An RBAC-Based Access Control and Security Architecture for UAV Networks in Precision Agriculture Using Software-Defined Drone Networking

Nadia Kammoun, Aida Ben Chehida Douss, Ryma Abassi
Innov'com Laboratory, Sup'com,
University of Carthage,
Tunis, Tunisia

***Abstract*—Unmanned Aerial Vehicles (UAVs), commonly referred to as drones, are widely employed in applications such as surveillance, delivery, mapping, and precision agriculture. Their flexibility, mobility, and cost effectiveness have accelerated their adoption in both civilian and industrial domains. However, the rapid evolution of UAV technologies introduces significant challenges related to limited resources, data processing constraints, and, most critically, security and privacy. Cyberattacks targeting UAV systems may result in data breaches, mission failures, operational disruptions, and risks to human safety. In our previous work, we proposed a lightweight identity authentication scheme based on Elliptic Curve Cryptography (ECC) and integrated it into a Software-Defined Drone Network (SDDN) architecture to ensure strong security with low computational overhead. Building on this foundation, the present study focuses on the agricultural domain, where UAVs are increasingly used for crop monitoring, precision farming, and environmental data collection. Due to the sensitivity of agricultural data and the involvement of multiple stakeholders, fine-grained access control is essential. The main contribution of this work is the design and evaluation of an SDDN-based security framework that integrates role-based access control (RBAC) with trust management to enable secure, scalable, and controlled UAV operations in agricultural environments. The framework restricts user actions according to predefined roles, improving system security and manageability. Simulation results demonstrate that the proposed approach effectively enforces access policies, enhances trust-aware decision making, and maintains low computational overhead suitable for resource-constrained UAV networks. Validation is conducted using Python and YAML-based configurations on Google Colab, confirming the practicality of the proposed solution.**

Keywords*—Unmanned Aerial Vehicles; Software-Defined Drone Network; role-based access control; security; attacks; trust management; authentication; access control

I. Introduction

In recent years, Unmanned Aerial Vehicles (UAVs), commonly known as drones, have seen rapid adoption across a wide range of civilian and industrial applications. Originally developed for military purposes, UAVs are now widely used in agriculture, environmental monitoring, logistics, infrastructure inspection, and public safety [1]. Their ability to operate remotely, cover large areas, and collect high-resolution data in real-time makes them valuable tools for both routine and critical missions. With the advancement of communication technologies and autonomous flight systems, UAVs are no longer limited to isolated tasks but increasingly operate in coordinated swarms, enabling more complex and large-scale operations [2]. However, this growth also introduces significant challenges related to data security, system integrity, and operational safety.

To address security challenges in dynamic and decentralized UAV environments, trust management plays a key role in evaluating the reliability of users, devices, and links. It enhances security by enabling data fusion, excluding malicious nodes, protecting privacy, and supporting context-aware services [3]. Trust management also improves decision-making through reputation and recommendation mechanisms, while optimizing resource allocation and scalability in multi-UAV systems.

Clustering complements trust management by organizing UAVs into hierarchical groups, improving scalability, energy efficiency, and control through designated cluster heads. Our previous work proposed a trust-based clustering approach for IoT systems, where nodes are evaluated based on trust levels, reputation, and recommendations to form reliable clusters. Malicious nodes are excluded, and a rehabilitation process allows trusted reintegration [4]. This method considers IoT constraints such as limited energy and processing.

However, trust alone is insufficient. Authentication remains critical to prevent attacks like spoofing, replay, and injection. Robust authentication protocols are necessary to verify identities and secure UAV coordination, especially in swarm-based or mission-critical scenarios.

In earlier work, we developed a lightweight identity authentication scheme tailored for UAV networks, utilizing Elliptic Curve Cryptography (ECC) within a SDDN architecture[5]. ECC, known for using smaller key sizes compared to Rivest–Shamir–Adleman (RSA) while maintaining equivalent security levels, was chosen due to its suitability for resource-constrained environments such as UAV systems [6], [15]. The proposed architecture enabled secure and scalable authentication by coordinating Drones (DR), a Ground Station (GS), and an Authentication Center (AC). The authentication phase involved message exchanges between these components, after which each drone generated a private key used for encrypt-

ing communications throughout the mission. To evaluate the robustness of the approach, we presented in [5], a case study demonstrating its resistance to various attacks and conducted a formal security analysis using the Automated Validation of Internet Security Protocols and Applications (AVISPA) tool, which confirmed the protocol's effectiveness.

In this work, we focus on the agricultural domain as the application context for our access control model, aiming to address the specific security needs of UAV swarm operations in precision farming. This work extends our previous efforts on UAV security by applying and enhancing those concepts in a real-world, domain-specific scenario. Unlike prior studies that approached UAV security from a general perspective, often focusing on isolated aspects such as authentication or trust evaluation, this study is centered on a concrete and highly relevant use case. In [7], the increasing integration of UAVs in agriculture for tasks like crop monitoring, soil analysis, and pesticide spraying highlights the need for structured and secure management of UAV functions and data. To address this, we propose in this work a comprehensive security architecture tailored to the agricultural UAV environment. This architecture integrated multiple layers: trust management for dynamic evaluation of UAV behavior and cluster head selection; lightweight authentication protocols based on ECC to verify identities with minimal computational overhead; and a RBAC model to ensure task-specific access rights. The RBAC model was demonstrated through a Python and YAML based implementation on Colab, simulating realistic role assignments, such as pilot, engineer, technician, etc. and evaluating their permissions to perform specific drone actions (e.g., takeoff, data capture, firmware updates). This end-to-end framework was supported by a detailed architectural diagram and tables summarizing each security layer, offering a unified view of how the different mechanisms interact to secure UAV operations. By grounding the simulation in a real-world scenario, this work not only demonstrated the applicability of access control in agriculture but also emphasized the importance of combining various security approaches to achieve scalable, reliable and context-aware UAV deployments.

The remainder of this study is structured as follows: Section II presents a review of related work on access control, trust management, and security frameworks in UAV networks. Section III outlines the main security challenges facing UAVs. Section IV summarizes our prior contributions related to lightweight authentication and trust-based node management which will be foundational components of the global architecture. Section V presents the research methodology, introducing the proposed role-based access control (RBAC) model tailored for Unmanned Aerial Vehicles (UAVs) in agricultural environments. This section also describes the integration of the proposed access control mechanism within a global Software-Defined Drone Network (SDDN) architecture, highlighting its core security components. Section VI discusses the experimental results and provides a detailed analysis of the performance and security implications of the proposed approach. Finally, the conclusion in Section VII summarizes the main findings of the study and outlines potential directions for future research.

