

Article

AERO-GUARD: A Post-Quantum Mutual Authentication Drone Protocol with Homomorphic Encryption for Secure Road Surveillance in Smart Cities

Albandari Alsumayt ^{1,*} , Arwa Almalki ² , Reema Almassary ¹, Hotoon Alghamdi ¹, Reemas Alqahtani ¹, Ryouf Alzuabie ¹, Reham Alharthi ¹ and Naya Nagy ¹

- ¹ Saudi Aramco Cybersecurity Chair, Networks and Communications Department, College of Computer Science and Information Technology, Imam Abdulrahman Bin Faisal University, Dammam 31441, Saudi Arabia; 2220006331@iau.edu.sa (R.A.); 2220005293@iau.edu.sa (H.A.); 2220002428@iau.edu.sa (R.A.); 2220001327@iau.edu.sa (R.A.); 2220005875@iau.edu.sa (R.A.); nmnagy@iau.edu.sa (N.N.)
- ² Computer Information Systems Department, College of Computer Science and Information Technology, Imam Abdulrahman Bin Faisal University, Dammam 31441, Saudi Arabia; ammalki@iau.edu.sa
- * Correspondence: afaalsumayt@iau.edu.sa

Abstract

This paper presents AERO-GUARD, a formally verified drone authentication and road surveillance system that integrates Kyber post-quantum key encapsulation, physical unclonable functions (PUFs), decentralized IPFS-based identity storage, and blockchain-anchored audit logging. AERO-GUARD operates across three phases, key provisioning, enrollment, and authentication, enforcing mutual authentication, replay resistance, and privacy-preserving comparison through an off-chain evaluator (OCE) that performs homomorphic subtraction on encrypted PUF responses without accessing plaintext secrets. The protocol is modeled and verified using ProVerif 2.05 under the Dolev–Yao adversary model. To evaluate the system beyond theoretical verification, a simulation environment was developed to replicate realistic road conditions, incorporating a simulated road network and a virtual drone traversing monitored routes. An AI model is deployed to perform real-time detection of suspicious and anomalous activities along the road. All detection events are surfaced through a centralized monitoring dashboard that provides authorized personnel with live alerts, a drone camera livestream with detection annotations, and contextual drone telemetry, enabling timely and informed incident response. Formal verification results demonstrate that AERO-GUARD satisfies the targeted security properties, including mutual authentication, secrecy preservation, and replay resistance, confirming the protocol’s resilience against common authentication attacks.

Keywords: post-quantum cryptography; CRYSTALS-Kyber; physical unclonable functions; drone authentication; blockchain; IPFS; mutual authentication; road surveillance; anomaly detection; smart city



Academic Editors: Francesco Buccafurri and Sara Lazzaro

Received: 20 June 2026

Revised: 28 July 2026

Accepted: 28 July 2026

Published: 4 August 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

Road traffic accidents remain one of the major causes of death and injury throughout the world. Approximately 1.19 million deaths occur yearly from road traffic collisions. Road traffic injuries are still the leading cause of death in children and young adults of 5–29 years old [1]. The effects of accidents and injuries in remote areas like desert highways, border roads, and mountainous terrain can be worse than elsewhere due to the limited surveillance equipment available and the delay in responding to emergencies.

Recently, advancements made in Unmanned Aerial Vehicles (UAVs) or “drones” have opened up numerous opportunities to monitor and inspect our roads and enhance public safety. In comparison to stationary roadside video cameras, UAVs offer more flexible operation, faster deployment, and wider area coverage. As a result, UAVs are an appealing choice for cities that want to develop smart city solutions through the use of surveillance technologies, traffic monitoring programs, border protection and control, emergency response situations, and large-scale events. Several recent studies have successfully demonstrated the effectiveness of using UAVs to assist in monitoring difficult-to-reach locations, such as deserts, and large public gathering spaces.

Although UAVs have several operational benefits over traditional surveillance methods, they do present some major cybersecurity risks. All communication paths between UAVs, the ground control station, and all back-end services associated with UAVs are at risk of being compromised through various types of attacks. Broad surveys of UAV and flying ad hoc network security catalog eavesdropping, impersonation, and message-tampering threats across these communication links [2,3]. Attack-specific studies further document these risks in centralized UAV-cloud architectures and in commercial drone deployments [4–6], while electronic-warfare-based attacks such as jamming and GPS spoofing represent an additional threat class unique to UAV operating environments [7]. Complementary work on UAV communication security proposes countermeasures addressing many of these attack types [8]. Additionally, the advent of quantum computers represents a potential long-term threat to most cryptographic algorithms now in use in UAV and IOT security systems. Today, most IOT devices utilize classical public key cryptographic algorithms for both secure communication and authentication. Examples of this include RSA and elliptic curve cryptography (ECC) [9]. Both of these algorithms, however, are susceptible to quantum-based attacks enabled by Shor’s algorithms that could potentially break the fundamental mathematical problems upon which these security assumptions were based [10]. Therefore, there is a growing need to transition to post-quantum cryptography (PQC) in order to ensure the continued security of future UAV and internet-of-drones (IOD) networks.

To address this challenge, the National Institute of Standards and Technology (NIST) selected CRYSTALS-Kyber as its primary post-quantum key encapsulation mechanism (KEM) [11]. Kyber provides strong resistance against known quantum attacks while maintaining efficient performance characteristics suitable for resource-constrained platforms. Recent studies have demonstrated its practicality in IoT and embedded environments due to its relatively low computational overhead, implementation efficiency, and suitability for devices with limited processing, memory, and energy resources [12]. These characteristics make Kyber particularly attractive for UAV authentication systems, where lightweight yet quantum-resistant security mechanisms are required.

In addition to cryptographic threats, UAV systems must also defend against physical attacks targeting device identity. A particularly important threat is drone cloning or impersonation, in which an adversary attempts to replicate or masquerade as a legitimate UAV to gain unauthorized access to the system. Traditional software-based credentials cannot fully prevent such attacks because adversaries with physical access may extract, duplicate, or manipulate stored secrets. Physical unclonable functions (PUFs) provide a hardware-rooted security mechanism by exploiting inherent manufacturing variations in integrated circuits to generate unique and practically unclonable challenge–response pairs [13]. As a result, PUFs enable device authentication that is intrinsically tied to the physical hardware itself and have been widely investigated for secure identification and authentication in resource-constrained IoT devices due to their resistance to cloning and low computational overhead. In addition to establishing the legitimacy of UAV devices, secure

communication environments require mutual authentication, whereby both the drone and the authentication server verify each other's identities before exchanging sensitive information. This prevents unauthorized entities from impersonating either party and forms a fundamental security requirement for trustworthy UAV operations.

Another challenge in UAV authentication architectures is based on their dependency upon centralized data storage and credential management services. Centralized infrastructure can be considered a single point of failure by an attacker, and can make it difficult to have complete visibility regarding which security audits are being performed. Distributed storage models, like the InterPlanetary File System (IPFS), store information on multiple peers in a peer-to-peer network using a Content Addressable Network (CAN). This distributed model improves the availability of data and increases the resiliency of the system. Blockchain technology also provides an immutable and tamper-evident record of all authentication events. The use of blockchain and IPFS in the proposed architecture will improve the integrity of stored information, increase auditability, and reduce dependency upon centralized storage infrastructures. However, all of the functions relating to making authentication decisions, generating challenges, and managing keys still reside within the Trusted AERO-GUARD System (AGS), which is currently considered the primary authentication authority.

Research into UAV security has taken significant strides over the years; however, many of the current solutions typically address only a fraction of the challenges present in contemporary UAV deployments. Several approaches provide lightweight authentication, yet rely on traditional cryptographic techniques that would likely fall victim to attacks from a quantum-enabled adversary. Many other methods utilize hardware-based identities, blockchain platforms, or decentralized data storage solutions, but fail to implement all three of these components concurrently: post-quantum secure authentication, private verification, and formally validated protocols. Therefore, there exists an opportunity for a single UAV-security architecture to integrate post-quantum cryptography, hardware-rooted authentication, private verification, decentralized record storage and formally validated protocols together.

Beginning with the need for a secure method for authenticating drones in real-time surveillance systems, this paper develops AERO-GUARD, an authentication and road surveillance framework based on UAV, which combines post-quantum cryptography, hardware rooted identity verification, decentralized storage, and formal security validation. Utilizing CRYSTALS-Kyber's post-quantum primitives—the CPA-secure Kyber.CPAPKE scheme for PUF-response protection—AERO-GUARD provides a basis for potential resistance to any future quantum-enabled adversaries. Additionally, physical unclonable functions (PUFs) are used to establish a hardware-bound drone identity capable of resisting both cloning and impersonation attacks. All authentication records are stored securely within an IPFS-based storage architecture and all records are chained to a blockchain ledger, providing improved integrity, traceability, and auditability. Furthermore, to additionally protect sensitive device identities, AERO-GUARD incorporated privacy-preserving verification utilizing homomorphic encryption on encrypted PUF responses.

The security properties of the authentication protocol proposed by the framework were evaluated through ProVerif 2.05 under the Dolev–Yao advisory model. The advisory model is based on the assumption that the attacker has complete control over all communication between components within the system. As such, the attacker can intercept, modify, replay, or delete any message they wish to send. The analysis of the protocol confirmed that the protocol provides secrecy, mutual authentication, and resistance to replay attacks against an adversary capable of all types of attacks described above. In addition to specifying the security properties of the authentication protocol used by the

framework, we also developed a simulation environment for road surveillance systems to study the operation of AERO-GUARD through UAV-based monitoring and artificial intelligence-based incident detection.

The remainder of this paper is organized as follows. Section 2 reviews the relevant literature. Section 3 presents the system design and implementation. Section 4 describes the proposed drone authentication protocol. Section 5 presents the experimental evaluation and results. Section 6 discusses the findings and limitations. Finally, Section 7 concludes the paper and outlines future research directions.

2. Literature Review

The development of UAVs and IoD has led to an explosion in UAV deployment within the realm of smart cities, border surveillance, disaster relief efforts, environmental monitoring, and large-scale public event support [14,15]. UAVs have gained widespread adoption due to their flexible design, high level of mobility, and ability to provide real-time data collection and transmission.

As the number of UAVs increases, so does the need to secure UAV communication and data exchange. Given that IoD systems primarily utilize wireless communications it is exposed to many different types of cyber threats, including replay attacks, impersonation attacks, eavesdropping, and unauthorized access.

Additionally, the resources available for drone networks are extremely limited, with most drones having very low levels of processor performance, memory capacity, and battery life. Due to the resource constraints inherent to drones, traditional security solutions such as complex cryptographic protocols are generally not feasible for drone use. This makes a lightweight and efficient approach necessary to meet the requirements needed for drone environments.

Researchers have developed several methods to improve UAV security through the development of physical authentication methods known as physical unclonable functions (PUF). PUFs create a unique identity for each device based on manufacturing variations. These PUF-based systems are being utilized to increase UAV authentication capabilities and decrease the potential for cloning and impersonation attacks [16–18].

At the same time, researchers have developed an interest in utilizing post-quantum cryptography (PQC), specifically lattice-based schemes such as CRYSTAL-Kyber. They provide good security against quantum-enabled attackers while maintaining feasibility for use in lightweight devices similar to those found in UAVs [19]. In addition to advancements in cryptography, researchers have utilized blockchain technology as a method of providing decentralized trusted management and secure authentication without reliance on a central authority. Additionally, Inter-Planetary File System (IPFS) was implemented as a decentralized file system that provides improved reliability for data integrity, availability, and resiliency through the distribution of data across numerous nodes instead of depending on a single node [20,21].

Homomorphic encryption is another developing area where private-preserving authentication will protect sensitive UAV identity information during the verification process. Homomorphic encryption allows computations to use encrypted data without divulging underlying values, thus reducing exposure of hardware identifier information such as PUF responses. Homomorphic encryption performs computations directly on encrypted values using operations such as the following:

$$Enc(m1) \oplus Enc(m2) = Enc(m1 + m2) \quad Enc(m1) \otimes Enc(m2) = Enc(m1 \cdot m2)$$

which enables secure verification without exposing plaintext information. However, its utilization in UAV-based authentication systems remains limited [22].

Recent research has demonstrated an increasing number of UAVs being utilized in extreme and large-scale environments, such as deserts and large-scale crowd monitoring at the hajj pilgrimage in Saudi Arabia. Within these types of environments, UAVs are being used for crowd monitoring, emergency response, traffic management, and surveillance in highly dense and safety-critical environments. While demonstrating significant operational value, they further highlight the need for authenticated UAV authentication frameworks that can operate reliably and securely, and which are scalable under extreme environments [14,15].

