic Overhead

Bio-2FA-IoD demonstrated a very low control traffic overhead of **150** packets per hour. This minimal overhead is crucial for preventing network congestion and efficiently utilizing the limited bandwidth available in IoD networks. The protocol's direct communication model and reliance on compact authentication messages reduce the need for extensive signaling or state synchronization, differentiating it from protocols that might involve more frequent control plane updates (e.g., Protocol A at 400 packets/h, Protocol B at 250 packets/h, Nyangaresi et al. [5] at 300 packets/h). This efficient use of network resources ensures scalability and stability in dense IoD deployments. In summary, the simulation results strongly corroborate the theoretical analysis, affirming Bio-2FA-IoD's position as a highly efficient and lightweight authentication solution for the Internet of Drones. Its superior performance across critical metrics like latency, PDR, and energy consumption, under realistic network conditions, makes it well suited for resource-constrained IoD components.

6.5. Comparison of Security and Functionality Features

Table 6 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 the IoD literature [1,2,5,6,30]. The proposed Bio-2FA-IoD effectively provides strong mutual authentication and resists keylogging and shoulder-surfing attacks by adapting Khedr's logic [8] with biometrics. Its lightweight nature is a key advantage for the 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 [2] and HOOPOE [6] 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.

Table 6. Comparison of security and functionality features.

Feature	Khedr [8] (Adapted)	Nyangaresi [5]	Jan [1]	HCALA [2]	HOOPOE [6]	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**

Table 6. *Cont.*

Feature	Khedr [8] (Adapted)	Nyangaresi [5]	Jan [1]	HCALA [2]	HOOPOE [6]	Bio-2FA-IoD (Proposed)
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**

7. Proposed Extensions to Bio-2FA-IoD

This section conceptually outlines potential extensions to the Bio-2FA-IoD protocol aimed at enhancing perfect forward secrecy (PFS) and improving user anonymity. These extensions introduce more sophisticated cryptographic mechanisms, acknowledging their potential impact on computational and communication overhead, and serve as crucial directions for future research.

7.1. Enhanced Session Key Agreement with Diffie–Hellman for Perfect Forward Secrecy

The current Bio-2FA-IoD protocol's session key KS is derived from the operator's long-term local password P_{ID_O} and a timestamp, which provides limited perfect forward secrecy (PFS). To achieve stronger PFS, we propose integrating an ephemeral Diffie–Hellman (DH) key exchange during the authentication phase. This would ensure that the compromise of long-term secrets (P_{ID_O} or $RPMD_{ID_O}$) does not compromise past session keys.

7.1.1. Conceptual Integration

The integration would modify step 2 and step 5 of the authentication and key exchange phase:

2. **OD**—OTAC generation with DH key generation:

- The OD generates a fresh random exponent x_{OD} and computes its DH public key $Pub_{OD} = g^{x_{OD}} \pmod{p}$, where g and p are globally agreed-upon DH parameters.
- The OD generates a current timestamp T_{ID} and a fresh random nonce PIN.
- The ephemeral session key KS is initially derived as $KS = KDF(P_{ID_O}, T_{ID})$. This KS would now serve as a temporary session key to protect the DH exchange and other critical parameters.
- The OD computes $OTAC = E_{KS}(PIN \| T_{ID} \| RP_{ID_O} \| Pub_{OD})$.
- OD sends $M_1 = (ID_O, OTAC, T_{ID})$ to drone D_i.

4. **GCS**—verification process and DH key generation:

- Upon receiving M_1, GCS retrieves $(RPMD_{ID_O}, Salt_1, LLT_O)$ for ID_O.
- GCS computes $KS_{GCS} = KDF(P_{ID_O}, T_{ID})$.
- It decrypts $(PIN', T'_{ID}, RP'_{ID_O}, Pub'_{OD}) = D_{KS_{GCS}}(OTAC)$.
- GCS verifies the authenticity of RP'_{ID_O} and T'_{ID} as in the original protocol.
- GCS generates its own fresh random exponent x_{GCS} and computes its DH public key $Pub_{GCS} = g^{x_{GCS}} \pmod{p}$.
- GCS computes the shared DH secret $SS = (Pub'_{OD})^{x_{GCS}} \pmod{p}$.
- The final session key for the communication session is now $KS_{final} = KDF(SS, T_{ID})$.