II. Related Works

Several studies have explored the use of SDN architectures for Drones (SDDN) in UAV systems due to their scalability and flexibility in implementing diverse security mechanisms such as authentication, data integrity, and trust management. In this section, we summarize relevant contributions on securing a uav network based on SDDN as well as access control.

In their literature review, Alquwayzani and Albuali in [8] examined the integration of RBAC within Zero Trust Architectures (ZTA) for military UAV systems. They emphasized that RBAC improves security by restricting system privileges based on predefined roles, such as pilot, maintenance, or commander. This role-based restriction minimizes unauthorized access and potential data breaches, aligning with ZTA principles to enhance UAV system security.

In [9], Zhang et al. propose a decentralized access control mechanism for UAV swarms by integrating blockchain technology with Ciphertext-Policy Attribute-Based Encryption (CP-ABE). This approach aims to enhance data security by ensuring that only UAVs satisfying specific attribute-based policies can decrypt and access sensitive information stored on the blockchain. Unfortunately, this work while providing fine-grained access control, introduces significant computational complexity, particularly during encryption and decryption processes, in addition to policy update challenges and latency issues.

Lehmoud et al. (2025) [10] propose a comprehensive security architecture for UAV networks that leverages 5G, SDN/NFV, and AI techniques. Their design uses Curve448-based authentication and integrates SDN's global control to systematically distribute security policies across switches and UAVs. Anomaly detection is achieved through Shannon entropy and self-organizing maps in the NFV layer, ensuring rapid response to threats. While this work demonstrates the benefits of programmable security in SDDNs, it focuses heavily on detection and resilience—without explicitly addressing fine-grained access control policies for drone missions, which is a key distinction in our proposed framework.

Hu et al. (2024) [11] surveys emerging techniques in software-defined UAV networks and highlight several promising approaches: blockchain-assisted orchestration, SDN-NFV security slicing, and lightweight challenge–response authentication based on ECC. They note a trend towards decentralized identity management and programmable policies, though most implementations emphasize network-level protection rather than role- or attribute-based access control per mission. In contrast, our architecture layers in a full RBAC policy module, aligned with trust metrics—reputation, recommendations, and dynamic rehabilitation—enabling mission-tailored access governance embedded directly within the SDDN control plane.

After a comprehensive review of recent literature, it becomes evident that although some works have explored the integration of RBAC in UAV systems, and others have investigated the application of SDDN in agricultural contexts, there is a noticeable lack of research that combines these three elements within a unified framework. Furthermore, security components such as authentication and trust management are often treated separately rather than as part of an end-to-end architecture. In this study, we address this gap by proposing

an integrated approach that leverages RBAC, SDDN, and trust-based mechanisms to enhance the overall security and coordination of UAV swarms in agricultural missions. Our model offers a coherent and scalable solution that meets both the operational and security demands of precision farming environments.

III. UAV's Security Challenges

Security protocols for UAV's must be designed with low communication and computation costs, given the restricted onboard computing capabilities of these devices. To safeguard UAV networks against unauthorized access to sensitive information or potential harmful attacks, it is crucial to ensure a set of key security and privacy properties: authentication, confidentiality, integrity, authorization, non-repudiation, and availability. These are addressed in recent works on lightweight secure communication architectures and trust-based benchmarks in UAV systems [12], [13].

- Authentication: is fundamental for establishing secure communication within an UAV network. It enables the authentication and identification of UAVs participating in flight operations. Digital signature mechanisms are used to verify the trustworthiness of each UAV, allowing only authenticated UAVs to participate in the flight mission. Authentication also safeguards the UAV network from adversaries attempting to impersonate legitimate UAVs.
- Confidentiality: guarantees that data remains inaccessible or undisclosed to unauthorized individuals. Protecting sensitive data and data exchange between UAVs and the GS from unauthorized access is critical in UAV networks to prevent leaks of sensitive flight mission information, such as telemetry data and control commands. Encryption algorithms, both symmetric and asymmetric, can be employed to achieve confidentiality in UAV networks.
- Integrity: involves ensuring data consistency and trustworthiness during the communication process. Data integrity is crucial in UAV networks to prevent alterations like modifications, fabrications, substitutions, and data injections. The use of hash algorithms with advanced encryption mechanisms can be employed to ensure data integrity.
- Authorization: is a security method identifying a user's privileges or access levels to system resources. UAV networks should restrict access to data, allowing only authorized users to communicate with the network. Access control policies must be implemented to monitor and regulate access to resources.
- Non-Repudiation: prevents an entity from denying earlier agreements or activities. The UAV network must establish protocols to ensure non-repudiation through the adoption of a digital signature mechanism.
- Availability: refers to the immediate availability of services to authorized parties when needed for effective functioning. In mission-critical UAV networks, services must be available at all times without intentional or unintentional interruptions. Redundancy and backup mechanisms may be useful for highly critical information services to ensure availability. Additionally, the UAV system must be resilient against classical denial-of-service (DoS) attacks.

In this work, we focus on providing authentication, confidentiality, integrity, and authorization to our architecture to prevent and detect multiple types of attacks such as ID spoofing, man-in-the-middle, replay, Sybil, DoS, etc.

IV. Foundational Security Mechanisms and Prior Contributions

Let us recall that we proposed in previous papers [4], [5] different security measurements such as trust management for IoT networks in general and authentication based on SDDN architecture for UAVs. The trust management model excludes malicious nodes from an IoT network, while the authentication model ensures secure message exchanges between drones and the GS through the AC. Table I summarizes the main notations and their corresponding definitions used throughout the study.

TABLE I. Table of Notations used in the Proposed Architecture

Symbol	Description
GS	Ground Station
AC	Authority Center
$AuthC$	Authentication Controller
ACC	Access Control Controller
TMC	Trust Management Controller
CC	Centralized Controller
i	Index of drone in the network
DR_i	Drone i, where $DR_i \in \{DR_1, DR_2, \ldots, DR_n\}$
ID_{DRi}	Drone ID sent by drone DR_i
IDt_{DRi}	Drone ID already registered in the GS table
tab_{D_R}	Table of drone IDs created by GS
t	A prime number and the order of P
e	A network entity, where $e \in \{DR_i, GS, AC\}$
Pub_e	Public key of entity e
$Priv_e$	Private key of entity e
S	Digital signature
N	A nonce to prevent replay attacks
t	Timestamp
CH	Cluster Head
CM	Cluster Member
BS	Base Station
T	Trust Level
E	Energy Level
$Rep(a, b)$	Reputation value assigned by node a to node b
$RepTab_{DR}$	A table contains all reputation values of different DR_i

A. Trust Management Model

Trust management is a key component in securing IoT networks, helping to ensure reliable data fusion, protect user privacy, and exclude malicious nodes to maintain secure and dependable system operations. Our prior work [4] focused on IoT environments, where we implemented a clustering-based architecture involving CHs and CMs, managed by BSs. This hierarchical structure reduced the computational burden on individual nodes by delegating processing tasks to BSs, improving scalability, energy efficiency, and communication reliability across the network.