While current research addresses individual elements related to UAV security, such as authentication, cryptography, decentralized storage, and privacy preservation, there remains a gap in the integration of these technologies into a single, lightweight, quantum-resistant security framework applicable in real-world UAV deployments. The AERO-GUARD project seeks to close this gap through the integration of PUF-based authentication, post-quantum cryptography, blockchain, IPFS, and privacy-preserving mechanisms into one single secure IoD architecture.

2.1. IoD Security and Authentication Protocols

The growth of the Internet of Drones has created new cybersecurity issues. This is mainly due to an increase in drones being used in smart cities, at borders, for surveillance, in emergency and disaster situations, and in military missions. Since UAVs continually send and receive data through radio-frequency communication links, there is high potential for UAVs to be subjected to many types of cyberattacks, including but not limited to replay attacks, impersonation attacks, eavesdropping and unauthorized access. Because of their dynamic, resource-limited and potentially hostile operating environment, many established security solutions cannot effectively protect the data sent from UAVs. As a result, researchers have been working to develop lightweight authentication and secure communication protocols, especially for UAV environments. Oláh et al. developed a secure registration mechanism for IoD systems using blockchain technology combined with physical unclonable function (PUF). They enhanced trust management by providing hardware-based identity verification and anti-cloning features. Nevertheless, Oláh's solution still employed classic cryptography and had no means of defense against possible future quantum-based threats [16].

Park et al. also developed a mutually authenticated key agreement protocol that was proven to be able to be securely used in IoT systems. While Park's lightweight authentication method protected UAV communications from replay attacks, it did not include a decentralized data storage feature or post-quantum-resistant cryptographic protection [17].

Authentication is critical in IoD systems since it verifies which UAVs may operate on a given network. The majority of current authentication methodologies depend upon symmetric encryption, elliptic curve cryptography (ECC) and a challenge-response methodology to create secure communication sessions between UAVs and their control systems. In addition, mutual authentication is important because both parties must authenticate one another prior to sending or receiving confidential information.

Zhang et al. proposed PRLAP-IoD, a PUF-based lightweight authentication protocol for resource-constrained UAVs. Zhang's methodology minimized the computational overhead associated with authentication processes while ensuring the authenticity of all exchanged messages and session keys. Although Zhang's methodology provided robustness against replay attacks and ensured the secrecy of session keys, it failed to provide mechanisms to support distributed trust relationships or private comparisons of PUF responses [18].

Choi et al. proposed a secure PUF-based authentication and key agreement protocol suitable for IoD applications. Choi's methodology demonstrated resilience to replay attacks and impersonation attacks and provided higher reliability than previous authentication frameworks for UAV communications. However, despite these enhancements, Choi's methodology continued to assume that classical cryptographic assumptions could protect UAV communications; therefore, although Choi's methodology provided better protection than many of its predecessors, it was not completely formally verified to possess injective authentication property, which ensures that every successful authentication session corresponds to a unique communication instance between legitimate parties and cannot be mistakenly reused or duplicated [20].

Research continues to concentrate on lightweight authentication schemes for IoD devices. Tang et al. presented PUF-AKA-IoD (Tang et al.), a provable, secure, PUF-based authentication and key agreement protocol for IoD systems running in 5G networks. PUF-AKA-IoD provided an improvement in efficiency of the mutual authentication process, and also reduced communication overhead for UAVs. Nevertheless, PUF-AKA-IoD relied heavily upon a central server for authentication and had no ability to store identities using decentralized data stores such as IPFS [23].

Xie and Zhang also developed a lightweight authentication protocol for drone-to-ground station and drone-to-drone authentication. Xie and Zhang's scheme offered better privacy protections for IoD communications, including forward secrecy. However, their authentication process did require assistance from ground stations during the UAV authentication phase. In addition, Xie and Zhang's scheme made use of conventional cryptographic assumptions, which are vulnerable to quantum computer attacks, whereas this work focused on developing a post-quantum approach to security [24].

Al-Mekhlafi et al. used Chebyshev polynomials to develop an authentication mechanism for IoD devices (ChebIoD). ChebIoD also included mechanisms for updating keys dynamically, as well as revoking them when necessary. The authors were able to demonstrate that ChebIoD is formally secure and can efficiently generate session keys. Although Al-Mekhlafi et al. did include an analysis of how identities could be stored securely on the device itself, they did not consider how identities could be stored in a way that preserves user privacy when comparing them via a PUF [25].

2.2. PUF-Based Post-Quantum Authentication for UAV Security

Quantum algorithms (e.g., Shor's algorithm) may potentially disrupt classical encryption methodologies by rapidly solving the factorization and discrete logarithm problems. As a result, researchers have been increasing their focus on developing post-quantum cryptography (PQC), a methodology intended to provide cryptographic protection against adversaries enabled with quantum computers.

CRYSTAL-Kyber is one of the NIST-approved, lattice-based, post-quantum cryptographic algorithms. It provides robust defenses based on the hardness of the Module Learning With Errors (M-LWE) problem, while providing sufficient processing speed for utilization in resource-constrained devices like drones. CRYSTAL-Kyber provides an ideal trade-off between security and computational efficiency, making it an attractive option for securing future UAV communications in various IoD platforms.

Several recent studies have researched hybrid security frameworks that integrate PUF-based authentication methods with post-quantum cryptography. Nair et al. proposed a hybrid authentication framework incorporating Kyber with PUF-based authentication to develop a quantum-resistant UAV authentication platform. The framework was able to demonstrate enhanced resistance against future cryptographic threats; however, it

lacked decentralized data storage capabilities, homomorphic comparison capabilities, and comprehensive formal security validation [19].

Researchers in recent years have been studying more robust post-quantum authentication methods that can be used in IOD (Internet of Drones) systems. For example, Tawfeeq et al. designed a new lightweight quantum-resistant authentication method to protect IoDs operating within military environments utilizing dilithium signatures. This solution was developed to provide greater security against potential quantum attacks, yet still be computationally lightweight enough to run on UAVs. However, this design failed to implement decentralized trust management or blockchain technology [26] as part of its framework.

Lastly, Xie and Wang presented a PUF-based anonymous authentication protocol for use in cross-domain UAV environments. This protocol provided increased levels of UAV anonymity and authentication efficiency. Moreover, the protocol included mechanisms to increase resistance to impersonation attacks. However, despite the above-mentioned improvements, the protocol did not address decentralized trust management via blockchain or homomorphic encryption techniques [27].

2.3. Blockchain and Decentralized Storage in IoD

Blockchain provides a viable way to establish the required levels of decentralized trust and authentication in Internet of Drones systems. Centralized architectures do not allow for decentralized verifications, immutable records, and transparently managed authentication; thus, they present additional risk when compared to decentralized authority models and the presence of single points of failure (SPoF).

Wazid et al. developed a framework based upon blockchain that includes authentication and key management for IoT drone applications. The new framework provided improved methods for managing decentralized trust and increased resistance against replay attacks in UAVs. However, this framework failed to include other hardware-rooted identity technologies, including PUFs, or decentralized file storage technologies, including IPFS [22].

The InterPlanetary File System (IPFS) represents a distributed decentralized storage system that uses Content Addressable Storage (CAS) to store and retrieve files from multiple nodes. IPFS does not rely on centralized server storage; instead files stored via IPFS are identified by their Content Identifiers (CID). CAS allows for Content Identifiers to be used to identify and locate content, thereby improving data integrity, availability and redundancy. CAS also presents lower risk than centralized data storage systems because of its ability to provide better protection against data manipulation and failure.

Recent studies have begun exploring the use of blockchain technology combined with IPFS as a method to develop secure decentralized identity management systems. These blockchain/IPFSs work so that blockchain can be utilized to securely store immutable identifiers such as CIDs while off-chain data are distributed throughout the network by utilizing IPFS nodes. Using both blockchain and IPFS together provides the benefit of reduced blockchain storage overhead and scalable solutions while continuing to maintain the benefits of decentralized trust management. Unfortunately most current IoD-based authentication systems continue to require the use of centralized enrollment databases that are subject to unauthorized access and SPoF [28].

Recent research in blockchain technology has looked at how they could be used to develop improved lightweight decentralized authentication systems (LDASs) for Unmanned Aerial Vehicles (UAVs). Hossain et al. developed SoftPUF, a software-based blockchain authentication framework that uses machine learning combined with PUF-inspired identification mechanisms to enable scalable blockchain authentication without specialized

hardware requirements. However, SoftPUF provided more of an improvement to general blockchain systems than secure Internet of Drones (IoD) deployment environments [29].

2.4. Privacy-Preserving Authentication Techniques

Researchers have been exploring homomorphic encryption techniques, which enable computations to occur over encrypted data without the need to decrypt the data, thereby allowing secure authentication and protecting the confidential nature of sensitive information.

Existing drone authentication protocols typically compare PUF responses in plaintext form during their verification phase, therefore disclosing hardware identities. Therefore, researchers have developed homomorphic computation methods so that PUF-based authentication systems may utilize encrypted computation while maintaining the ability to securely perform comparison operations [30].

Frameworks have recently been proposed for privacy-preserving authentication based upon encrypted computation techniques to support comparisons of sensitive values without divulging plaintext information. While this type of method provides significant improvements in authentication privacy, there has been very little research into directly applying homomorphic encryption to PUF-based Internet of Drones authentication systems [31].

Advanced authentication techniques have been researched by recent studies in order to be used in future quantum networks. Hybrid authentication protocols were developed by Goswami et al., which integrate physical uncloneable functions (PUFs) along with the properties of a quantum network to provide an authenticated exchange with a low communication cost. Goswami's model provided good theoretical security. However, his model is based on quantum communications systems and it was not modified to fit the use case of lightweight Unmanned Aerial Vehicle (UAV) deployments [32].

Mefgouda et al. LPUF-AuthNet was proposed as a scalable and machine learning-resistant lightweight PUF (physical unclonable function)-based authentication framework utilizing split learning with tandem neural networks. The authors of this work have demonstrated that their solution can be used to enable low-power authentication in the Internet Of Things (IoT). They do note, however, that their design is not optimized for UAV-based IoT devices (UAV-IoTDs) or post-quantum secure communication environments [33].

Roberto Román et al. have developed a quantum-resistant authentication mechanism for IoT applications using homomorphic encryption combined with weak physical unclonable functions (W-PUFs), which does not require additional helper data. This approach enables the performance of authentication operations on encrypted W-PUF responses. Therefore, it prevents the exposure of sensitive hardware identity information in the verification process. Additionally, this method provides increased protection against quantum-enabled attacks and reduces the storage overhead caused by helper data management. In addition to the robust privacy preservation and quantum-safe authentication features of the proposed scheme, the framework is mainly intended for general IoT environments and has not been optimized for lightweight UAV deployments, decentralized trust mechanisms or large-scale Internet of Drones systems [31].

2.5. UAV Applications in Desert and Hajj Environments

UAVs are becoming increasingly common in harsh and large-scale operational environments such as deserts, border surveillance areas, and large public gatherings, including Hajj and Umrah. In Saudi Arabia, drones are currently under consideration for various applications, such as crowd monitoring, traffic observation, emergency response, environmental surveillance, hazard detection, and security operations. The above applications clearly

indicate that UAVs have become more important than ever before in terms of providing support for public safety, infrastructure monitoring, and smart city initiatives.

As indicated previously, desert environments pose significant operational challenges for the successful deployment of UAVs. These include but are not limited to extreme temperature ranges, dust storm conditions, unstable weather conditions, and long communication distance requirements. Such conditions may negatively impact the performance of a UAV as well as the reliability of the communications required between a UAV and its command center.

Similarly, Hajj operations require highly reliable monitoring and fast emergency response systems because millions of pilgrims gather in densely populated areas. Therefore, drones are now viewed as very useful tools for aerial surveillance and crowd management in addition to disaster responses at large-scale events.

Recent work has been conducted on utilizing drones along with computer vision techniques for monitoring and managing crowded spaces during Hajj operations. One recent study presented a drone-based system for instantaneous crowd-counting and vehicle detection during Hajj operations using computer vision techniques in order to enhance surveillance and monitoring efficiency [14]. A second study utilized hybrid ai classifiers to detect abnormal behavior in large crowds of pilgrims during Hajj operations using spatiotemporal analysis and computer vision techniques [15]. Both of these studies demonstrated the potential of intelligent UAV-assisted surveillance systems to enhance both public safety and emergency response.