5. **Mutual** authentication response ($GCS \rightarrow D_i \rightarrow OD$):
   - If all checks pass, GCS updates $LLT_O = T_{ID}$, and sends $M_2 = (PIN' \| Pub_{GCS})$ to D_i. The PIN' value remains encrypted with the initial KS for integrity, or a separate MAC is used.
   - Drone D_i forwards M_2 to the OD.
6. **OD**—final verification (mutual authentication) and final session key derivation:
   - OD receives M_2 containing PIN' and Pub_{GCS}.
   - OD compares PIN' with its original PIN. If match, GCS is authenticated.
   - OD computes the shared DH secret $SS' = (Pub_{GCS})^{x_{OD}} \pmod{p}$.
   - The final session key is $KS'_{final} = KDF(SS', T_{ID})$. Since $SS = SS'$, then $KS_{final} = KS'_{final}$.

7.1.2. Security Implications

This DH integration would provide **perfect forward secrecy** because the session key KS_{final} is derived from ephemeral secrets (x_{OD} and x_{GCS}) that are generated uniquely for each session and are not stored long-term. Even if an attacker compromises the long-term P_{ID_O} or $RPMD_{ID_O}$, past session keys cannot be re-derived. The initial KS derived from P_{ID_O} would protect the DH public key exchange, mitigating potential man-in-the-middle attacks during the DH part of the exchange, as the 'OTAC' itself is authenticated.

7.1.3. Performance Considerations

Integrating classical DH operations (exponentiation modulo a large prime) would significantly increase the computational cost compared to the current hash- and symmetric-key-only operations. A single modular exponentiation is considerably more expensive than a hash or AES operation. This would impact the "Computational Cost" figures presented in Section 5.1. Similarly, including DH public keys would slightly increase the "Communication Cost", although this is often a small overhead compared to the security benefits. The overall lightweight nature of the protocol would be somewhat affected, but the enhanced PFS is often a trade-off worth considering for critical IoD applications.

7.1.4. Note on Quantum Diffie–Hellman

The paper by Chen et al. [31] proposes a Diffie–Hellman quantum session key establishment protocol that operates without entanglement. While intriguing, the direct integration of such quantum protocols into Bio-2FA-IoD, which is designed for classical computational environments and IoD devices that currently do not support quantum computing or quantum states manipulation, represents a highly specialized and complex future research direction. It would require a fundamental shift in the underlying cryptographic model and hardware capabilities of IoD components, distinct from classical cryptographic enhancements discussed here. Our focus in this section remains on classical cryptographic extensions for near-term practical application.

7.2. Sybil-Free Pseudonym Management for Enhanced Anonymity

The current protocol explicitly sends ID_O, offering only partial user anonymity. To achieve stronger user anonymity and unlinkability while preserving accountability (e.g., for misbehaving operators), we propose conceptually integrating a Sybil-free pseudonym management scheme, drawing inspiration from Martucci et al.'s work [29]. This approach allows operators to use unlinkable pseudonyms across different "service contexts" ($\mathcal{C}$) within the IoD ecosystem, while preventing a single operator from creating multiple identities (Sybil attacks) to manipulate trust or obscure malicious activities.

7.2.1. Conceptual Integration

This extension would primarily impact the registration phase and the format of ID_O in the authentication request:

1. **TA's** role (initial identifier issuance):
   - During initial registration, the trusted authority (TA) acts as a trusted issuer, providing each legitimate operator (U_O) with a unique, unforgeable "initial identifier" (u). This u would be akin to a non-transferable e-token dispenser, incorporating cryptographic elements like a Camenisch and Lysyanskaya (CL) signature. This process would involve algorithms like IKg, UKg, $Obtain$, and $Issue$.
   - This initial identifier u is securely provisioned to the operator's device (OD).
2. **OD's** role (pseudonym generation and storage):
   - The OD, using the initial identifier u, generates a new pseudonym $p_{u_O}^{\mathcal{C}}$ for each specific IoD "service context" ($\mathcal{C}$) (e.g., a drone mission type, a geographical area) it participates in. This involves a 'Sign' algorithm that binds a freshly generated public key ($pk_{u,\mathcal{C}}$) to the pseudonym, producing a unique serial number (S) and a pseudonym certificate (τ).
   - The OD uses S and τ instead of ID_O in the authentication message M_1. The new public key $pk_{u,\mathcal{C}}$ can be used for signing messages originating from the pseudonym.
3. **GCS's** role (pseudonym verification and Sybil detection):
   - Upon receiving an authentication request with a pseudonym (S, τ), the GCS uses a 'Verify' algorithm to check its validity and authenticity against the TA's public key.
   - The 'Identify' algorithm [29] could be invoked by the GCS (or a central auditing entity) if two pseudonyms from the same service context are observed to be potentially linked, allowing the detection of Sybil attacks (i.e., if an operator attempts to create multiple identities within the same context). This can compute the initial identifier u of the owner, enabling accountability for misbehavior (e.g., revocation).

7.2.2. Security Implications

This pseudonym management system would provide enhanced **user anonymity** and **unlinkability** across different IoD service contexts. An adversary would be computationally unable to link an operator's activities in one context to their activities in another, thus preventing comprehensive profiling. Furthermore, the "Sybil-free" property offers strong resistance against **Sybil attacks**, a critical threat to reputation systems and privacy in distributed environments. The ability to detect and potentially revoke misbehaving pseudonyms ensures accountability while preserving anonymity for compliant users.

7.2.3. Performance Considerations

The cryptographic primitives underlying Sybil-free pseudonyms (e.g., anonymous credentials, group signatures, zero-knowledge proofs) are generally more computationally intensive than basic symmetric key operations or simple hashing. Algorithms like Camenisch and Lysyanskaya signatures and non-interactive zero-knowledge proofs can introduce significant overheads in terms of processing time and message size. This would likely increase the "computational cost" on the od, and potentially the "communication cost" due to larger certificate sizes, affecting the lightweight property of the current Bio-2FA-IoD. A careful trade-off analysis between enhanced anonymity/Sybil resistance and performance would be critical during detailed design and future evaluations.

8. Conclusions

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. The simulation-based analysis using the Contiki OS and Cooja simulator [16,17] further corroborated these findings, demonstrating an average end-to-end latency of 0.45 s, a high packet delivery ratio of 95%, and an impressively low average radio ON time of 1.8% for energy consumption, along with a minimal control traffic overhead of 150 packets per hour. 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. The demonstrated superior efficiency in both theoretical analysis and simulation results, coupled with its strong security properties against relevant IoD threats, positions Bio-2FA-IoD as a highly viable and practical solution for securing critical drone operations.

Future Work

Based on the current analysis and conceptual extensions, several promising directions for future research are identified to enhance Bio-2FA-IoD's capabilities and address broader IoD challenges.

- **Scalability** for swarm operations and peer-to-peer authentication: The current protocol is optimized for operator-to-GCS authentication via a single drone relay. Future research will investigate extensions to support large-scale drone swarm operations, potentially through hierarchical authentication schemes or group key management. Adapting the protocol for secure peer-to-peer drone-to-drone authentication within dynamic mesh networks will also be a key area of focus.
- **Data** privacy and compliance: Further investigation into the implications of biometric data handling will include a more in-depth analysis of compliance with data privacy regulations such as GDPR. Emphasizing that the raw biometric template is not stored, and helper data (HD_O) is kept locally and securely on the operator's device, significantly contributes to privacy. Future work will explore formal privacy-preserving mechanisms if biometric data processing were to extend to cloud environments.

- **Post**-quantum cryptography integration: As quantum computing capabilities evolve, assessing and integrating post-quantum cryptographic primitives into Bio-2FA-IoD will be crucial to ensure long-term security against quantum attacks.

Author Contributions: H.K.: conceptualization, methodology, validation, formal analysis, investigation, writing—original draft preparation, writing—review and editing, supervision, project administration; S.P.: funding acquisition, writing—review and editing, supervision, project administration. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2022R1A2C2005705). This research was financially supported by Hansung University for Seunghyun Park.