We designed a trust management system comprising four core algorithms: trust level calculation to evaluate node reliability, reputation assessment based on historical interactions, a recommendation mechanism for selecting new CHs, and a rehabilitation process to reintegrate previously blacklisted

nodes based on gradual trust recovery. These mechanisms were designed with resource-constrained IoT devices in mind, considering factors such as energy limits and processing capacity. This trust-based framework, initially proposed for general IoT applications, is now adapted to secure UAV networks in the agricultural domain. By integrating these mechanisms into clustered UAV architectures, we aim to enhance security, reliability, and efficiency in data collection and communication, particularly in large-scale deployments, where dynamic node behavior and harsh environmental conditions present ongoing challenges.

B. Authentication Model

We already proposed a study which introduces a secure authentication mechanism for SDDNs [5], ensuring trusted interactions among key entities: DRs, AC, and GS shown in Fig. 5. This model was divided into two phases: the authentication phase and the GS drones list confirmation phase. In the first phase, each drone initiated authentication by encrypting its identity and a random nonce using the AC's public key and sending this request to the AC. The AC decrypted the message and forwarded the drone's identity to the GS for verification. If the drone was recognized in the GS's registry, the GS confirmed the identity by returning an encrypted response with a new nonce to the AC. Following successful confirmation, the AC generated a public key for the authenticated drone and distributed it to the relevant network entities. The GS updated its internal drone table with the new key, while the drone computed its private key using a randomly chosen scalar to ensure confidentiality. Once all drones had been authenticated, the GS signed and broadcasted the finalized list of authorized drones, ensuring that each UAV was informed about its legitimate peers for communication. In the second phase, the GS initiated the mission by sending a start signal to all authenticated drones. Each drone then created role-specific messages, encrypted them using its private key, and transmitted them securely within the network. This approach ensured both the integrity and confidentiality of drone-to-drone communication throughout the operation.
These validated security mechanisms constitute the foundational components upon which the proposed access control model and global SDDN security architecture are constructed.

V. Research Methodology

A. Access Control Proposition

In UAV environments, access control is essential to ensure secure communication, data integrity, and mission reliability. Unauthorized access can lead to data breaches, manipulation of flight paths, or hostile takeovers, posing serious security and safety threats. Robust access control mechanisms help protect sensitive commands and telemetry data from interception or misuse. Moreover, they enable accountability and traceability in multi-user and multi-UAV systems. Thus, implementing strong and adaptive access control is critical for maintaining operational security and trust in UAV deployments [14].
In this study, we develop an access control model based on RBAC to detect and prevent unauthorized UAVs by assigning permissions according to predefined roles. This mechanism is combined with lightweight authentication and trust management, all integrated within a Software-Defined Drone Network (SDDN) architecture to provide a flexible and secure framework for UAV operations in agriculture.
In precision agriculture, the use of UAV swarms is becoming increasingly common for tasks such as crop monitoring, spraying, and soil analysis. To ensure secure and organized access to UAV functions and data, RBAC offers an effective solution. By assigning permissions according to user roles—such as field operator, drone technician, or system supervisor—RBAC reduces unauthorized access, limits operational risks, and streamlines task delegation. This model not only improves security but also enhances efficiency and accountability in managing UAV activities across large agricultural zones.

To illustrate the RBAC model in an agricultural UAV scenario, we implemented a simulation using Python and YAML configuration files on Google Colab. This setup allows us to define roles, permissions, and access rules in a clear and modular way.

1) Access policy definition: This policy, shown in Fig. 1, defines three operational roles relevant to UAV-based agricultural missions: *pilot*, *agri_engineer*, and *technician*. Each role is associated with a distinct set of permissions that determine what actions a user or agent can perform on a drone. The *pilot* role covers flight control operations, including arming, takeoff, landing, and autonomous return. The *agri_engineer* role includes tasks related to field data acquisition and analysis, such as collecting crop data, capturing aerial images, and analyzing soil conditions. Meanwhile, the *technician* role focuses on maintenance and operational readiness, enabling firmware updates, sensor calibration, and battery status checks. Roles are assigned to individual drones under the `assignments` section, thereby enforcing mission-specific and role-based access control within the swarm through a structured YAML policy embedded in the SDDN framework.

2) Access control function: The following function, as presented in Fig. 2, checks whether a user is allowed to perform a given action on a specific drone based on their assigned role.

This function implements a role-checking mechanism: it first verifies that the user has a role assigned for the given drone, then checks whether the requested action is permitted under that role.

3) Simulation results: We tested the function with a set of example scenarios, as shown in Fig. 3, to evaluate whether access is properly granted or denied.

These results, shown in Fig. 4, confirm that the access control mechanism correctly enforces permissions based on role assignments. Authorized actions, such as data collection by the agricultural engineer or drone landing by the pilot, are permitted. Unauthorized actions, such as a technician attempting to arm the drone, are denied as expected.

B. A Layered Model of the Global Architecture Proposition

The objective of the proposed global architecture is to establish a secure, scalable, and mission-aware UAV swarm framework that integrates authentication, trust management, and RBAC-based access control within an SDDN environment

```yaml
import yaml

# YAML swarm policy as a string
swarm_policy_yaml = """
roles:
  pilot:
    - arm
    - takeoff
    - land
    - return_home

  agri_engineer:
    - collect_crop_data
    - capture_images
    - analyze_soil

  technician:
    - update_firmware
    - calibrate_sensors
    - check_battery

assignments:
  drone_1:
    Dr. Sara: agri_engineer
    Captain Omar: pilot

  drone_2:
    Tech Ali: technician
    Captain Omar: pilot
"""
# Load it into a Python dictionary
policy = yaml.safe_load(swarm_policy_yaml)
```

Fig. 1. Access policy definition.

```python
def check_access(user, drone, action):
    assigned_users = policy["assignments"].get(drone, {})
    role = assigned_users.get(user)

    if not role:
        print(f"[X] {user} is not assigned to {drone}")
        return

    allowed_actions = policy["roles"].get(role, [])

    if action in allowed_actions:
        print(f"[√] {user} ({role}) is allowed to perform '{action}' on {drone}")
    else:
        print(f"[X] {user} ({role}) cannot perform '{action}' on {drone}")
```

Fig. 2. Access control function.

tailored for precision agriculture. Before detailing this architecture, we first outline our security objectives and introduce the network model.