Although these works have provided some insight into the potential benefits of intelligent UAV-assisted surveillance systems for enhancing both public safety and emergency response, there are still several limitations associated with existing UAV deployment frameworks. Most existing UAV deployment frameworks mainly focus on enhancing the operational efficiency of UAV systems, as well as their ability to monitor crowds and other areas. However, most existing UAV deployment frameworks do little, if anything, to address issues related to secure authentication/authorization, decentralized trust management, privacy-preserving authentication, and post-quantum security protection. Since drones operating within critical environments often transmit sensitive surveillance data both to and from multiple network entities, there exists a high risk of compromised UAV systems creating serious security and public safety risks.

Therefore, it becomes apparent that what is needed are lightweight yet robust, secure, and quantum-resistant authentication architectures specifically developed for the deployment of UAV systems within sensitive environments like deserts and Hajj operations.

Table 1 presents an overview of all major contributions and findings from the present state of the art in this area. It also presents both the advantages and disadvantages of the current methods for the authentication of IoT devices, post-quantum cryptographic protection, the integration of blockchain and IPFS, and privacy-preserving techniques. Additionally, it clearly indicates the gap in the knowledge, which is a motivation for developing the proposed AERO-GUARD architecture.

Table 1. Gap analysis.

Reference	Focus	Key Contribution	Limitation
[14] Hajj Drone Monitoring (Counting and Detection)	UAV surveillance in Hajj environments	Uses drones for real-time crowd and vehicle monitoring using computer vision	Focuses on surveillance only; lacks security, authentication, and privacy mechanisms
[15] Hybrid AI Crowd Behavior Detection (Hajj)	AI-based abnormal behavior detection in crowds	Detects abnormal crowd behavior using spatio-temporal AI models for safety during Hajj	Does not address UAV security, authentication, or communication protection

Table 1. Cont.

Reference	Focus	Key Contribution	Limitation
[16] Oláh et al.	Blockchain + PUF-based UAV registration	Introduced secure registration using blockchain and PUFs for UAV identity management, improving anti-cloning and trust establishment	Relies on classical cryptography and does not provide post-quantum security or protection against future quantum attacks
[17] Park et al.	Mutual authentication and key agreement for IoD	Provides provably secure lightweight authentication with resistance to replay attacks in UAV communication	Does not include decentralized storage or post-quantum cryptographic protection
[18] Zhang et al.	Lightweight PUF-based authentication protocol	Reduces computational overhead and ensures secure session key establishment for resource-constrained UAVs	Lacks decentralized trust management and privacy-preserving comparison of PUF responses
[19] Nair et al.	PUF-based cross-domain authentication with PQC	Integrates PUFs with post-quantum cryptography to improve resistance against quantum threats	Does not include privacy-preserving authentication or decentralized storage mechanisms
[20] Choi et al.	PUF-based authentication and key agreement	Enhances replay and impersonation attack resistance with efficient PUF-based authentication	Relies on classical cryptographic assumptions and lacks full formal verification
[21] Algarni et al.	Robust IoD authentication protocol	Provides improved security analysis and robustness for UAV authentication systems	Does not integrate PUF-based identity or decentralized storage technologies
[22] Wazid et al.	Blockchain-based authentication and key management	Enables decentralized trust management and improves resistance to replay attacks	Does not use hardware-based identity (PUF) or IPFS-based decentralized storage
[23] Tang et al.	PUF-based authentication over 5G IoD	Improved lightweight mutual authentication and communication efficiency	Still relies on centralized authentication architecture
[24] Xie and Zhang	Drone-to-drone authentication	Enhanced privacy protection and forward secrecy in IoD communication	Did not integrate post-quantum or decentralized technologies
[25] Al-Mekhlafi et al.	Chebyshev-based lightweight authentication	Added dynamic key update and revocation mechanisms with formal verification	Lacked IPFS integration and privacy-preserving authentication
[26] Tawfeeq et al.	Quantum-resistant UAV authentication	Applied dilithium-based post-quantum authentication for military IoD	Did not integrate blockchain or decentralized trust management
[27] Xie and Wang	Anonymous PUF-based authentication	Improved UAV anonymity and impersonation resistance	Lacked blockchain-based trust and privacy-preserving comparison
[29] Hossain et al.	Blockchain + machine learning authentication	Proposed scalable blockchain authentication using SoftPUF techniques	Not specifically optimized for UAV or IoD environments
[31] Román et al.	Quantum-safe privacy-preserving authentication using homomorphic encryption and PUFs	Proposed a quantum-safe authentication scheme that performs authentication on encrypted PUF responses without helper data, improving privacy protection and resistance against quantum attacks	Designed mainly for general IoT systems and does not address lightweight UAV optimization, decentralized trust management, blockchain/IPFS integration, or large-scale IoD deployment environments
[32] Goswami et al.	Hybrid quantum authentication	Proposed secure authentication for advanced quantum networks	Not adapted for lightweight UAV deployments
[33] Mefgouda et al.	Lightweight PUF-IoT authentication	Improved resistance against ML-based attacks using split learning	Not designed for UAV systems or post-quantum environments

3. System Design and Implementation

3.1. System Name Overview

AERO-GUARD: A post-quantum mutual authentication drone protocol with homomorphic encryption for secure road surveillance in smart cities. The purpose of the project is to create a secure monitoring and authentication framework for Unmanned Aerial Vehicles (UAVs) that will be used for road safety in smart city environments. The word AERO reflects the drone's role of aerial monitoring, and GUARD relates to the protection of UAV identity, communication, mission data, and operational records.

3.2. System Architecture

AERO-GUARD is a secure UAV-based road monitoring system supported by a web dashboard and a security framework. It aims to support road security authorities by enhancing traffic surveillance using UAVs to monitor road environments, detect abnormal events, and report incidents through a management dashboard. The developed framework protects UAV operations from cybersecurity threats such as unauthorized access, GPS-spoofing, signal hijacking, future quantum-based attacks, replay attacks, and data tampering. The system combines PUF-based authentication, IPFS, blockchain, homomorphic encryption, and post-quantum cryptography.

The architecture consists of the following main components:

- **UAV:** This is responsible for monitoring roads and collecting road-related data, and reporting the detected incidents, such as car crashes or abnormal road events. Each drone has a hardware-based identity through a physical unclonable function (PUF); it generates a unique response that is intended to resist reproduction or cloning under the assumed PUF model (see Section 6.4).
- **AERO-GUARD System:** This acts as the main control and verification entity. It manages drone registration, authentication requests, challenge generation, session verification, and authentication decisions. It also communicates with IPFS, blockchain, and the off-chain evaluator to verify drone identity securely.
- **IPFS storage:** IPFS is used to store encrypted authentication-related records, such as encrypted PUF responses and authentication results. Instead of storing large data directly in the blockchain, the system stores it in IPFS and records the returned content identifier (CID) for verification.
- **Permissioned Blockchain (PBC):** The blockchain stores immutable references to IPFS records. This provides auditability and prevents the silent modification of enrollment and authentication records. AERO-GUARD adopts a permissioned, consortium-operated ledger (e.g., a hyperledger fabric-style deployment) run by a known set of authorized ground-station and regulatory nodes, using a crash/Byzantine-fault-tolerant consensus protocol such as PBFT or Raft, which better suits the low-latency, high-throughput requirements of IoD authentication logging than a permissionless public chain. Each transaction anchors only the IPFS content identifier (CID) and outcome metadata—never raw PUF data—and write access is restricted to authenticated AGS instances via smart-contract access control. IPFS record persistence is maintained by dedicated pinning nodes operated by the consortium members.
- **Off-Chain Evaluator (OCE):** The OCE performs privacy-preserving comparisons on encrypted PUF responses through homomorphic operations without accessing the plaintext PUF responses.
- **Management Dashboard:** This provides a user interface for monitoring UAV, viewing incidents, and reviewing system logs. It includes a web-based simulation system with a WebGL scene, camera controls, motion controls, and live recording controls.

- **3D Simulation Environment:** The prototype includes a WebGL-based simulation environment. The application creates the WebGL renderer, scene, fog, lights, GUI controls, statistics panel, and animation loop.

The system architecture, therefore, is divided into two connected parts: the drone Authentication protocol, which protects drone identity and communication, and the dashboard interface, which illustrates the operational road-monitoring scenario.

Figure 1 shows the relationship between AERO-GUARD components and how they interact with each other:

1. The process starts at the ground control station, where the operator assigns a patrol mission to an authorized drone.
2. Then the authorized drone performs real-time surveillance operations over the assigned road area. During the mission, the drone captures road scene data and monitors for abnormal events.
3. Using onboard AI models, the drone analyzes the captured data and detects events such as accidents or anomalies.
4. When an incident is detected, the drone securely sends the detected event back to the control station. The AERO-GUARD system then verifies the data source to ensure authenticity, integrity, and trust in reported events.
5. Finally, all verified mission data, detected events, authentication results, and operator actions are logged and stored for auditing, monitoring and future analysis.



Figure 1. An overview of the AERO-GUARD system workflow.

3.3. Dashboard Interface and Operational Scenarios

The AERO-GUARD dashboard was designed to support road security operators in monitoring UAV activities and responding to detected incidents.

In the implemented prototype, the dashboard includes camera controls, motion controls, and live recording features. The simulation environment also contains real-environment assets such as vehicles, lights and trees, which support the realistic visualization of road-monitoring scenarios. In addition, the dashboard is integrated with a deep-learning-based AI vision component that analyzes frames extracted from the drone video feed. The AI component supports the detection of road-safety events such as car accidents, abnormal speeding cars, and normal road conditions.

Figure 2 illustrates the AERO-GUARD dashboard, which enables operators to monitor drone activities, review detected incidents, visualize the road environment, and observe authentication events in real time.

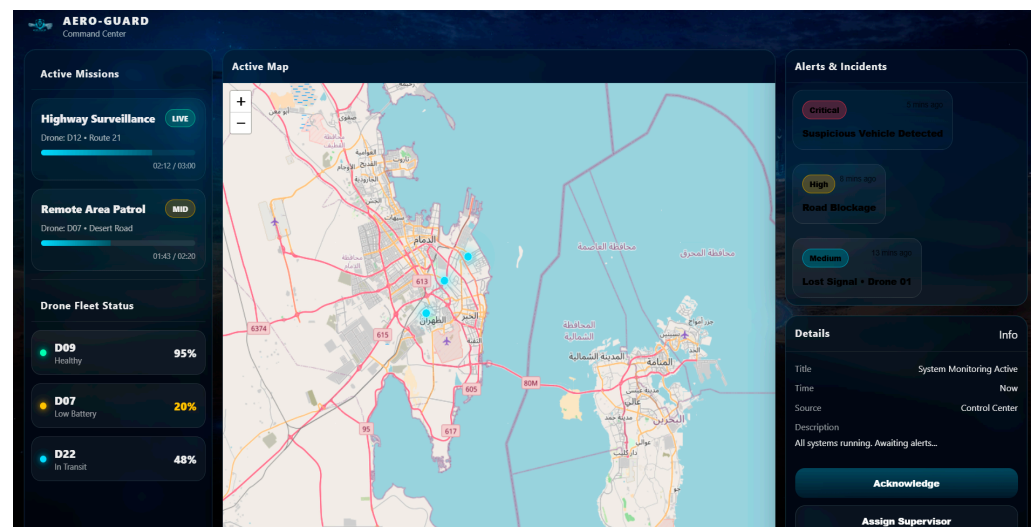


Figure 2. AERO-GUARD dashboard.

Figure 3 shows the drone monitoring a road accident involving vehicles in the observed area. The event is considered an abnormal road-safety incident and reported to the management dashboard for review and response.



Figure 3. Car accident event.

Figure 4 shows the drone monitoring a clear road in the observed area. The event is considered a normal situation, and no alert was reported to the management dashboard.