Data Availability Statement: Data is contained within the article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Jan, S.U.; Qayum, F.; Khan, H.U. Design and Analysis of Lightweight Authentication Protocol for Securing IoD. *IEEE Access* **2021**, *9*, 69287–69306. [CrossRef]
2. Berini, A.D.E.; Ferrag, M.A.; Farou, B.; Seridi, H. HCALA: Hyperelliptic curve-based anonymous lightweight authentication scheme for Internet of Drones. *Pervasive Mob. Comput.* **2023**, *92*, 101798. [CrossRef]
3. Jan, S.U.; Khan, H.U. Identity and Aggregate Signature-Based Authentication Protocol for IoD Deployment Military Drone. *IEEE Access* **2021**, *9*, 126038–126050. [CrossRef]
4. Hussain, S.; Farooq, M.; Alzahrani, B.A.; Albeshri, A.; Alsubhi, K.; Chaudhry, S.A. An Efficient and Reliable User Access Protocol for Internet of Drones. *IEEE Access* **2023**, *11*, 59689–59700. [CrossRef]
5. Nyangaresi, V.O.; Al-Joboury, I.M.; Al-sharhanee, K.A.; Najim, A.H.; Abbas, A.H.; Hariz, H.M. A biometric and physically unclonable function-Based authentication protocol for payload exchanges in internet of drones. *e-Prime-Adv. Electr. Eng. Electron. Energy* **2024**, *7*, 100471. [CrossRef]
6. Khalid, H.; Hashim, S.J.; Hashim, F.; Ahamed, S.M.S.; Chaudhary, M.A.; Altarturi, H.H.M.; Saadoon, M. HOOPOE: High Performance and Efficient Anonymous Handover Authentication Protocol for Flying Out of Zone UAVs. *IEEE Trans. Veh. Technol.* **2023**, *72*, 10906–10920. [CrossRef]
7. Wu, T.; Guo, X.; Chen, Y.; Kumari, S.; Chen, C. Amassing the Security: An Enhanced Authentication Protocol for Drone Communications over 5G Networks. *Drones* **2022**, *6*, 10. [CrossRef]
8. Khedr, W.I. Improved keylogging and shoulder-surfing resistant visual two-factor authentication protocol. *J. Inf. Secur. Appl.* **2018**, *39*, 41–57. [CrossRef]
9. Khan, A.A.; Kumar, V.; Ahmad, M. An elliptic curve cryptography based mutual authentication scheme for smart grid communications using biometric approach. *J. King Saud Univ.—Comput. Inf. Sci.* **2022**, *34*, 698–705. [CrossRef]
10. Wu, T.Y.; Wu, H.; Kumari, S.; Chen, C.M. An enhanced three-factor based authentication and key agreement protocol using PUF in IoMT. *Peer-to-Peer Netw. Appl.* **2025**, *18*, 83. [CrossRef]
11. Najafi, F.; Kaveh, M.; Mosavi, M.R.; Brighente, A.; Conti, M. EPUF: An Entropy-Derived Latency-Based DRAM Physical Unclonable Function for Lightweight Authentication in Internet of Things. *IEEE Trans. Mob. Comput.* **2025**, *24*, 2422–2436. [CrossRef]
12. Dodis, Y.; Reyzin, L.; Smith, A. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. In *Eurocrypt 2004*; LNCS 3027; Springer: Berlin/Heidelberg, Germany, 2004; pp. 523–540.
13. Akram, M.A.; Ahmad, H.; Mian, A.N.; Jurcut, A.D.; Kumari, S. Blockchain-based privacy-preserving authentication protocol for UAV networks. *Comput. Netw.* **2023**, *224*, 109638. [CrossRef]
14. Burrows, M.; Abadi, M.; Needham, R. A Logic of Authentication. *ACM Trans. Comput. Syst. (TOCS)* **1989**, *8*, 18–36. [CrossRef]
15. Bellare, M.; Pointcheval, D.; Rogaway, P. Authenticated Key Exchange Secure Against Dictionary Attacks. In *Eurocrypt 2000*; LNCS 1807; Springer: Berlin/Heidelberg, Germany, 2020; pp. 139–155.
16. Tall, H.; Chalhoub, G.; Misson, M. Implementation of IEEE 802.15.4 Unslotted CSMA/CA Protocol on Contiki OS. *Int. J. Eng. Res. Technol. (IJERT)* **2016**, *4*, 1–5.
17. Ali, H. A Performance Evaluation of RPL in Contiki: A Cooja Simulation Based Study. Master's Thesis, Blekinge Institute of Technology, Karlskrona, Sweden, 2012.
18. Akram, J.; Akram, A.; Jhaveri, R.H.; Alazab, M.; Chi, H. BC-IoDT: Blockchain-based Framework for Authentication in Internet of Drone Things. In Proceedings of the ACM MobiCom Workshop on Drone Assisted Wireless Communications for 5G and Beyond (DroneCom '22), Sydney, Australia, 17 October 2022; ACM: New York, NY, USA, 2022.