In this proposition, while we adopted a UAV network for agriculture, we will merge the base station BS and the ground station GS because GS could perform BS task's made in our previous IoT network, while both serve as central communication points.

```python
check_access("Dr. Sara", "drone_1", "collect_crop_data")
check_access("Captain Omar", "drone_1", "land")
check_access("Tech Ali", "drone_1", "calibrate_sensors")
check_access("Visitor", "drone_2", "view_status")
```

Fig. 3. Results scenarios.

```
[√] Dr. Sara (agri_engineer) is allowed to perform 'collect_crop_data' on drone_1
[√] Captain Omar (pilot) is allowed to perform 'land' on drone_1
[X] Tech Ali is not assigned to drone_1
[X] Visitor is not assigned to drone_2
```

Fig. 4. Results output.

1) Security objectives: Our proposed solution is developed with the following security objectives:

- Establishing an SDDN architecture to enable efficient network management and dynamic reconfiguration, ensuring scalability for future enhancements. Additionally, the SDDN framework facilitates the rapid deployment of security mechanisms.
- Enabling mutual authentication between a legitimate DR and GS via AC by optimizing the number of exchanged messages during the authentication process.
- Preserving drone anonymity by preventing unauthorized entities from tracking its identity through encryption techniques.
- Ensuring secure communication between DR and between drones and the GS by employing private keys for message exchanges.
- Enhancing trust management by continuously evaluating node behavior, detecting malicious activity, and isolating compromised entities to maintain network integrity.
- Implementing access control mechanisms to restrict unauthorized access, enforce security policies, and dynamically manage permissions based on trust levels and authentication results.
- Strengthening the protocol's resilience against common security threats, including:
  -ID spoofing attacks
  -Man-in-the-Middle attacks
  -Replay attacks
  -Injection attacks
  -Malicious nodes infiltration
  -SYbil attacks
  -Data modification attacks
  -DoS attacks

This approach ensures a secure and scalable UAV network, while mitigating various cyber threats through authentication, trust management, and access control enforcement.

2) Basic concepts: In our proposal, we extend our previous [5] SDDN architecture model which is organized into three layers: the data plane, the control plane, and the application plane, as illustrated in Fig. 5.

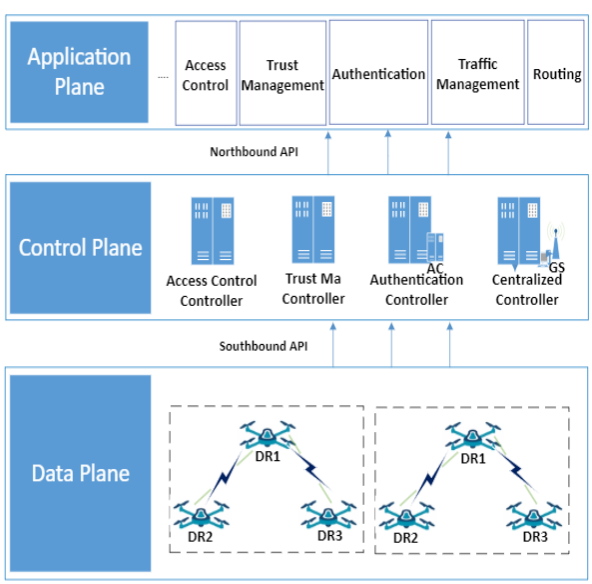

Fig. 5. Considered SDDN modeling.

- Data Plane: This layer comprises swarms of drones, which serve as the primary operational units within the UAV system. Each drone is constructed from lightweight materials and is outfitted with various components including sensors, actuators, payload devices, batteries, and GPS laser systems—to enhance maneuverability by minimizing weight. Typically, the navigation system and sensors are positioned at the front of the drone. Additionally, drones manufactured by different companies incorporate sophisticated features such as vibration damping, advanced communication modules, and compact onboard computing systems to effectively supervise, monitor, and control flight operations [16].These capabilities enable autonomous and semi-autonomous mission execution while supporting real-time data collection and coordination within the swarm. Furthermore, the modular hardware design allows drones to adapt to diverse agricultural tasks and operational conditions.
- Control Plane: This layer is composed of both centralized and decentralized controllers that work together to ensure secure and efficient UAV operations [17]. The centralized controller (CC), embodied by the GS, serves as the primary command and control center. It manages operational parameters, monitors drone sensors, oversees surveillance cameras, and enforces flight separation protocols. Additionally, the GS coordinates mission-critical activities, controls payload subsystems, and processes the data collected during drone operations.
  Complementing the GS, the control plane integrates several decentralized controllers to enhance security and system functionality. An Authentication Controller(AuthC), represented by the AC, verifies UAV system services by issuing certifications and generating cryptographic keys, thereby ensuring that all interactions are legitimate. To further secure the network, a Trust Management Controller (TMC) continuously evaluates the trustworthiness of nodes, aiding in dynamic decision-making and maintaining operational integrity. Moreover, an Access Control Controller enforces stringent policies to regulate node interactions and prevent unauthorized access to critical resources. Together, these controllers provide real-time problem resolution, robust data processing, and reliable networking services across the entire UAV environment.
- Application Plane: This layer represents the topmost level of our architecture, hosting both traditional network applications and drone-specific solutions. It delivers a broad range of services, including routing, traffic management, and security functions, alongside specialized applications such as mission planning, coordination, and data analytics tailored for UAV operations [16]. Additionally, this plane provides a comprehensive dashboard for real-time monitoring, performance visualization, and integration with external systems, ensuring dynamic control of network resources. To fortify security within this layer, we have integrated authentication, trust management, and access control mechanisms. These security measures guarantee that only authorized users and devices interact with the system, continuously assess the trustworthiness of communications, and enforce strict access policies to safeguard the integrity and efficiency of drone operations. Moreover, the application plane is designed to support future extensions such as UAV clustering, enabling scalable coordination and policy enforcement in large-scale agricultural deployments.

3) Security measures description: We detailed each proposed security measure in Table II and Table III, as well as in Fig. 6. We start by detailing the different steps of security measures shown in 5 layers as follows:

Layer 1: AUTHENTICATION: Authentication of DRs is performed through different steps as follows:

STEP 1-1:Authentication Initialization by DR_i :
To obtain its public key from AC, DR_i encrypts its identity ID_{DRi} using the public key of AC (Pub_{AC}), selects a random nonce N_{DRi} (to prevent replay attacks), and transmits the Init message to AC.

$$\text{Init} = (DR_i, AC, [\text{E}(ID_{DRi}, Pub_{AC}) — N_{DRi}]).$$

STEP 1-2: Request ID Verification by AC :
Upon receiving the message, AC decrypts it using its private key $Priv_{AC}$ to extract the drone's identity ID_{DRi}. It then forwards this ID to GS for verification to determine whether DR_i is under its control, using the Verif message.

$$\text{Verif} = (AC, GS, [\text{E}(ID_{DRi}, Pub_{GS}) — DR_i — N_{AC}]).$$

STEP 1-3: ID verification by GS:
The GS verifies the presence of the ID_{DRi} within its drone

Table Tab_{DR}. If the drone is recognized, we pass to the ID confirmation step, else, the authentication of DR_i is rejected.