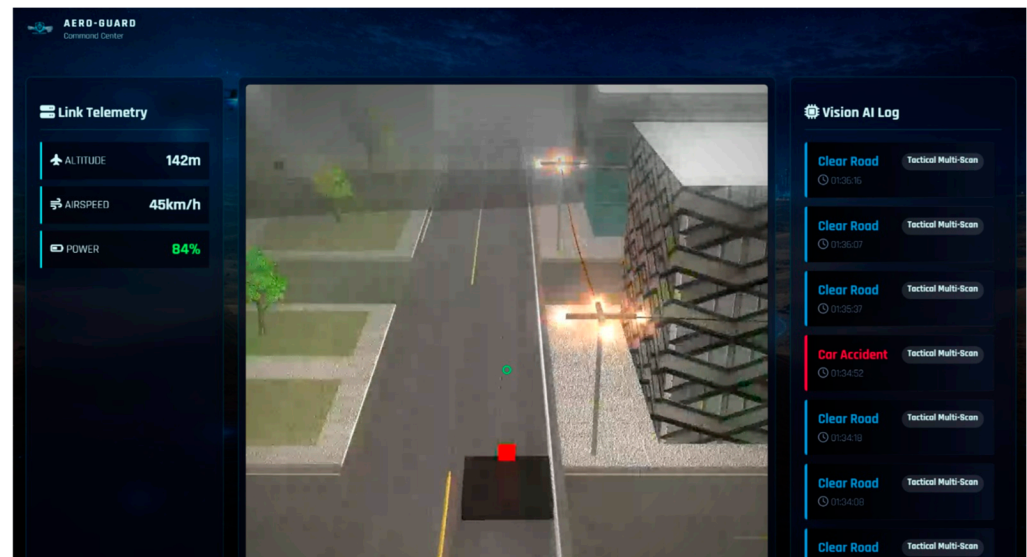


Figure 4. Clear road.

3.4. Security Mechanisms

AERO-GUARD employs a multi-layer security approach to protect UAV authentication, communication, and incident reporting. The main security mechanisms are as follows:

- **PUF-Based Drone Identity:** Each drone uses a physical unclonable function as a hardware-based identity. The PUF secret is never transmitted and remains physically embedded in the drone hardware.
- **Post-Quantum Cryptography:** The Kyber.CPAPKE cryptographic mechanism is used to support post-quantum security. In the provisioning phase, the AERO-GUARD system keeps the Kyber.CPAPKE private key securely, while the drone receives the Kyber.CPAPKE public key and long-term master key through a secure channel.
- **HMAC, Nonces, Timestamps:** The authentication protocol uses HMAC values, nonces, and timestamps to protect message integrity and prevent replay attacks.
- **Homomorphic Evaluation:** The off-chain evaluator performs comparisons between stored PUF responses and fresh PUF responses using homomorphic operations on encrypted data without decrypting the PUF responses and without holding the private key.
- **Blockchain and IPFS Logging:** The system stores an encrypted record in the IPFS and records the CID in the blockchain, which provides integrity, auditability, and tamper-resistance for enrollment and authentication results.
- **Dashboard Access and Logging:** The dashboard supports controlled access to monitoring information and incident reports.

Together, these security mechanisms protect the UAV reporting process from impersonation, data tampering, unauthorized access, replay attacks, and exposure of sensitive authentication data.

3.5. Simulation and Verification

AERO-GUARD uses simulation and formal verification to evaluate both the operational and security aspects of the system. CARLA was used for simulation because it provides strong support for autonomous driving search, high-fidelity traffic scenarios, diverse weather conditions, and an AI-based perception system. This makes it more suitable for smart city and road operation use cases. For security verification, the system uses ProVerif to analyze the authentication protocol. The verification focuses on confidentiality, integrity, authentication, and replay protection.

4. Proposed Drone Authentication Protocol

This section presents the proposed AERO-GUARD drone authentication protocol, which consists of three phases: the provisioning phase, enrollment phase, and authentication phase. The main entities involved in the proposed protocol are summarized in Table 2.

Table 2. Protocol entities.

Entity	Description
AGS	AERO-GUARD System
Drone	Drone
IPFS	InterPlanetary File System (decentralized storage)
PBC	Permissioned Blockchain
OCE	Off-Chain Evaluator

AERO-GUARD compares encrypted PUF responses using the additive-homomorphic property of the Kyber CPA-secure public-key encryption primitive, *Kyber.CPAPKE*, following the construction validated by Román et al. [31]. This is distinct from the FIPS 203-standardized ML-KEM key-encapsulation interface, whose Fujisaki–Okamoto transform removes the algebraic structure this comparison relies on. *Kyber.CPAPKE* is used specifically for PUF-response protection. The three operations used are as follows:

$$\begin{aligned}
 R &= \text{Kyber.CPAPKE.Enc}(r, PK) \\
 \text{Poly.Sub}(R1, R0) &= \text{Kyber.CPAPKE.Enc}(r1, PK) - \text{Kyber.CPAPKE.Enc}(r0, PK) \\
 &= \text{Kyber.CPAPKE.Enc}(r1 \text{ XOR } r0, PK) \\
 HD(r1, r0) &= HW(\text{Kyber.CPAPKE.Dec}(\text{Poly.Sub}(R1, R0), SK))
 \end{aligned}$$

where PK/SK are the AGS's *Kyber.CPAPKE* key pair and *HW*(·) is the Hamming weight. This construction assumes the CPA-security of *Kyber.CPAPKE* and that the OCE and AGS do not collude and behave honestly but curiously.

4.1. Provisioning Phase

The initial provisioning process occurs in a safe factory or manufacturing area once it establishes the credentials for the long-term communication sessions between each time the drone enrolls in and authenticates the AERO-GUARD system. The shared values used during the provisioning phase are summarized in Table 3.

Table 3. Provisioning phase shared values.

Credential	Description
skGS	Kyber.CPAPKE private key held by AGS
pk(skGS)	Kyber.CPAPKE public key
Ks_long	Pre-shared long-term master key
puf_secret	Drone hardware identity

AGS stores its own Kyber private key skGS within an HSM. At no time does the AGS expose the raw key material. The AGS delivers the public key generated from skGS, as well as a pre-established master key Ks_long, to the drone over an unbreachable, factory-to-drone private network. On the drone side, these two values are then stored in either a secure element (SE) or a trusted execution environment (TEE).

Important security note: The drone's PUF secret (puf_secret), which represents the physical characteristics of the drone hardware determined by manufacturing variances in silicon, is never transmitted. It can only be accessed through a read-only method utilizing the PUF_D(·) function, and is intended to resist cloning, extraction, or reproduction by

another device under the assumed PUF model. The provisioning phase of the proposed protocol is illustrated in Figure 5.

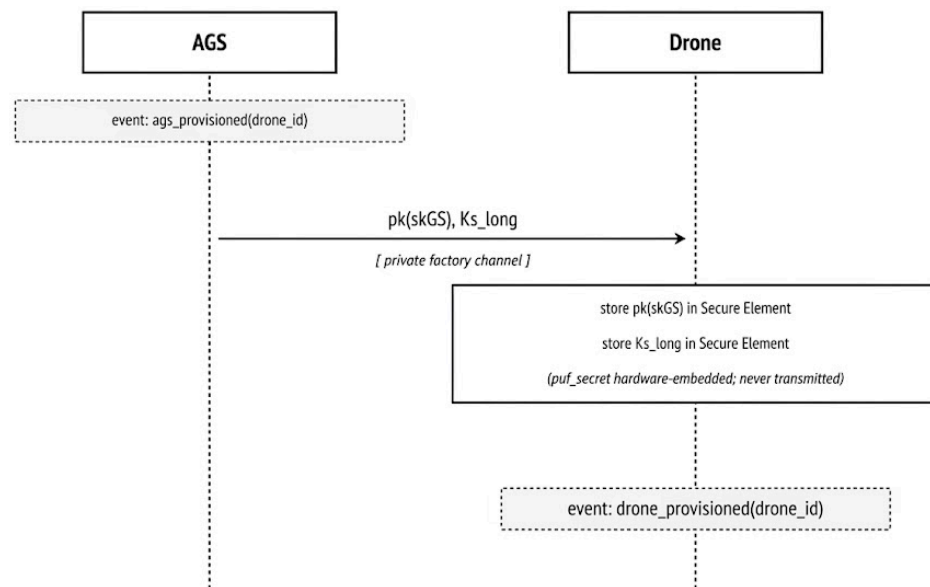


Figure 5. Workflow of the provisioning phase.

4.2. Enrollment Phase

The enrollment phase securely registers the drone within the AERO-GUARD system by generating and storing a trusted hardware-rooted identity using PUF-based authentication and decentralized storage. The shared values used during the Enrollment phase are summarized in Table 4.

Table 4. Enrollment shared values.

Credential	Description
ID	Drone ID
Reg_request	Registration request token (randomly generated)
pk(skGS)	AGS Kyber.CPAPKE public key (received during provisioning)
C	Challenge (fresh random value generated by AGS)
r0	PUF response
R0	Encrypted PUF response
CID	Content identifier (returned by IPFS)

This phase starts when the drone sends the identifier ID along with a new registration request token to the AERO-GUARD system (AGS). The AGS receives the identifiers, creates a new random challenge C and sends it back to the drone along with the public part of the AGS' secret key Ppir pk(skGS), although it was previously sent to the drone in the provisioning phase to facilitate stateless operation and key refresh scenarios.

Upon receiving challenge C from the AGS, the drone uses C as an input for its physical unclonable function (PUF), obtaining a response r0 that is exclusive to this particular drone and cannot be replicated by any other device due to silicon-level manufacturing variations.

$$r0 = PUF(C)$$

After this, the drone encrypts r0 with the AGS's Kyber public key, obtaining Ciphertext R0 and sending it to the AGS.

$$R0 = Kyber.CPAPKE.Enc(r0, pk(skGS))$$

The AGS stores the Tuple (ID, R0, C) in IPFS. In turn, IPFS returns a content identifier (CID) that is a hash-based pointer to the data stored. As such, the AGS stores both the Tuples (ID, CID) on the permissioned blockchain. In this way, the permissioned blockchain acts as an immutable index whereas all the larger data objects are stored off-chain in IPFS.

Lastly, the AGS responds to the drone, confirming that enrollment was successful by returning the CID. The enrollment phase of the proposed protocol is illustrated in Figure 6.

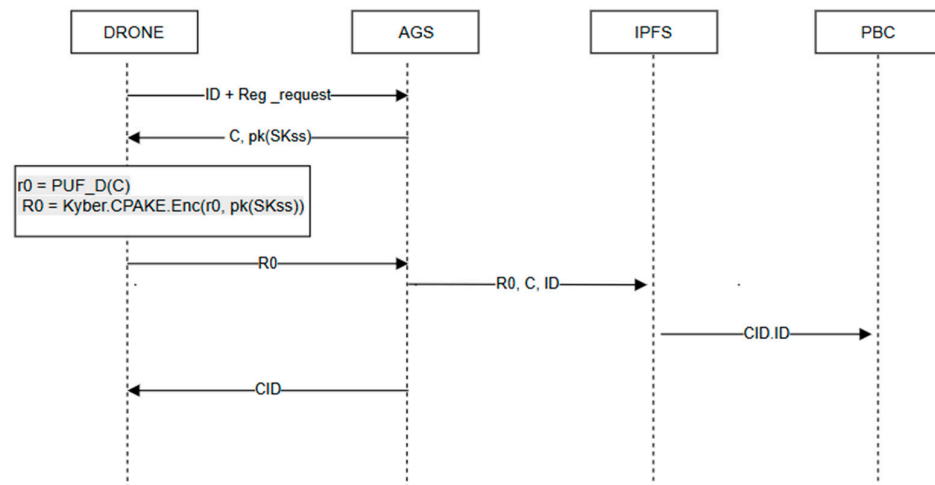


Figure 6. Workflow of the enrollment phase.

4.3. Authentication Phase

The initial authentication stage will be initiated at the time that a drone initiates a request to use the system. The purpose of this stage is to authenticate the identity of the drone as well as protect the integrity of the messages sent by the drone; it also protects against an attacker that is able to intercept or duplicate messages intended for another party through replay or parallel sessions. The shared values used during the authentication phase are summarized in Table 5.

Table 5. Authentication-phase shared values.