19. Jafarian, I. Cryptanalysis of Nikooghadam et al.'s Lightweight Authentication Protocol for Internet of Drones. *arXiv* **2023**, arXiv:2311.02512.
20. Nikooghadam, M.; Amintoosi, H.; Islam, S.H.; Moghadam, M. A provably secure and lightweight authentication scheme for Internet of Drones for smart city surveillance. *J. Syst. Archit.* **2021**, *115*, 101955. [CrossRef]
21. Zhang, N.; Jiang, Q.; Li, L.; Ma, X.; Ma, J. An efficient three-factor remote user authentication protocol based on BPV-FourQ for internet of drones. *Peer-to-Peer Netw. Appl.* **2021**, *14*, 3319–3332. [CrossRef]
22. Zhang, S.; Liu, Y.; Han, Z.; Yang, Z. A Lightweight Authentication Protocol for UAVs Based on ECC Scheme. *Drones* **2023**, *7*, 315. [CrossRef]
23. Cabuk, U.C.; Dalkilic, G.; Dagdeviren, O. CoMAD: Context-Aware Mutual Authentication Protocol for Drone Networks. *IEEE Access* **2021**, *9*, 78400–78414. [CrossRef]
24. El-Zawawy, M.A.; Brighente, A.; Conti, M. Authenticating Drone-Assisted Internet of Vehicles Using Elliptic Curve Cryptography and Blockchain. *IEEE Trans. Netw. Serv. Manag.* **2023**, *20*, 1775–1789. [CrossRef]
25. Gupta, A.; Barthwal, A.; Vardhan, H.; Kakria, S.; Kumar, S.; Parihar, A.S. Evolutionary study of distributed authentication protocols and its integration to UAV-assisted FANET. *Multimed. Tools Appl.* **2023**, *82*, 42311–42330. [CrossRef]
26. Dolev, D.; Yao, A. On the security of public key protocols. *IEEE Trans. Inf. Theory* **1983**, *29*, 198–208. [CrossRef]
27. Vangala, A.; Agrawal, S.; Das, A.K.; Pal, S.; Kumar, N.; Lorenz, P.; Park, Y. Big Data-Enabled Authentication Framework for Offshore Maritime Communication Using Drones. *IEEE Trans. Veh. Technol.* **2024**, *73*, 10196–10210. [CrossRef]
28. Hasson, M.; Yassin, A.A.; Yassin, A.J.; Rashid, A.M.; Yaseen, A.A.; Alasadi, H. Password authentication scheme based on smart card and QR code. *Indones. J. Electr. Eng. Comput. Sci.* **2021**, *23*, 140–149. [CrossRef]
29. Martucci, L.A.; Ries, S.; Mühlhäuser, M. Sybil-Free Pseudonyms, Privacy and Trust: Identity Management in the Internet of Services. *Inf. Media Technol.* **2011**, *6*, 1001–1015. [CrossRef]
30. Fatima, S.; Akram, M.A.; Mian, A.N.; Kumari, S.; Chen, C.M. On the Security of a Blockchain and PUF-Based Lightweight Authentication Protocol for Wireless Medical Sensor Networks. *Wirel. Pers. Commun.* **2024**, *136*, 1079–1106. [CrossRef]
31. Chen, Y.; Hsiang, C.; Wang, L.-C.; Chou, Y.-Y.; Chou, J.-S. A Diffie-Hellman quantum session key establishment protocol without entanglement. *J. Physics Conf. Ser.* **2019**, *1308*, 012019.