STEP 1-4: ID confirmation by GS:
GS sends the Confirm message, which includes the encrypted ID using Pub_{AC} and a newly generated nonce N_{GS}.

Confirm = (GS,AC,[E(ID_{DRi},Pub_{AC})—N_{GS}]).

STEP 1-5: Pub_{DRi} Generation by AC:
Upon receiving confirmation from GS, AC generates the public key Pub_{DRi} for the authenticated drone and shares it with the registered entities in the network through the GenPub message.

GenPub = (AC,DR_i,[Pub_{DRi}—N_{AC}]).

STEP 1-6: Pub_{DRi} addition to Tab_{DR} :
GS updates its table, Tab_{DR}, which stores the drones DR_i and their corresponding IDs ID_{DRi}, by incorporating their public keys Pub_{DRi}.

STEP 1-7: $Priv_{DRi}$ Calculation by DR_i:
Once DR_i receives its assigned public key Pub_{DRi}, it selects a random number k and computes its private key as $Priv_{DRi} = k \cdot Pub_{DRi}$, ensuring that no external entity can derive it.

STEP 1-8: Signature calculation of Tab_{DR} by GS:
After the authentication of all DR_i, Tab_{DR} will be signed and broadcasted by GS to all authenticated drones, ensuring that each drone is informed about the legitimate drones authorized for communication during the mission.

STEP 1-9: Begin mission by GS:
After that, GS transmits the following message to begin the mission.

$$\text{BeginMiss} = (\text{GS}, \text{DR}_i, [S \mid\mid N_{\text{GS}}, t])$$

Layer 2: INTEGRITY AND CONFIDENTIALITY:
The following steps ensure integrity and confidentiality:

STEP 2-1: Message creation by DR_i:
When the mission begins, DR_i will start communicating by creating messages according to their role.

STEP 2-2: Message encryption with $Priv_{DRi}$ by DR_i:
Before sending any message, each drone uses its private key $Priv_{DRi}$ to encrypt it so that all messages are secured against integrity and confidentiality attacks.

STEP 2-3: Send message by DR_i:
After encryption, messages are secured and could be sent safely between different drones.

Layer 3: CLUSTERING:
Clustering enhances scalability, energy efficiency, and communication reliability in UAVs.

STEP 3-1:DR_i Clustering by GS In this step, GS will divide drones DR_i according to their positions and identify CHs and CMs.

Layer 4: TRUST MANAGEMENT:
The following steps ensure trust management during drones missions:

STEP 4-1: Trust Initialization:
GS provides for all drones a common starting trust level T_{DRi}=1

STEP 4-2: T_{DRi} calculation
T_{DRi}=T_{DRi}+x : T_{DRi} will decrease when nodes fail to respond to messages (-0.2), neglect to send important updates (-0.1), or transmit false information (-0.3).

STEP 4-3: Drone Exclusion:
DR_i will be excluded when its T_{DRi} attaints -1.

STEP 4-4: Blacklist Creation:
GS creates a blacklist for the malicious drones excluded from the network.

STEP 4-5: Reputation Calculation:
Every drone DR_i calculates the reputation value of other nodes according to their responses to requests.

STEP 4-6: Reputation Table Creation:
DR_i creates a Rep_{TabDR} containing all reputation values of different DR_i.

STEP 4-7: CH Selection:
GS use these reputation values to select the CH. CH will have the best reputation value Max rep(CH,DR_i).

STEP 4-8: DR_i Notification about new CH:
GS notifies other DR_i about the new CH.

STEP 4-9: Resolve Tie:
When 2 DR_i have the same reputation value, GS will select the max ID_{DRi} to choose the CH.

STEP 4-10: Send Recommendation:
CH uses the reputation values of its CMs to formulate and send a recommendation for GS to be replaced when its energy level falls below a threshold.

STEP 4-11: Well_behave Check:
When an excluded DR_i well behave it will have a chance to get back to the network.

STEP 4-12: Positive Trust Update:
T_{DRi} will increase by 0.1 for its good behavior by sending real notifications to GS about its state and detected events in

the network T_{DRi}= T_{DRi}+0.1.

STEP 4-13: Rehabilitation:
When T_{DRi} reaches 0, the malicious drone regains its trustworthiness and is removed from the blacklist.

Layer 5: ACCESS CONTROL:
Users authorization is ensured through the following steps:

STEP 5-1: Service Access Demand:
When a user U_j wants to perform an action using DR_i, an access demand will be sent to the Centralized Controller (CC).

STEP 5-2: Policy Request from CC to ACC:
CC should verify the access control policy of U_j already stored in the Access Control Controller (ACC).

STEP 5-3: Policy Response from ACC to CC:
ACC searches for U_j's access control policy and sends it back to CC.

STEP 5-4: User Role Request:
As a verification step, CC asks U_j about its role to match it to the role in the policy.

STEP 5-5: Access Control Function:
In ACC, the access control function implements a role-checking mechanism: First, it verifies that the user U_j has a role assigned for the given drone DR_i, then checks whether the requested action is allowed under that role.

STEP 5-6: Access Decision Notification:
After applying the function, U_j will be informed if access is confirmed or denied.

The following Table II and Table III summarizes the different security measurements.

4) Layered model description: As previously introduced, Fig. 6 illustrates the proposed layered model outlining the key components of our security environment. This figure shows the different steps of each mechanism to ensure a high-security level for an UAV network. The upper layer is about drones authentication, it details the process from ID_{DRi} verification by GS through AC to $Priv_{DRi}$ and Tab_{DR} signature calculation. Also, after the mission begins, messages are encrypted before being sent between the different components of the environment to ensure the integrity and confidentiality in Layer 2 services. Layer 3 is about clustering drones by GS before trust management process in Layer 4. In this layer, different steps are described such as trust level calculation, blacklist of malicious nodes creation, reputation calculation, recommendations and rehabilitation. The last layer (Layer 5) details access control steps from the user's service access demand to the access confirmation or denial according to RBAC policies. As a single table for the 5 layers becomes too long we have to split this table in two: Layer 1 and Layer 2 are represented in Table II and Layer 3, Layer 4 and Layer 5 are represented in Table III.