Credential	Description
ID	Drone ID
CID	Content identifier (from blockchain)
Ks_long	Pre-shared long-term master key (from provisioning)
K_session	Session key
Ts, Ts2, Ts3	Timestamps
ΔT	Maximum allowed clock-skew window
nonce_D	Fresh random nonce generated by drone per session
HMAC-SHA256()	Hash-based message authentication code
C	Challenge retrieved from IPFS
r0, R0	Enrolled reference PUF response (plaintext/ciphertext)
r1, R1	Fresh PUF response for this session (plaintext/ciphertext)
Di	Homomorphic difference
Di	Decrypted difference
HD	Hamming distance
HD_max	Authentication threshold

The authentication starts with a single initiation message composed of a merged set of data items. The drone creates a new random number (nonce_D) and adds a timestamp (Ts) to create a session key. Following RFC 5869, HKDF is instantiated as HKDF-SHA256 with the long-term shared secret Ks_long as the input keying material (IKM), a fixed protocol-specific string “AERO-GUARD-v1” as the salt, and $Ts||nonce_D||ID$ as the info/context

parameter binding the derived key to this specific session; the output length is 256 bits, matching the HMAC-SHA256 key size used below:

$$\begin{aligned} K_{\text{session}} &= \text{HKDF-SHA256}(\text{salt} = \text{"AERO-GUARD-v1"}, \\ &\quad \text{IKM} = K_{\text{s_long}}, \\ &\quad \text{info} = \text{Ts} || \text{nonce_D} || \text{ID}, \\ &\quad L = 256). \end{aligned}$$

Then, the drone immediately calculates

$$\text{HMAC-SHA256}(K_{\text{session}}, \text{ID} || \text{Ts} || \text{nonce_D})$$

where HMAC-SHA256 follows FIPS 198-1, keyed with the 256-bit K_{session} derived above. Unless noted otherwise, HMAC denotes HMAC-SHA256 throughout the remainder of this protocol description.

Upon receiving the initiation message, the AGS independently derives the same K_{session} using the received Ts, nonce_D, and ID. It verifies both the timestamp freshness check ($|\text{current_time} - \text{Ts}| \leq \Delta T$) and the HMAC. The timestamp check enforces replay protection by rejecting messages outside the allowed clock-skew window ΔT . If both checks pass, the AGS retrieves the enrolled data: it queries the blockchain using the drone ID to obtain the CID, then fetches the stored reference response R0 and challenge C from IPFS.

The AGS sends challenge C back to the drone, along with a fresh timestamp Ts2 and $\text{HMAC-SHA256}(K_{\text{session}}, C || \text{Ts2} || \text{ID} || \text{nonce_D})$. This message simultaneously delivers the challenge for PUF evaluation and proves to the drone that the AGS possesses the correct session key K_{session} , achieving mutual authentication.

The drone verifies the AGS timestamp and HMAC, then evaluates its PUF using challenge C to produce fresh response r1. It encrypts r1 under $\text{pk}(\text{skGS})$ to obtain $R1 = \text{Kyber.CPAPKE.Enc}(r1, \text{pk}(\text{skGS}))$. The drone's response HMAC critically includes Ts2: $\text{HMAC-SHA256}(K_{\text{session}}, R1 || \text{Ts3} || \text{Ts2} || \text{ID} || \text{nonce_D})$. Including the AGS challenge timestamp Ts2 binds the drone's response to the specific AGS instance that issued that challenge. Without Ts2, an attacker could forward a single valid response message to two parallel AGS sessions that happen to share the same K_{session} , breaking injective authentication. When Ts2 is included, those two sessions have different Ts2 values and the HMAC would fail on all but the intended session.

After verifying the drone's HMAC and the timestamp Ts3, the AGS forwards both ciphertexts R1 and R0 to the off-chain evaluator (OCE) over a private channel. Following the Poly.Sub construction defined for Kyber.CPAPKE ciphertexts by Román et al. [31], the OCE performs a coefficient-wise ciphertext subtraction.

$$\begin{aligned} D_i &= \text{Poly.Sub}(R1, R0) \\ &= \text{Kyber.CPAPKE.Enc}(r1, \text{pk}(\text{skGS})) - \text{Kyber.CPAPKE.Enc}(r0, \text{pk}(\text{skGS})) \\ &= \text{Kyber.CPAPKE.Enc}(r1 \text{ XOR } r0, \text{pk}(\text{skGS})) \end{aligned}$$

entirely on ciphertexts, without ever decrypting them; the underlying additive-homomorphic property holds for Kyber.CPAPKE specifically, not for the FIPS 203 ML-KEM key-encapsulation interface. The OCE never holds the private key skGS and cannot learn the plaintext PUF responses. The encrypted difference D_i is returned to the AGS, which decrypts it using Kyber.CPAPKE.Dec with skGS to obtain the plaintext difference d_i . The Hamming weight of d_i equals the Hamming distance HD between the stored reference $r0$ and the fresh response $r1$. If $\text{HD} \leq \text{HD}_{\text{max}}$, the drone is considered authentic. Otherwise, the authentication request is rejected. Subtracting two Kyber.CPAPKE ciphertexts approximately doubles the decryption-noise variance relative to a single ciphertext; Kyber's

parameters retain a comfortable margin against decryption failure under this operation, as demonstrated empirically by Román et al. [31] for 512-bit PUF responses. AERO-GUARD adopts the same response size and security level for consistency; validating this margin against AERO-GUARD's own protocol parameters on target hardware is identified as a direction for future work. In both cases, the result is stored in IPFS and the corresponding CID is recorded on the blockchain, maintaining an immutable and auditable record of every authentication outcome. The authentication phase of the proposed protocol is illustrated in Figure 7.

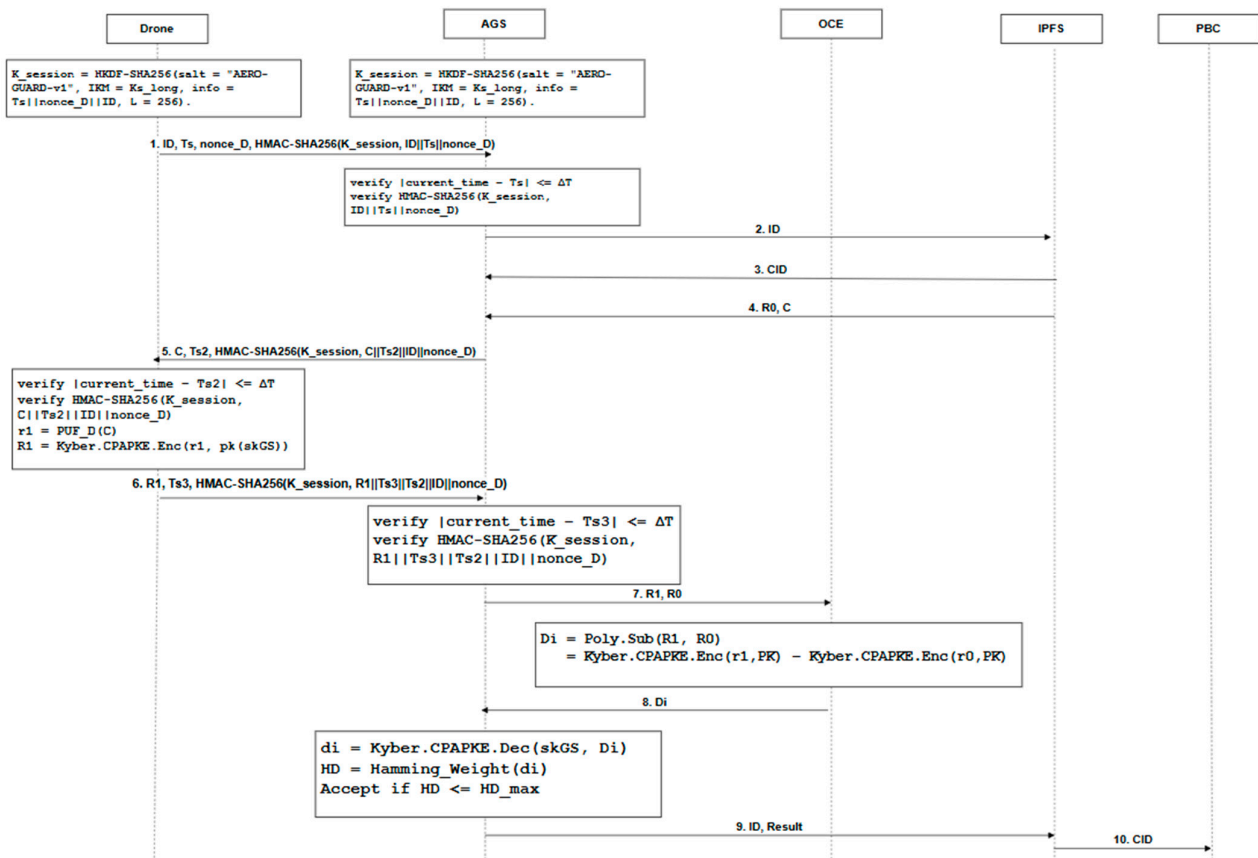


Figure 7. Workflow of the authentication phase.

4.4. Drone Authentication Protocol Summary

The proposed AERO-GUARD drone authentication protocol is a combination of post-quantum cryptography and hardware-rooted identity validation, with decentralized data stores and a blockchain-based auditing system, which will create an environment for UAVs that are both secure and resistant to failure in the authentication process. In addition to providing mutual authentication through its three phases (provisioning/enrollment/authentication), the protocol also protects against replay attacks and has privacy-preserving PUF verification as well as providing tamper-resistant identification of all entities participating in the process. This integration of Kyber encryption and PUF technologies, along with IPFS, blockchain and homomorphic operations, provides AERO-GUARD with a scalable, formally proven security architecture applicable to smart cities and autonomous drone applications.

5. Results

5.1. ProVerif Findings

Verification of the security mechanisms in the AERO-GUARD protocol using ProVerif was conducted focusing on three aspects: the secrecy of long-term credentials, entity authentication, and replay protection. First, the confidentiality of the three core secrets—the pre-shared master key Ks_long , the AGS Kyber.CPAPKE private key $skGS$, and the drone’s PUF hardware secret puf_secret —was verified to ensure that no information leaks to an external attacker throughout the protocol lifecycle. In detail, Ks_long is provisioned exclusively over a private factory channel $c_provision$ that is invisible to the Dolev–Yao attacker and is never placed on the public channel in any process branch. The session key exposed on the public channel is always a one-way HKDF derivation of Ks_long , leaving no algebraic path for recovery. ProVerif approved the secrecy of all three credentials, confirming that queries S1, S2, and S3 are satisfied under the full Dolev–Yao attacker model.

Second, the authentication mechanism was verified by modeling the HMAC-based challenge-response exchange and the PUF verification step in ProVerif. The non-injective correspondence query A1 confirmed that whenever the ground station fires the acceptance event gs_accept_auth , the legitimate drone must have genuinely initiated the session with a matching identity, nonce, and timestamp—ruling out impersonation entirely. The result proved that the HMAC chain effectively ensures valid drone identity, preventing unauthorized access by any party that does not hold Ks_long .

Third, replay protection was verified through the injective correspondence query A2, which is a strictly stronger property than entity authentication alone. This query confirmed that every ground station acceptance maps to a distinct, previously unmatched drone initiation. The protection is enforced by two cooperating mechanisms: a fresh nonce $nonce_D$ generated at the start of every session, and the $within_window()$ timestamp destructor that rejects any message carrying a previously seen timestamp. ProVerif verified that replaying any recorded valid session message will always fail this check, and that gs_accept_auth can never fire twice for the same $drone_begin_auth$ event.

Finally, the provisioning integrity query A3 confirmed that the factory key-loading phase is sequentially ordered in every trace—the AGS always completes its provisioning before the drone can receive any credentials. Enrollment authentication was intentionally left out of scope, as the enrollment phase is designed to operate in a physically trusted environment. Importantly, even if an attacker were to enroll a fraudulent entry, the puf_verify destructor at step 10 of the authentication phase would reject any response that does not match the genuine PUF output, ensuring that the authentication guarantees remain intact regardless of enrollment integrity. The verification questions and results are compiled in Table 6. Under the symbolic model and the explicitly modeled assumptions, ProVerif verifies the specified secrecy and correspondence queries, as indicated by the statement “Satisfied = True”.

Table 6. AERO GUARD Protocol ProVerif formal verification results.

Property Tested	Mechanism	Query	Result
Secrecy—Pre-shared Master Key Confidentiality	HKDF-derived session keys; long-term key provisioned over private channel and stored in secure element.	S1	True
Secrecy—AGS Kyber Private Key	Kyber private key remains inside HSM/TPM and is never transmitted.	S2	True
Secrecy—Drone PUF Secret	PUF secret is hardware-embedded and only used locally.	S3	True
Authentication—Impersonation Resistance	HMAC-bound session key prevents forgery.	A1	True
Authentication—Replay Protection	Fresh nonce and timestamp validation.	A2	True

Table 6. Cont.