The second part of the splitted table, which is Table III, represents the Layers 3, 4 and 5.

VI. Results and Discussion

The proposed security architecture addresses the growing need for structured and lightweight protection mechanisms in agricultural UAV networks, where multiple stakeholders interact within a dynamic and resource-constrained environment. The experimental results obtained from the RBAC simulation demonstrate that role-based access control can be effectively implemented within an SDDN context to regulate user actions and enforce permission boundaries. By explicitly defining roles, permissions, and access rules through configuration files, the system ensures consistency, transparency, and reduced risk of unauthorized operations.

Recent works have explored security and access control in UAV networks using decentralized and zero-trust approaches. Dong et al. [18] propose a blockchain-based authentication scheme for multi-cluster UAV systems, improving robustness but introducing non-negligible computational and communication overhead. Similarly, Xie et al. [19] present a blockchain-assisted zero-trust model emphasizing continuous identity verification. In contrast, the proposed architecture relies on a lightweight ECC-based authentication combined with RBAC and trust management within an SDDN framework, enabling fine-grained access control with lower complexity and improved suitability for large-scale agricultural UAV deployments.

The RBAC evaluation confirms the feasibility of enforcing fine-grained access control without introducing excessive complexity, which is particularly important in agricultural scenarios involving farmers, operators, and service providers with distinct operational privileges. Furthermore, the modular design of the architecture detailed through operational phases, interaction schemes, and configuration tables facilitates extensibility and supports the integration of additional security and management mechanisms. In addition, the experimental observations highlight the practical alignment between the proposed access control logic and the operational requirements of agricultural UAV missions. The clear separation of control responsibilities and data access privileges contributes to improved system reliability and reduces the likelihood of configuration errors during mission execution. These characteristics are particularly valuable in real-world deployments, where operational simplicity and predictable behavior are essential for maintaining secure and efficient UAV-assisted agricultural services.

Nevertheless, this study has certain limitations. The experimental evaluation focuses specifically on the RBAC component, while other security mechanisms are validated through formal analysis rather than full-scale simulation. Additionally, performance metrics such as communication latency, energy overhead, and scalability under dense UAV swarm conditions are not yet quantitatively assessed, which may impact the generalizability of the results.

Future work will address these limitations by extending the validation of the complete architecture through inference-based modeling and formal verification, enabling formal reasoning over trust evolution and access decisions in dynamic environments. In addition, the integration of a lightweight

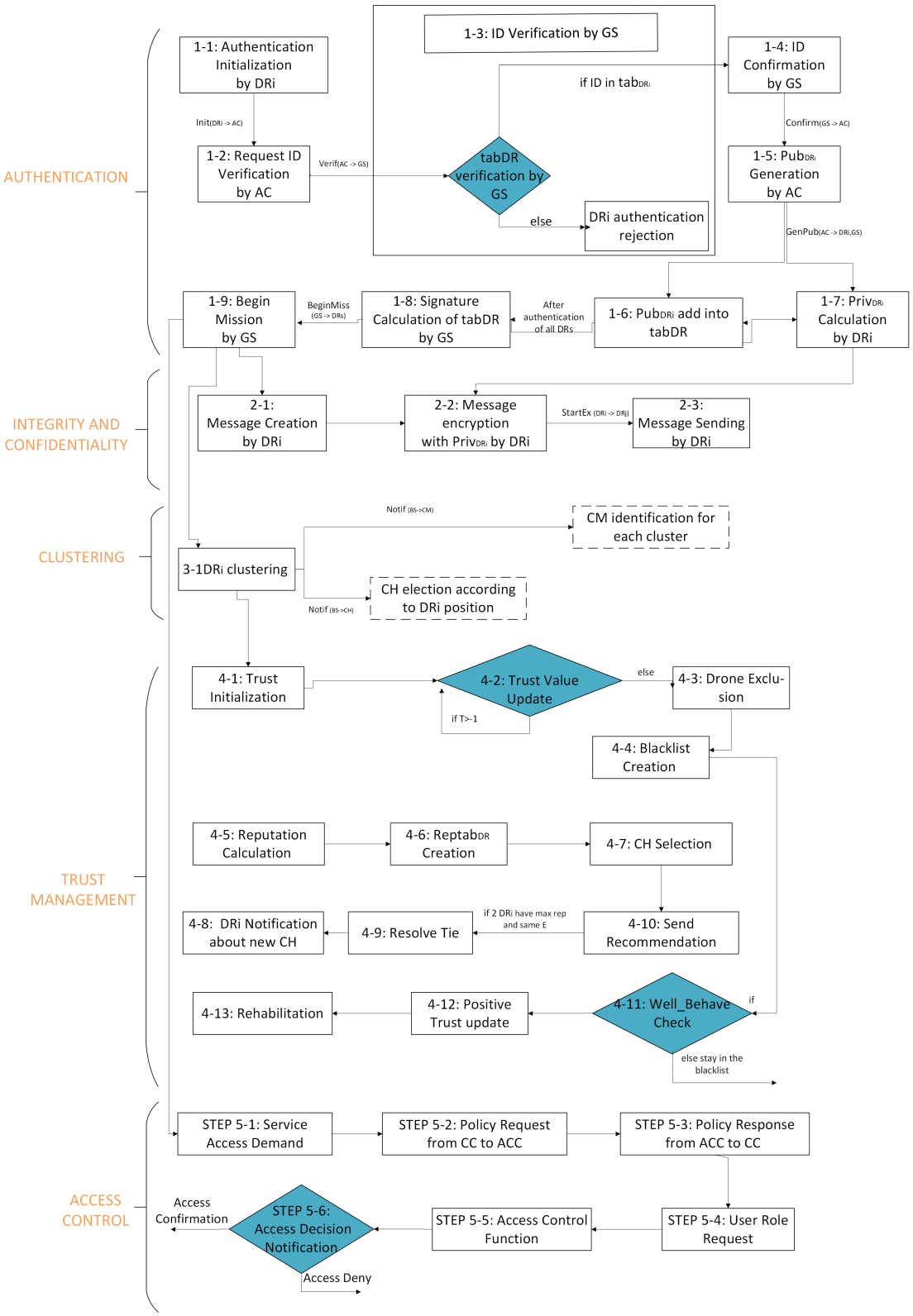

Fig. 6. Layered model description.

TABLE II. TABLE OF MESSAGES EXCHANGED (PART 1)

Steps	Service	Active Entities	Data Units	Data Units Parameter	Subservices	Sender	Receiver
1-1	Authentication Initialization by DR_i	DR_i, AC	Init	(DR_i,AC, [E(ID_{DRi},Pub_{AC})	-ID_{DRi} encryption with Pub_{AC}. -Addition of a random nonce N_{DRi}.	DR_i	AC
1-2	Request ID Verification by AC	AC, GS	Verif	(AC,GS, [E(ID_{DRi},Pub_{GS}) —DR_i—N_{AC}])	-ID_{DRi} encryption with Pub_{GS}. -Include the drone DR_i and addition of a random nonce N_{AC}	AC	GS
1-3	ID Verification by GS	GS	N.C	N.C	- ID_{DRi} verification by GS if it exists in Tab_DR or not. -If the ID is included in the table a confirmation will be sent from GS to AC . Else, DR_i rejection of authentication.	N.C	N.C
1-4	ID Confirmation by GS	GS, AC	Confirm	(GS, AC, [E(ID_{DR_i}, Pub_{AC}) || N_{GS}])	-Confirmation of the existence of ID_{DRi} in its drone table Tab_{DR}. -ID_{DRi} encryption with Pub_{AC}. -Addition of a random nonce N_{GS}.	GS	AC
1-5	Pub_{DRi} Generation by AC	AC, DR_i, GS	GenPub	(AC, DR_i, [Pub_{DRi}‖N_{AC}])	-Pub_{DRi} generation by AC so DR_i could calculate its private key $Priv_{DRi}$. -AC includes Pub_{DRi} and a nonce N_{AC}.	AC	DR_i, GS
1-6	Pub_{DRi} Addition in Tab_{DR}	GS	N.C	N.C	- Pub_{DRi} addition to GS table which already contains ID_{DRi} of drones which will start the mission.	N.C	N.C
1-7	$Priv_{DRi}$ Calculation by DR_i	DR_i	N.C	N.C	-Pub_{DRi} reception by DR_i -$Priv_{DRi}$ calculation using Pub_{DRi} and a random number k	N.C	N.C
1-8	Signature calculation of Tab_{DR} by GS	GS	N.C	N.C	-Digital signature S of Tab_{DR} calculation using GS private key $Priv_{GS}$. - Digital signature of Tab_D_R is shared with authenticated drones to identify legitimate nodes before mission start.	N.C	N.C
1-9	Beginning of the Mission by GS	GS, D_{Rs}	BeginMiss	(GS, DR_i, [S‖N_{GS}‖t])	-Start of the mission indication after a successful authentication phase. - GS sent S which contain the updates tab_D signed by $Priv_{GS}$, a nonce N_{GS} and a timestamp t.	GS	D_{Rs}
2-1	Create Message	DR_i	Msg	(DR_i, role, payload)	- Message generation by DR_i based on its assigned role in the mission.	DR_i	N.C
2-2	Message Encryption	DR_i	EncMsg	(DR_i, [E(payload, $Priv_{DR_i}$)])	- Message encryption using private key $Priv_{DR_i}$ to ensure confidentiality and integrity.	DR_i	N.C
2-3	Send Message	DR_i	EncMsg	(DR_i, DR_j, [E(payload, $Priv_{DR_i}$)])	- Transmission of the encrypted message to a peer drone DR_j.	DR_i	DR_j

TABLE III. TABLE OF MESSAGES EXCHANGED (PART 2)

Steps	Service	Active Entities	Data Units	Data Units Parameter	Subservices	Sender	Receiver
3-1	DR_i Clustering by GS	GS	Cluster	(GS, DR_i)	- GS clusters DR_i based on their positions and identifies CHs and CMs.	GS	DR_i
4-1	Trust Initialization	GS	InitTrust	(GS, $T_{DR_i} = 1$)	- GS assigns an initial trust value $T_{DR_i} = 1$ to all drones.	GS	DR_i
4-2	Trust Value Update	GS	UpdateTrust	($T_{DR_i} = t_{DR_i} + x$)	- Trust decreases due to non-responsiveness (-0.2), neglect (-0.1), or false data (-0.3).	GS	N.C
4-3	Drone Exclusion	GS	Excl	($T_{DR_i} = -1$)	- DR_i is excluded when trust level reaches -1.	GS	N.C
4-4	Blacklist Creation	GS	Blacklist	(GS, List)	- GS creates a blacklist for malicious or excluded drones.	GS	DR_i
4-5	Reputation Calculation	DR_i	RepCalc	(DR_i, other nodes)	- Each drone calculates reputations of others based on their behavior.	DR_i	N.C
4-6	Reputation Table Creation	DR_i	$RepTab_{DR}$	(DR_i, table)	- DR_i builds $RepTab_{DR}$ containing reputations of other drones.	DR_i	N.C
4-7	CH Selection	GS	SelectCH	max(Rep(CH, DR_i))	- GS selects CH with highest reputation.	GS	DR_i
4-8	DR_i Notification about new CH	GS	NotifyCH	(GS, CH info)	- GS notifies all DR_i of the selected CH.	GS	DR_i
4-9	Resolve Tie	GS	TieBreak	max(ID_{DR_i})	- If reputations are equal, GS selects CH by highest ID.	GS	DR_i
4-10	Send Recommendation	CH	Reco	(CH, GS)	- CH recommends replacement if its energy drops below threshold.	CH	GS
4-11	Well Behave Check	GS	CheckBehavior	(DR_i)	- If excluded DR_i behaves well, it's reconsidered.	DR_i	GS
4-12	Positive Trust Update	GS	TrustBoost	($T_{DR_i} = t_{DR_i} + 0.1$)	- Trust increases for good behavior and real alerts sent.	GS	N.C
4-13	Rehabilitation	GS	Rehab	($T_{DR_i} \geq 0$)	- If $T_{DR_i} \geq 0$, the drone is removed from the blacklist.	GS	DR_i
5-1	Service Access Demand	U_j, CC	AccessReq	(U_j, DR_i)	- U_j sends a request to access a service on DR_i to the Centralized Controller (CC).	U_j	CC
5-2	Policy Request from CC to ACC	CC, ACC	PolicyReq	(CC, U_j)	- CC asks ACC for the stored access control policy of user U_j.	CC	ACC
5-3	Policy Response from ACC to CC	ACC, CC	PolicyResp	(Policy(U_j))	- ACC retrieves U_j's policy and sends it back to CC.	ACC	CC
5-4	User Role Request	CC, U_j	RoleReq	(CC, U_j)	- CC asks U_j for its role to match with the access control policy.	CC	U_j
5-5	Access Control Function	ACC	Role-Check	(U_j, Role, Action)	- ACC verifies that U_j has a valid role and that the action is allowed for DR_i.	ACC	CC
5-6	Access Decision Notification	CC, U_j	AccessResp	(Confirm / Deny)	- CC sends a confirmation or denial of access to U_j based on policy check.	CC	U_j

clustering mechanism will be explored to enhance scalability, reduce control overhead, and improve resilience in large-scale agricultural UAV swarm scenarios.