Property Tested	Mechanism	Query	Result
Integrity—Key Provisioning	Provisioning over private channel with ordered events.	A3	True
Authentication—PUF Response Authenticity	PUF verification succeeds only for valid response.	Step 10	True
Integrity—OCE Cannot Decrypt PUF Responses	OCE cannot access secret keys or decrypt PUF responses.	Step 8–9	True
Integrity—Challenge Binding	Challenge cryptographically bound to session.	Step 5–7	True
Integrity—Enrollment Data Integrity	Enrollment phase outside authentication scope.	A4	N/A

The assertion that AERO-GUARD is secure-by-design is reinforced by these quantitative verification results, which show that it is not only built with cryptographically sound components Kyber.CPAPKE asymmetric encryption, HKDF-SHA256 session key derivation, HMAC-SHA256 message authentication, and hardware PUF binding but is also formally and mechanically verified against an adversary with full control over the public communication channel.

5.1.1. ProVerif Verification Results

Table 6 summarizes the formal verification results gained using ProVerif 2.05. All specified secrecy and correspondence queries were verified under the symbolic Dolev–Yao adversary model. Due to the successful verification of secrecy queries, the confidential values, including the long-term master key and session key, remain inaccessible to an attacker with full control of the communication channel. The queries show that mutual authentication between the drone and the AERO-GUARD system (AGS) is preserved, which ensures that successful authentication events occur only between legitimate participants. Moreover, the verification results confirm the protocol’s resistance to replay attacks using timestamps, nonces, and authenticated message exchanges. These results provide formal evidence that the protocol satisfies the modeled authentication and confidentiality properties under the stated assumptions.

5.1.2. Interpretation of Verification Results

Figure 8 demonstrates the output generated by ProVerif verification tool after analyzing the proposed authentication protocol. The figure complements Table 6 by providing evidence of the actual verification process and confirms that the protocol satisfies the security properties under the symbolic attacker model.

```

Verification summary:

Query not attacker(Ks_long[]) is true.
Query not attacker(skGS[]) is true.
Query not attacker(puf_secret[]) is true.

Query event(gs_accept_auth(id1_3,n,ts)) ==> event(drone_begin_auth(id1_3,n,ts))
is true.
Query inj-event(gs_accept_auth(id1_3,n,ts)) ==> inj-event(drone_begin(id1_3,n,ts))
is true.
Query event(drone_provisioned(id1_3)) ==> event(ags_provisioned(id1_3)) is true.

```

Figure 8. ProVerif confirms it satisfies key security properties and prevents attacker access to sensitive data.

5.1.3. Security Assumptions and Scope of Verification

The security guarantees demonstrated in this section are established under the symbolic Delov–Yao adversary model implemented in ProVerif 2.05. This model assumes that the adversary has full control over the communication channel and can intercept, modify, replay, and inject messages while cryptographic primitives are treated as ideal. Under these assumptions, the formal analysis verifies the specified secrecy and correspondence properties of the proposed protocol.

An important note is that ProVerif does not establish computational security, physical clonability of PUF devices, blockchain integrity, IPFS availability, AI detection correctness, or resistance to implementation-specific attacks such as timing, power analysis, or other side-channel attacks. These properties depend on the correctness of the underlying implementations and deployment environment and are therefore outside the scope of symbolic formal verification.

5.2. Simulation Results

The system was tested by streaming a live drone video feed directly to the Flask server through a browser-based client. As soon as recording began, the server started saving the incoming video, and a background process woke up automatically to monitor it. Every 5 s, this process opened the video file, jumped to the most recent frames, and extracted the latest available image. That image was then converted and sent to the AI model for analysis. The whole process ran continuously in the background without any manual input, keeping the dashboard updated with fresh results throughout the session.

The simulation confirmed that the system remained stable across different scenarios. When a frame could not be read, for example, while the video file was still being written, the system logged a warning and moved on to the next cycle without any interruption. API errors were also handled quietly, printing the issue without stopping the scanner. When recording was stopped by the operator, the background scanner paused immediately, and the dashboard updated to show a standby status. The system proved reliable from start to finish, with clean transitions between active monitoring and idle states.

5.3. AI-Based Accident Assessment

The AI-based accident assessment described below is presented as a prototype demonstration of system integration and dashboard behavior, not as a scientifically evaluated detector: we do not report the underlying model name, training dataset, validation split, or standard detection metrics (accuracy, precision, recall, F1-score, confusion matrix, inference time, or false-alarm rate over a defined test-scenario set). A rigorous quantitative evaluation of the accident-detection component, including these metrics, is left as future work (Section 6.9).

Every 5 s, the latest frame from the drone feed is sent to an AI vision model along with a short instruction. The AI is said to act as a road safety monitor and to look for specific events only: a car accident, a single car, or a clear road. By keeping the instruction focused and simple, the AI always returns a short, clean response that the system can read and act on immediately, with no extra processing needed.

The AI response is used to trigger a color-coded alert on the operator's dashboard. If a single car is spotted, a blue indicator is shown to signal normal traffic activity. If any type of accident is detected, whether involving one car or multiple cars, the alert turns red immediately. Red always overrides blue, so the most critical situation is always the one displayed to the operator.

Once the AI produces a result, it is pushed to the dashboard in real time through a live WebSocket connection. The alert appears instantly on the screen, showing both the

type of event detected and the matching color indicator. This means the operator does not need to watch the raw video feed at all; the system highlights what matters and delivers it in a simple, readable format.

The AI assessment works for continuous road monitoring, but there are a couple of straightforward ways it could be improved. Since the system checks only once every 5 s, a very brief event like a sudden swerve might happen between scans and go undetected. Scanning more frequently would reduce this gap. Another improvement would be to require the same hazard to appear in two or more consecutive frames before triggering a red alert, which would help avoid false alarms caused by a single unclear or blurry frame.

5.4. Authentication Protocol Performance Analysis

This section reports an estimated latency budget for the AERO-GUARD post-quantum mutual authentication protocol, not a measurement taken on a physical implementation. The per-primitive timing figures are drawn from published benchmarks of CRYSTALS-Kyber, HMAC-SHA256, PUF-based authentication, and IPFS-based storage on comparable embedded platforms, and are combined analytically to project the cost of AERO-GUARD's three phases: provisioning, enrollment, and authentication. Section 6.8 (Limitations) discusses this distinction and identifies hardware-based measurement as a required next step. The analysis considers four aspects: end-to-end latency per phase, additional communication overhead relative to prior systems, scalability with drone fleet size, and the ability of PUFs to discriminate genuine from cloned hardware identities.

5.4.1. End-to-End Latency per Protocol Phase

Figure 9 presents the estimated latency budget for each protocol phase, obtained by aggregating the literature-reported timings for the underlying cryptographic and networking primitives rather than by measuring the proposed implementation directly (Section 6.8). Phase 0 (provisioning) involves only Kyber.CPAPKE key generation and HKDF-SHA256 derivation, with an estimated total of about 13 ms; it is executed once at factory time over a private channel. Phase 1 (enrollment) adds IPFS storage and blockchain logging, raising the estimated cost to approximately 96 ms; like provisioning, this occurs only once during the drone's operational life. Phase 2 (authentication) is estimated at approximately 85 ms, which remains within the commonly cited 200 ms real-time threshold for interactive authentication reported by Aissaoui et al. [34] and Xia et al. [35]. The homomorphic subtraction step performed by the OCE contributes an estimated 6.3 ms, indicating that the added privacy-preserving comparison does not substantially increase the overall budget. These estimates are broadly consistent with those of Aissaoui et al. [34] and Xia et al. [35], who report Kyber-768 operation times of 12–19 ms on ARM Cortex-class processors representative of drone controllers. We emphasize that these are projected, not measured, values; Section 6.8 outlines the hardware benchmarking needed to validate them.

5.4.2. PUF Hamming Distance Distribution

Figure 10 illustrates the discriminative power of the PUF-based hardware binding mechanism. Genuine re-evaluations of the same PUF challenge produce Hamming distances in the range [2,8] (mean ~4.1 bits) due to minor silicon-level noise—well below the $HD_{max} = 10$ threshold adopted in AERO-GUARD's authentication phase. An attacker attempting to clone or emulate the PUF—whether through physical extraction or modeling attacks—produces responses whose Hamming distance from the enrolled reference clusters is around 64 bits (~50% of a 128-bit response), reflecting the near-random behavior of a non-matching PUF. The separation between the two distributions is approximately 5.3 standard deviations, yielding a theoretical false acceptance rate (FAR) below 10^{-6} and a false rejection rate (FRR) below 0.3%. These characteristics indicate that the PUF layer

is intended to provide a hardware-bound identity that resists bypass via credential compromise alone under the assumed PUF model and simulated noise parameters reported here; as noted in Sections 6.4 and 6.8, this claim has not been validated against physical silicon or implementation-level (e.g., side-channel) attacks and remains a direction for future work. The analysis assumes a 128-bit-response silicon PUF (e.g., an SRAM- or Arbiter-based construction) paired with a fuzzy-extractor/error-correcting code (e.g., a BCH-style code) capable of correcting up to $HD_{max} = 10$ bit-flips, consistent with the reliability characteristics reported in the literature and the ProVerif query S3 result reported in Table 6.

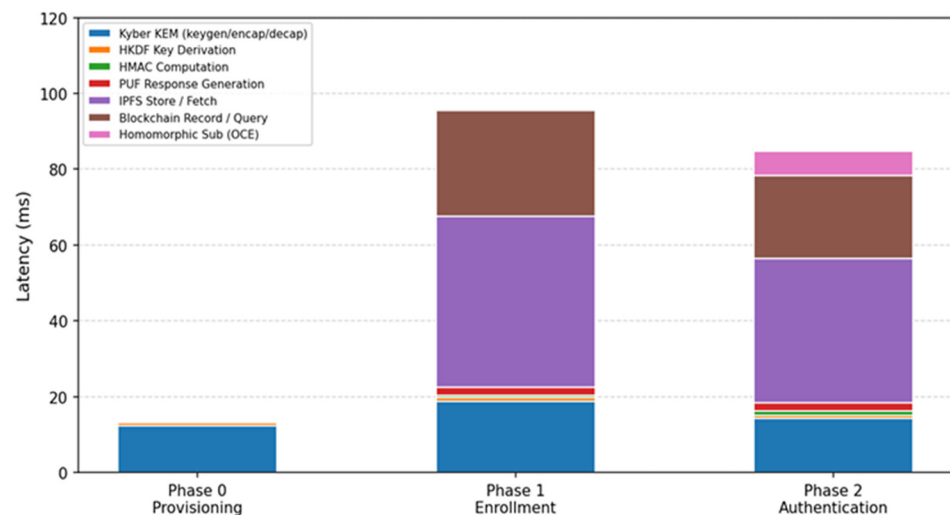


Figure 9. AERO-GUARD protocol latency breakdown per phase. Stacked latency breakdown of each AERO-GUARD protocol phase. Cryptographic primitives (Kyber.CPAPKE, HKDF-SHA256, HMAC-SHA256) contribute less than 20 ms; the dominant overhead in Phases 1 and 2 is network I/O to IPFS and the permissioned blockchain. Phase 0 (provisioning) is executed once at factory time and does not affect runtime performance.

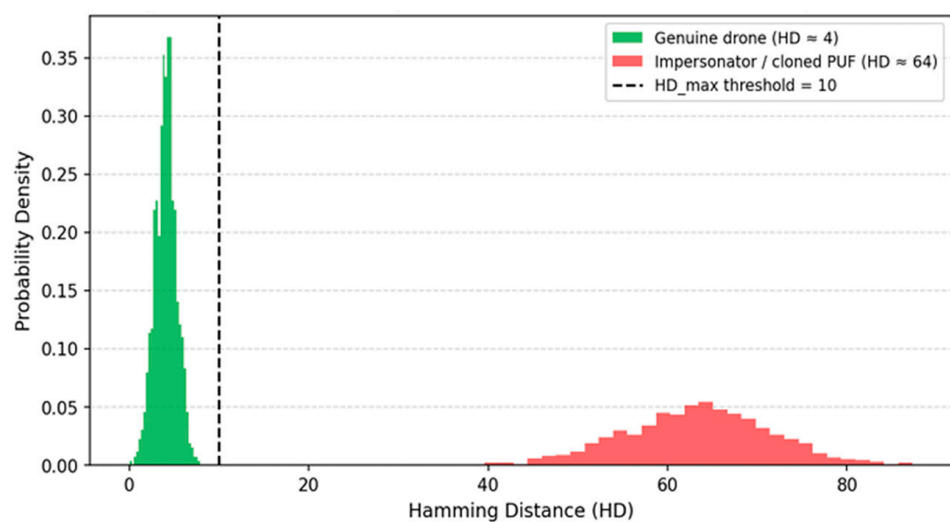


Figure 10. PUF Hamming distance distribution. Simulated Hamming distance (HD) distribution for genuine drone PUF responses (green) versus cloned or impersonator responses (red). The dashed vertical line marks the $HD_{max} = 10$ acceptance threshold; genuine responses cluster tightly below it while impersonators fall far above.