VII. Conclusion

The integration of UAVs into agricultural environments represents a major advancement in precision farming, enabling efficient data acquisition, resource optimization, and real-time monitoring. However, these benefits also introduce significant security challenges due to the distributed, dynamic, and multi-stakeholder nature of agricultural UAV systems.

In this work, we proposed a comprehensive and lightweight security architecture tailored for Software-Defined Drone Networks (SDDNs) operating in agricultural contexts. The proposed framework provides a global and structured security vision that combines multiple complementary mechanisms, including ECC-based identity authentication, secure communication ensuring integrity and confidentiality, trust management based on dynamic behavioral evaluation, and role-based access control (RBAC). The architecture is detailed through clearly defined operational phases, interaction workflows, and configuration tables, offering a coherent end-to-end security model rather than an isolated mechanism. This holistic design enables controlled access, accountability, and adaptability in environments involving diverse actors such as farmers, operators, and service providers. From an implementation perspective, the experimental evaluation in this study focuses on the RBAC component, which was simulated using Python and YAML configuration files in a cloud-based environment. The results confirm that the proposed RBAC model enables clear, consistent, and enforceable access policies aligned with predefined roles, demonstrating its practicality within an agricultural UAV setting. The remaining security components, including ECC-based authentication and trust management, have been formally validated in our previous work using AVISPA, ensuring their robustness against common security threats.

Overall, this study establishes a solid security baseline for agricultural UAV networks by combining architectural design, formal validation, and targeted simulation. Future work will focus on formal inference-based validation of the complete architecture, the integration of lightweight clustering mechanisms for scalable swarm management, and performance evaluation under large-scale and highly dynamic UAV deployments.

References

[1] Z. Wang, Y. Li, S. Wu, Y. Zhou, L. Yang, Y. Xu, and Q. Pan, "A survey on cybersecurity attacks and defenses for unmanned aerial systems," *Journal of Systems Architecture*, vol. 138, p. 102870, 2023.

[2] A. A. Baktayan, A. T. Zahary, A. Sikora, et al., "Computational offloading into UAV swarm networks: a systematic literature review," *EURASIP Journal on Wireless Communications and Networking*, vol. 69, 2024.

[3] S. Ogunbunmi, Y. Chen, E. Blasch, and G. Chen, "A survey on reputation systems for UAV networks," *Drones*, vol. 8, no. 6, p. 253, 2024.

[4] N. Kammoun, A. B. C. Douss, and R. Abassi, "Trust-based algorithms for securing IoT clusters in an edge computing environment," in *Proc. IEEE 10th Int. Conf. Communications and Networking (ComNet)*, Tunis, Tunisia, Nov. 2023, pp. 1–6.

[5] N. Kammoun, A. B. C. Douss, and R. Abassi, "A novel lightweight authentication mechanism for UAVs based on SDDN architecture," in *Proc. 20th Int. Conf. Wireless and Mobile Computing, Networking and Communications (WiMob)*, Paris, France, Oct. 2024, pp. 1–6.

[6] M. A. Alazab, R. H. Abdalla, S. H. Ismail, et al., "FPGA-based dual-layer authentication scheme utilizing AES and ECC for unmanned aerial vehicles," *EURASIP Journal on Wireless Communications and Networking*, vol. 2024, Article 91, 2024.

[7] A. Smith, B. Li, and C. Wang, "Leveraging precision agriculture techniques using UAVs and emerging disruptive technologies," *Energy Nexus*, vol. 14, p. 100300, 2024.

[8] A. Alquwayzani and R. Albuali, "Integrating role-based access control within Zero Trust architectures for securing military UAV systems," *Journal of Cybersecurity and Defense*, vol. 12, no. 3, pp. 145–158, 2023.

[9] Y. Zhang, H. Liu, M. Wang, X. Chen, and J. Li, "Decentralized access control for UAV swarms using blockchain and attribute-based encryption," in *Proc. IEEE Int. Conf. Blockchain and Distributed Systems (BDS)*, 2023, pp. 102–110.

[10] A. A. M. Lehmoud, N. T. Obeis, and A. F. Mutar, "Design security architecture for unmanned aerial vehicles by 5G cloud network based implementation of SDN with NFV and AI," *Electric Electronics and Informatics (EEI)*, vol. 12, no. 1, Article 4239, 2023.

[11] X. Hu, Y. Li, Q. Zhang, et al., "Empowering UAV communications with AI-assisted software-defined networks: A review on performance, security, and efficiency," *Journal of Network and Systems Management*, vol. 31, pp. 69–90, 2025.

[12] M. H. Al-Salem and R. Raza, "Development of a Lightweight Secure Communication System for UAVs Enhanced by an Unconventional Chaotic Communication Architecture," *Journal of Intelligent & Robotic Systems*, vol. 111, art. 85, 2025.

[13] F. T. Madhuvanthi and A. Revathi, "A survey on UAV network for secure communication and attack detection: A focus on Q-learning, blockchain, IRS and mmWave technologies," *KSII Transactions on Internet and Information Systems*, vol. 18, no. 3, pp. 779–800, 2024.

[14] W. Shafik, M. Matinkhah, and F. Shokoor, "Cybersecurity in unmanned aerial vehicles: A review," *Int. J. Smart Sensing and Intelligent Systems (IJSSIS)*, vol. 16, no. 1, pp. 1–12, 2023, doi: 10.2478/ijssis-2023-0012.

[15] M. Usman, R. Amin, H. Aldabbas, and B. Alouffi, "Lightweight challenge–response authentication in SDN-based UAVs using elliptic curve cryptography," *Electronics*, vol. 11, no. 7, p. 1026, 2022.

[16] M. H. Alsharif, A. K. Sutradhar, and S. Kim, "Software-defined networking for unmanned aerial vehicle systems: Architecture, challenges, and future directions," *IEEE Access*, vol. 12, pp. 115430–115452, 2024.

[17] J. Kang and T. Shon, "SDN-enabled control and communication frameworks for cooperative UAV networks: A comprehensive review," *Computer Networks*, vol. 245, p. 110927, 2025.

[18] Y. Dong, H. Li, and X. Chen, "Blockchain-based secure authentication and access control for multi-cluster UAV networks," *Ad Hoc Networks*, vol. 146, p. 103157, 2023.

[19] J. Xie, Z. Zhang, and L. Wang, "A zero-trust security framework for UAV networks with lightweight identity verification," *IEEE Internet of Things Journal*, vol. 11, no. 4, pp. 6123–6136, 2024.