6. Discussion

6.1. Overview: A Unified Multi-Layer Security Framework

This research demonstrates that AERO-GUARD effectively incorporates multiple state-of-the-art security technologies to create a single integrated system for UAV-based surveillance and authenticating (i.e., validating) UAVs operating in smart city environments supporting road operations. As opposed to most existing Internet-of-Dots (IoD) security frameworks that provide protection against only a single area of attack (blockchain-based logging; post-quantum cryptography), AERO-GUARD is the first framework that provides all five areas of protection together: post-quantum security; hardware-rooted authentication; decentralized storage; privacy preserving verification; and formalized protocol validation.

Figure 11 illustrates how AERO-GUARD compares against representative prior works across five critical security properties. While individual systems check one or two boxes, AERO-GUARD is the only framework to satisfy all five simultaneously.



Figure 11. Security property comparison: AERO-GUARD vs. selected prior IoD authentication frameworks: Wazid et al. [22], Choi et al. [20], Nair et al. [19], Zhang et al. [18], Park et al. [17], and Oláh et al. [16].

6.2. Formal Security Verification

6.2.1. ProVerif Results and Their Significance

The ProVerif analysis confirmed that the proposed authentication protocol meets many significant security requirements as follows: the secrecy of long-term credentials, replay resistance, mutual authentication, and the integrity of the provision process. Figure 12 presents a summary of all nine verified properties, grouped by category.

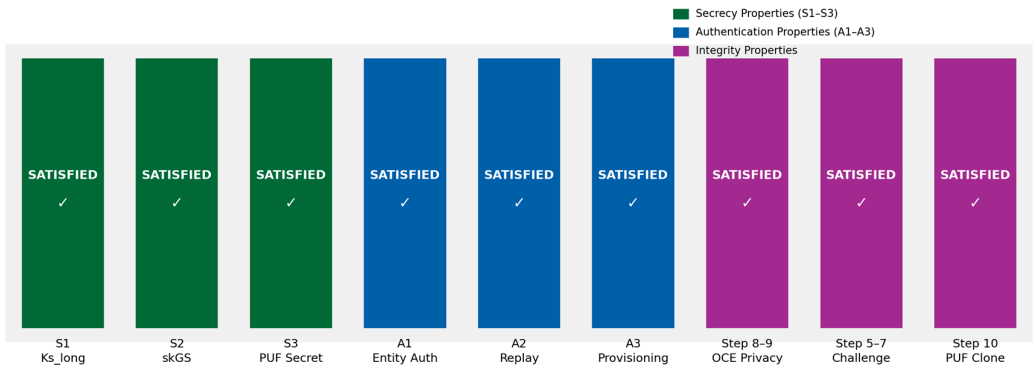


Figure 12. ProVerif formal verification results for AERO-GUARD. All nine security properties are satisfied under the Dolev–Yao adversary model.

These findings are significant for several reasons. Many existing UAV authentication frameworks rely on traditional cryptographic assumptions—RSA, ECC, or Diffie–Hellman—that may become vulnerable as quantum computing matures. By formally verifying the secrecy of Ks_long (S1), $skGS$ (S2), and puf_secret (S3) under a full Dolev–Yao attacker, AERO-GUARD provides machine-checked guarantees rather than informal security arguments.

The injective correspondence query A2, which is a strictly stronger property than non-injective entity authentication, confirms that every ground station acceptance event maps to a distinct and unmatched drone initiation—ruling out replay attacks even in the presence of a network-level adversary who can record, delay, and replay arbitrary messages.

6.2.2. Comparison with Formally Verified Prior Work

Among the systems reviewed in the literature, only Park et al. [8] and Al-Mekhlafi et al. [22] provide formal security proofs, and neither addresses post-quantum security or privacy-preserving PUF verification. AERO-GUARD extends formal verification to a richer set of properties, including provisioning integrity (A3) and OCE privacy (Steps 8–9), which have not been modeled in prior IoD verification efforts.

6.3. Post-Quantum Resilience

AERO-GUARD builds on the CRYSTALS-Kyber post-quantum family, standardized in part by NIST as ML-KEM, to provide resistance to adversaries using Shor’s algorithm (the basis for quantum computing) when an adversary uses a sufficiently powerful quantum computer. Most existing IOD authentication schemes employ ECC or RSA key mechanisms, which will become compromised if a sufficiently large quantum computer exists.

AERO-GUARD uses two distinct Kyber-based building blocks. The CPA-secure public-key encryption component $Kyber.CPAPKE$ —not the FIPS 203-standardized ML-KEM key-encapsulation interface, whose Fujisaki–Okamoto transform removes the algebraic structure relied on here—is used to encrypt PUF responses ($R0 = Kyber.CPAPKE.Enc(r0, pk(skGS))$) during enrollment and authentication, following the homomorphic-comparison construction of Román et al. [31]; the corresponding $Kyber.CPAPKE.Dec$ is used by the AGS to recover the plaintext Hamming difference di . This dual use maximizes the quantum-safety coverage of the most sensitive cryptographic operations in the protocol. This construction relies on the CPA-security of $Kyber.CPAPKE$ and on the assumption that the OCE and AGS do not collude and behave honestly but curiously, consistent with the assumptions adopted in [22].

Importantly, the Kyber private key $skGS$ is provisioned exclusively within a hardware security module (HSM) at the AGS and is never exposed on any network channel—a design decision confirmed by the ProVerif S2 secrecy query.

6.4. Hardware-Rooted Identity via Physical Unclonable Functions

The use of PUFs based on hardware identities provides increased security with respect to counterfeiting and spoofing attacks from drones. In previous research, it was shown that hardware UAVs can extract credentials stored in flash memory or secure elements via invasive techniques, for example, bus probing or side channel analysis, because credentials are only stored in the software.

The puf_secret (puf_secret) never leaves the drone. It is embedded into the chip and is inseparable from the physical PUF $PUF_D(\cdot)$; evaluating $PUF_D(C)$ returns a challenge-specific response without requiring or exposing the puf_secret as a separate, extractable parameter. This is proven by the S3 query made with ProVerif. Since the responses generated by a PUF arise from random nanoscale changes due to manufacturing processes, these responses are intended to resist copying or cloning onto another drone under the

assumed PUF model, making counterfeit UAV production difficult even if an attacker knows the schematic and/or firmware; as noted above, this claim has not been validated against silicon-level attacks and is a direction for future work.

6.5. Decentralized Storage and Blockchain-Based Auditability

Another significant aspect of AERO-GUARD is the integration of blockchain and IPFS technologies to provide a decentralized method of storing identities and providing tamper-evident audit logs. Centralized authentication systems have numerous documented problems; specifically, single points of failure (e.g., servers can be taken down), honeypot-like opportunities for credential exfiltration, and very obscure audit trails which are often impossible to verify independently.

The two-tier storage approach employed within AERO-GUARD solves these issues with respect to centralized authentication. Authentication records (encrypted with PUF-generated keys), consisting of the PUF ciphertexts and authentication outcomes, are stored in IPFS utilizing Content Addressable Hashes (CIDs). These CIDs, rather than the raw data, are then linked to an immutable public blockchain.

This reduces the amount of data being stored on-chain, as well as preserving both evidence of tampering, as any changes made to the contents of an IPFS record will result in a new CID and therefore invalidate the corresponding blockchain entry, and the auditing capabilities of a blockchain.

6.6. Privacy-Preserving PUF Verification

The privacy-preserving authentication mechanisms in AERO-GUARD represent improvements to many previous UAV PUF-based authentication mechanisms. Most of these were based upon comparing PUF response data in the same plain-text format during the verification process. This increases the possibility of PUF hardware identifiers leaking, as it would be possible for the authentication server to obtain access to these identifiers if they were compromised.

AERO-GUARD's off-chain evaluator (OCE) will perform homomorphic subtraction operations on the two ciphertexts R1 and R0 without obtaining decryption keys or being able to decrypt either of them. The OCE processes only ciphertexts; therefore, the plaintext difference d_i will be obtained only at the AGS once the homomorphic subtraction is completed. Also, the ProVerif analysis of steps 8–9 shows that this method provides protection so that even if the OCE is totally compromised, the plaintext values of PUF responses still cannot be determined.

The Hamming distance threshold HD_{max} provides noise tolerance to protect against the inherent inter-session variability of PUF responses due to temperature, voltage and aging effects. This noise tolerance is only applied after decryption at the authentication gateway server (AGS), thus ensuring that any comparison of the threshold values never exposes plaintext information to any intermediate party.

6.7. Operational Simulation and AI-Based Incident Detection

From an operational perspective, the simulation environment demonstrates that the system can continuously support real-time road surveillance and AI-assisted incident detection. The dashboard successfully displays alerts for events such as accidents and road hazards while maintaining stable operation through continuous video-streaming. The AI-based assessment model easily reduced complex video feeds to clear operator alerts, improving the response efficiency of road monitoring authorities.

6.8. Limitations

Although the above-mentioned results are promising, there are a number of factors which need to be considered in order to properly assess the present state of the system.

First, currently, the prototype is based upon a completely simulated environment and has not yet been deployed in actual UAV hardware. A real-world test would allow for an assessment of the latency associated with communication over wireless channels, energy usage for both Kyber and HMAC operations by embedded processors, the reliability of wireless links when subjected to environmental noise, and how adverse weather (e.g., sandstorms; torrential rains) affects drone telemetry and video quality.

Second, since the AI polls every five seconds, if a very short event lasts less than five seconds, it will likely go undetected in any sampled frame. An increase in the rate at which the AI polls the camera feed and/or the development of a continuously polled frame-difference detection mechanism could eliminate this time window, but at the expense of increased processing requirements.

Third, although the protocol was formally verified using the Dolev–Yao attacker model, this model is limited because it assumes a perfectly secure cryptography and does not take into account potential implementation-level security weaknesses (i.e., timing side channels; power analysis attacks; fault injections). Future work should include a side-channel analysis of the PUF readout and Kyber operations on the target hardware platform.

Fourth, the existing AI model uses each frame individually as opposed to using frames temporally. If we require a high-priority alert only after the same hazard has been detected in two or more consecutive frames, this will lower the number of false positives due to momentary image distortions.

6.9. Future Directions

There are also several potential directions for future research: enhancing cryptography, increasing the scalability of the overall system, and deploying the system operationally.

In terms of enhancing cryptography, there are two possible avenues of investigation. One avenue of investigation would be to develop hybrid post-quantum algorithms that combine Kyber with traditional ECC. These algorithms can then be deployed in transitional environments where some nodes still use classical ECC while others deploy Kyber. Another area of investigation would be to expand formal verification to include the IPFS and blockchain interactions that were outside the scope of the ProVerif model used in this paper. In addition to expanding formal verification, an additional direction for evaluating side-channel resistance in implementations of the PUF readout and Kyber.CPAPKE decryption functions is needed. The goal here is to evaluate these functions on representative embedded hardware.

With respect to increasing scalability, there are at least four different ways to increase scalability. First, one direction for investigation involves developing lightweight blockchain consensus mechanisms that are optimized for swarms of drones operating in parallel. This includes a full implementation and benchmarking of the PBFT/Raft consensus layer described in Section 3, a concrete smart-contract code for CID-anchoring transactions, a formal fault-tolerance analysis of the consortium's node count, and a production pinning-service strategy for long-term IPFS availability. Second, another direction for investigation involves developing federated learning approaches to enable distributed UAV intelligence. Federated learning preserves the privacy guarantees provided by the homomorphic evaluation framework. Third, one way to increase scalability is through adaptive trust management systems that are able to adjust HD_max thresholds dynamically depending upon what conditions in their environment cause variations in the PUF responses. Finally, one method

for increasing scalability is through the development of new AI models that are better suited for use in swarms of drones.

Finally, there is much value in deploying the AERO-GUARD framework at a large scale in smart cities, for example, during Hajj and other major events. Such deployments would allow researchers to assess how scalable the AERO-GUARD framework is when deployed at scales of thousands of drones, and how well the AI detection model performs when detecting hazards in highly dynamic crowds and traffic.

7. Conclusions

This paper presented an implementation of AERO-GUARD as an integrated authentication and surveillance framework for use in UAVs in smart cities. The proposed architecture integrates all components, such as PUFs, post-quantum cryptography through CRYSTAL KIBER, blockchain technology, privacy-preserving homomorphism evaluations, and decentralized data storage using IPFS, for the first time.

The formal analysis of the protocol was performed with ProVerif in the context of the Dolev–Yao adversary model, where it is shown that all the important characteristics of AERO-GUARD (the secrecy of critical credentials, replay protection, mutual authentication, provisioning integrity, defense against impersonation and cloning attacks) are fulfilled. Furthermore, an integration test environment demonstrated the practical feasibility of AERO-GUARD by means of UAV monitoring in real-time, AI-based anomaly detection and reporting incidents via a dashboard.

In comparison to the existing authentication frameworks in the Internet of Drones (IoD), AERO-GUARD solves several of their well-known limitations. For example, compared to previous solutions, AERO-GUARD integrates decentralized record anchoring, hardware-rooted identity verification, post-quantum security, and privacy-preserving authentication into one lightweight solution, which can be deployed on UAVs. The AGS remains the central authority for challenge generation, key management, and authentication decisions; the permissioned blockchain and IPFS decentralize the storage and tamper-evidence of authentication records, not the authentication decision itself. The use of blockchain and IPFS also increases auditability and tampering resistance. The homomorphic comparison mechanism protects sensitive PUF-responses during the verification process.

The overall outcome indicates that AERO-GUARD represents a promising basis for safe and scalable UAV deployment in a variety of smart-city applications, such as road-monitoring, border-surveillance, emergency-response and large-scale public-event management. The next steps will include real UAV deployments, optimizations of system performance, improvements in AI-based detection capabilities, and evaluations of scalability in large operational environments.

Author Contributions: Conceptualization, A.A. (Albandari Alsumayt); methodology, R.A. (Reema Almassary); software, H.A., R.A. (Reemas Alqahtani), R.A. (Ryouf Alzuabie) and R.A. (Reham Alharthi); validation, A.A. (Albandari Alsumayt) and A.A. (Arwa Almalki); formal analysis, R.A. (Reema Almassary), H.A., R.A. (Reemas Alqahtani), R.A. (Ryouf Alzuabie) and R.A. (Reham Alharthi); investigation, R.A. (Reema Almassary) and H.A.; resources, A.A. (Albandari Alsumayt) and A.A. (Arwa Almalki); data curation, R.A. (Reema Almassary), H.A., R.A. (Reemas Alqahtani), R.A. (Ryouf Alzuabie) and R.A. (Reham Alharthi); writing—original draft preparation, R.A. (Reema Almassary) and H.A.; writing—review and editing, A.A. (Albandari Alsumayt), A.A. (Arwa Almalki) and N.N.; visualization, H.A., R.A. (Reemas Alqahtani), R.A. (Ryouf Alzuabie) and R.A. (Reham Alharthi); supervision, A.A. (Albandari Alsumayt) and A.A. (Arwa Almalki); project administration, A.A. (Albandari Alsumayt). All authors have read and agreed to the published version of the manuscript.

Funding: We would like to thank the Saudi ARAMCO Cybersecurity Chair for funding this project.

Data Availability Statement: This study did not generate or analyze new datasets. The anomaly detection component uses a pretrained deep learning model; no original training data were produced as part of this work.

Conflicts of Interest: The authors declare no competing interests.

References

1. World Health Organization. Road Traffic Injuries. *WHO Fact Sheet*. Available online: <https://www.who.int/news-room/fact-sheets/detail/road-traffic-injuries> (accessed on 20 July 2026).
2. Tsao, K.-Y.; Girdler, T.; Vassilakis, V.G. A Survey of Cyber Security Threats and Solutions for UAV Communications and Flying Ad-Hoc Networks. *Ad Hoc Netw.* **2022**, *133*, 102894. [\[CrossRef\]](#)
3. Yaacoub, J.P.A.; Noura, H.; Salman, O.; Chehab, A. Security Analysis of Drones Systems: Attacks, Limitations, and Recommendations. *Internet Things* **2020**, *11*, 100218. [\[CrossRef\]](#) [\[PubMed\]](#)
4. Aldosari, F.; Alharbi, A.; Alosaimi, W.; Alyami, H. Cyber Attacks on Unmanned Aerial Vehicles and Cyber Security Measures. *Sensors* **2022**, *22*, 9634.
5. Airlangga, G.; Liu, A. A Study of the Data Security Attack and Defense Pattern in a Centralized UAV–Cloud Architecture. *Drones* **2023**, *7*, 289. [\[CrossRef\]](#)
6. Branco, B.; Silva, J.S.S.; Correia, M. Cyber Attacks on Commercial Drones: A Review. *IEEE Access* **2025**, *13*, 9568–9577. [\[CrossRef\]](#)
7. Yu, A.; Kolotylo, I.; Hashim, H.A.; Eltoukhy, A.E.E. Electronic Warfare Cyberattacks, Countermeasures and Modern Defensive Strategies of UAV Avionics: A Survey. *arXiv* **2025**, arXiv:2504.07358.
8. Wang, L.; Chen, Y.; Wang, P.; Yan, Z. Security Threats and Countermeasures of Unmanned Aerial Vehicle Communications. *IEEE Commun. Stand. Mag.* **2021**, *5*, 41–47. [\[CrossRef\]](#)
9. Rana, M.; Mamun, Q.; Islam, R. Current Lightweight Cryptography Protocols in Smart City IoT Networks: A Survey. *arXiv* **2020**, arXiv:2010.00852.
10. Shor, P.W. Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer. *SIAM J. Comput.* **1997**, *26*, 1484–1509. [\[CrossRef\]](#)
11. FIPS 203; Module-Lattice-Based Key-Encapsulation Mechanism Standard. Federal Information Processing Standards Publication 203. National Institute of Standards and Technology: Gaithersburg, MD, USA, 2024. [\[CrossRef\]](#)
12. Ehsan, M.A.; Alayed, W.; Rehman, A.U.; Hassan, W.U.; Zeeshan, A. Post-Quantum KEMs for IoT: A Study of Kyber and NTRU. *Symmetry* **2025**, *17*, 881. [\[CrossRef\]](#)
13. Al-Meer, A.; Al-Kuwari, S. Physical Unclonable Functions (PUF) for IoT Devices. *arXiv* **2022**, arXiv:2205.08587.
14. Algamdi, A.M.; Alghamdi, H.M. Instant Counting & Vehicle Detection during Hajj Using Drones. *J. Image Graph.* **2023**, *11*, 204–211. [\[CrossRef\]](#)
15. Alafif, T.; Hadi, A.; Allahyani, M.; Alzahrani, B.; Alhothali, A.; Alotaibi, R.; Barnawi, A. Hybrid Classifiers for Spatio-Temporal Abnormal Behavior Detection in Massive Hajj Crowds. *Electronics* **2023**, *12*, 1165. [\[CrossRef\]](#)
16. Oláh, N.; Molnár, B.; Huszti, A. Secure Registration Protocol for the Internet of Drones Using Blockchain and Physical Unclonable Function Technology. *Symmetry* **2023**, *15*, 1886. [\[CrossRef\]](#)
17. Park, Y.; Ryu, D.; Kwon, D.; Park, Y. Provably Secure Mutual Authentication and Key Agreement Scheme Using PUF in Internet of Drones Deployments. *Sensors* **2023**, *23*, 2034. [\[CrossRef\]](#) [\[PubMed\]](#)
18. Zhang, Z.; Hsu, C.; Au, M.H.; Harn, L.; Cui, J.; Xia, Z.; Zhao, Z. PRLAP-IoD: A PUF-Based Robust and Lightweight Authentication Protocol for Internet of Drones. *Comput. Netw.* **2024**, *238*, 110123. [\[CrossRef\]](#)
19. Nair, A.S.; Thampi, S.M.; Jafeel, V. A Post-Quantum Secure PUF-Based Cross-Domain Authentication Mechanism for Internet of Drones. *Veh. Commun.* **2024**, *47*, 100780. [\[CrossRef\]](#)
20. Choi, J.; Son, S.; Kwon, D.; Park, Y. A PUF-Based Secure Authentication and Key Agreement Scheme for the Internet of Drones. *Sensors* **2025**, *25*, 982. [\[CrossRef\]](#) [\[PubMed\]](#)
21. Algarni, A.D.; Innab, N.; Algarni, F. A Verifiably Secure and Robust Authentication Protocol for Synergistically-Assisted IoD Deployment Drones. *PLoS ONE* **2025**, *20*, e0314475. [\[CrossRef\]](#)
22. Wazid, M.; Mittal, S.; Das, A.K.; Islam, S.K.H.; Alenazi, M.J.F.; Vasilakos, A.V. Designing Secure Blockchain-Based Authentication and Key Management Mechanism for Internet of Drones Applications. *J. Syst. Archit.* **2025**, *156*, 103189.
23. Tang, M.; Li, H. PUF-AKA-IoD: A Provably Secure PUF-Based Authentication and Key Agreement Protocol for IoD over 5G Networks. *J. Reliab. Secur. Comput.* **2025**, *1*, 25–40. [\[CrossRef\]](#)
24. Xie, Q.; Zhang, J. Lightweight Drone-to-Ground Station and Drone-to-Drone Authentication Scheme for Internet of Drones. *Symmetry* **2025**, *17*, 556. [\[CrossRef\]](#)

25. Al-Mekhlafi, Z.G.; Altmemi, J.M.H.; Al-Shareeda, M.A.; Al-Hchaimi, A.A.J.; Homod, R.Z.; Mohammed, B.A.; Alshammari, G.; Al-Dhlan, K.A.; Alrashdi, R.; Alkhabra, Y.A. ChebIoD: A Chebyshev Polynomial-Based Lightweight Authentication Scheme for Internet of Drones Environments. *Sci. Rep.* **2025**, *15*, 12874. [CrossRef] [PubMed]
26. Tawfeeq, N.H.; Yousif, M.; Al-Shareeda, M.A.; Almaiah, M.A.; Shehab, R. Lightweight and Quantum-Resistant Authentication for the Internet of Drones Using Dilithium Signatures. *Int. J. Innov. Res. Sci. Stud.* **2025**, *8*, 2842–2853. [CrossRef]
27. Xie, Q.; Wang, H. PUF-Based Secure and Efficient Anonymous Authentication Protocol for UAV towards Cross-Domain Environments. *Drones* **2025**, *9*, 260. [CrossRef]
28. Inter Planetary File System (IPFS). IPFS Documentation. Available online: https://docs.ipfs.tech/?utm_source=chatgpt.com (accessed on 20 May 2026).
29. Hossain, S.M.M.; Ghafoor, S.; Yelamarthi, K.; Yanambaka, V.P. SoftPUF: A Software-Based Blockchain Framework Using PUF and Machine Learning. *arXiv* **2025**, arXiv:2503.18421.
30. Alzahrani, M.Y.; Khan, N.A.; Georgieva, L.; Bamahdi, A.M.; Abdulkader, O.A.; Alahmadi, A.H. Protecting Attacks on Unmanned Aerial Vehicles Using Homomorphic Encryption. *Indones. J. Electr. Eng. Inform.* **2023**, *11*, 88–96. [CrossRef]
31. Román, R.; Arjona, R.; Baturone, I. A Quantum-Safe Authentication Scheme for IoT Devices Using Homomorphic Encryption and Weak Physical Unclonable Functions with No Helper Data. *Internet Things* **2024**, *28*, 101389. [CrossRef]
32. Goswami, S.; Doosti, M.; Kashefi, E. Hybrid Authentication Protocols for Advanced Quantum Networks. *arXiv* **2025**, arXiv:2501.09876.
33. Mefgouda, B.; Khan, R.; Alhussein, O.; Saleh, H.; Eldeeb, H.B.; Pandey, A.; Muhaidat, S. LPUF-AuthNet: A Lightweight PUF-Based IoT Authentication via Tandem Neural Networks and Split Learning. *arXiv* **2024**, arXiv:2407.06215.
34. Aissaoui, R.; Deneuville, J.-C.; Guerber, C.; Pirovano, A. Evaluating Post-Quantum Key Exchange Mechanisms for UAV Communication Security. In Proceedings of the 2024 AIAA DATC/IEEE 43rd Digital Avionics Systems Conference (DASC), San Diego, CA, USA, 29 September–3 October 2024.
35. Xia, T.; Wang, M.; He, J.; Yang, G.; Fan, L.; Wei, G. A Quantum-Resistant Identity Authentication and Key Agreement Scheme for UAV Networks Based on Kyber Algorithm. *Drones* **2024**, *8*, 359. [CrossRef]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.